UNIVERSITY OF READING

Aspects of Emergent Cyclicity in Language and Computation

ARGUMENTS FOR MIXED COMPUTATION

PhD Thesis

School of Psychology and Clinical Language Sciences

Diego Gabriel Krivochen

e-mail: d.g.krivochen@pgr.reading.ac.uk

03/2018

Table of Contents

Introduction and health warnings

The present work attempts to bring together insights, questions, and methodology from three main disciplines in order to understand some crucial processes in cognition: physics, mathematics, and linguistics. In this introduction, we will present the specific aspects of these disciplines that will be relevant for this monograph, and hint at their interrelations, to be explored in depth in the course of the thesis.

The question we aim to answer is, very broadly put,

> *How do we make sense of external stimuli in the form of objects and relations? And, how do we externalize the internal representations of the phenomenological world?*

Such a question cannot be answered within the limits of a thesis, if a general answer is in principle possible. We will narrow our focus here, setting it to stimuli corresponding to *languages* of different kinds, natural and artificial. In order to investigate how linguistic stimuli are perceived, parsed, and produced, we cannot rely on a single discipline: we would be oversimplifying the problem to the extent that any provisional answer would hardly be applicable outside very narrow margins. The presentation of tools for the development of a consistent and fully explicit formal system with tools drawn from the exact and natural sciences, applied to linguistic phenomena, as manifestations of neurocognitive processes, is one of the main aims of the present work.

0.1 Physics

We will primarily be arguing for the hypothesis that we are looking at a complex system with interesting internal dynamics, which varies over space and time. Moreover, those variations are not random, but derive from the interaction between the system and external factors, as well as from the interplay among the system's own internal tendencies. This kind of system is usually referred to as a *dynamical system*, with the property of being *nonlinear* if the outputs are not directly proportional to the inputs. Some basic properties of these systems are listed below (based on Hasselblat and Katok, 2003; Ladyman, Lambert and Wiesner, 2012; Baranger, 2004; Boccara, 2002):

1)
   a) Open to external influence
   b) Complex (i.e., contain subsystems)
   c) Dynamic (i.e., change over time)
   d) Emergence (i.e., the collective behaviour of the system is not a linear function of the behaviour of its individual components)
   e) Nesting / hierarchical organization
   f) Existence of feedback loops

The physical properties of the systems we will look at are essentially those in (1). More specifically, we will focus on a particular kind of *nonlinear* dynamics, that which emerges from an irreconcilable tension between opposing tendencies, a so-called *dynamical frustration* (Stein and Newman, 2011; Binder, 2008; Moesner and Ramírez, 2006). The concept was first developed to analyse disordered magnetic systems in which two nearby locations in a lattice are equally likely to be interacting via ferromagnetic (aligned) or antiferromagnetic (antialigned) interaction, a kind of system known as *spin glass*. However, its domain of applicability has recently been extended to dynamical systems (i.e., systems whose behaviour changes over time) more generally, including, for instance, the modelling of neural networks (Dotsenko, 1995). The relation between complex dynamical systems and dynamical frustrations is made explicit by Binder (2008):

> *The essence of complexity may lie here: A system without dynamical frustration will either settle to equilibrium or grow without bounds.* (Binder, 2008: 322)

It will be our task to identify the opposing tendencies as well as the means for their resolution given finite resources and physical limitations in the case of human cognition from a computational perspective. We will see that, if there is a dynamical frustration at the core of cognitive processes (and if these processes are *computational* processes), some very interesting empirical predictions arise, like the existence of *cycles* in computations: we will derive functions that have a clear period, and are thus suitable for modelling cyclic, oscillatory processes, also being flexible enough to admit external factors influencing the maximum at each cycle (a crucial feature, as we will see), but keeping the derivative function untouched. The development of any function over time within a phase space (the space in which all possible states of the system are represented) divides that space in two (regardless of its dimensionality); the result is physically interpretable as zones where the possibility of finding a solution that satisfies a certain equation is relatively high and zones where that possibility is very low. The evolution of the system in time progressively restricts the phase space by cycles, making solutions easier to find, at the cost of increasing the dimensional complexity of the space where solutions are to be found. This cost-benefit ratio applying by cycles will be at the very core of our proposal

Interactions among measurable results will be modelled by means of a very reliable tool taken from quantum physics: so-called 'Dirac notation', or, perhaps more creatively, 'bra-ket notation' (because if you put a bra- and a –ket together, they form a pair of angular brackets, ⟨⟩). This notation will be explained in detail in **Part III**, but for the time being, let us present the general case for quantum measurements in bra-ket form:

2) $H |\psi\rangle = \lambda |\psi\rangle$

(read: 'H acting on psi equals lambda acting on psi')

The physical interpretation of (2) is simple: H is an *n*-by-*m* matrix, where all possible measure results are represented (what is referred to as the 'observables'). $|\psi\rangle$ is called a 'ket vector' (in this configuration, an 'eigenvector'), and is a vector of *n* components ordered in a row, with each component being a dimension along which measurement takes place. The *ket* represents a specific state of the system under consideration. If, say, we polarize monochromatic light, or prepare an electron with a certain spin (up or down), then, given a matrix H where H is a 'Hermitian' matrix[1], then H contains all possible measurables, and the *ket* vector encodes the

---

[1] To say that H is 'Hermitian' means that $H = H^\dagger$, the *transposed complex conjugate* of H in which rows become columns, columns become rows, and all imaginary terms change signs. For instance, consider the matrix M:

$$M = \begin{pmatrix} 1 & 1-i \\ 1+i & 0 \end{pmatrix}$$

The transposed complex conjugate of M is $M^\dagger$, where we transform rows into columns while changing the sign of all imaginary terms (those terms containing *i*):

specific experimental setting we have prepared.We can encode states of a system in terms of components, for instance, a photon polarized along the *x* axis would be defined by the vector:

3) $|\alpha\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$

Meaning, '1 component in axis *x*, 0 components in axis *y*' (of course, we could have done it the other way around, this is largely a matter of convention). λ is an 'eigenvalue', which represents the result of measurement, typically 1 or 0(2) (i.e., given the operator H acting on the eigenvector we can get the state of the field $|\psi\rangle$ or not, respectively). Let us see an example. Let H be the Hermitian matrix corresponding to ½ integer spin measurement (the Pauli matrix σ_3), and let $|u\rangle$ and $|d\rangle$ be the eigenvectors corresponding to 'up' and 'down' spin, respectively. Let us make everything explicit:

4) $\sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$

$|u\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$

$|d\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

Now, we can act with the Hermitian matrix on either of the vectors, which gives us an eigenvalue λ times the corresponding vector. The interpretation of the matrix is, basically, 'is the spin up or down?', and the eigenvalue tells us that, for a particular configuration we want to measure, the answer is either 'up' or 'down', each answer corresponding to an eigenvalue.

Going back to more general issues, another common expression in quantum mechanics is (5) (Dirac, 1958; Feynman, 2006: 3-2):

5) $\langle \alpha \mid \beta \rangle$

Where $\langle \alpha|$ is called a '*bra* vector', and is the transposed complex conjugate of the *ket*: the row becomes a column and imaginary terms change signs. In general, this is interpreted as 'we prepare a system in state β and measure it in state α'. Since both *bra* and *ket* are vectors, we can take their inner product, which will be a scalar: this is known as the 'probability amplitude' for the measurement of β into α. If we want to know the probability distribution, we have to multiply $\langle \alpha \mid \beta \rangle$ by $\langle \alpha \mid \beta \rangle^*$, where $\langle \alpha \mid \beta \rangle^*$ is the complex conjugate of $\langle \alpha \mid \beta \rangle$ (i.e., $\langle \alpha \mid \beta \rangle$ with the signs of all their respective *imaginary* terms changed). If we are dealing with expressions with zero imaginary parts, then $\langle \alpha \mid \beta \rangle\langle \alpha \mid \beta \rangle^* = \langle \alpha \mid \beta \rangle^2$ (i.e., the ordinary 'square'). The use of Dirac notation will be particularly helpful because, unlike most works in cognitive science and neurophysics, we work with mutually orthogonal results.

So far we have introduced some concepts we will need to account for the behaviour of the systems we are interested in, but have said nothing about the kinds of objects (or 'outputs')

$M^\dagger = \begin{pmatrix} 1 & 1-i \\ 1+i & 0 \end{pmatrix}$

Since $M = M^\dagger$, we say that M is Hermitian.

[2] -1 is also a possible value, but in that case, both the eigenvector and the eigenvalue are multiplied by -1, which renders 1 acting on $-|\psi\rangle$

thereby derived. Nor have we said anything about the properties of the *phase space* where solutions to the frustration are to be found.

0.2 Mathematics

A strict separation between physics and mathematics in the present work would be fallacious at best, and straightforwardly wrong at worst. For the purposes of this introduction, we set the boundary between them in terms of real objects and their modelling: in this subsection we will present the tools we will use to model the physical characteristics introduced above.

If we are dealing with a system which changes over time, it would be interesting to know *the rate of change* of some particular variable -or set thereof- of the system per time unit. This is doable by means of so-called *differential equations*, a basic tool in infinitesimal calculus. Generally speaking, an ordinary differential equation has the form

6) $\frac{dy}{dx}$ for $y = f(x)$

That is, we are interested in knowing the rate of change of y (a function of x) as the rate of change of x goes to zero (in Leibniz's terms, it is *infinitesimal*). So, we want to know how much y changes as x changes. In (6) we have an equation with a single dependent variable, but things are not always that simple; we will see that we have to resort to multivariable equations in order to provide a reasonably accurate model of the sort of dynamical system we are interested in, since there are a number of relevant interconnected variables to take into consideration, which we will make explicit in the course of this work. A slight adjustment is needed, then:

7) $\frac{\partial u}{\partial x}$ for $u = f(x, y)$

(7) is a function that takes *two* variables, x and y. A complex system of the kind we will be exploring requires us to consider more than a single dependent variable evolving per unit of time, and in this respect (7) is a simple yet extremely effective formalism to capture such dynamics. Equations of the kind in (7) are called *partial differential equations*, and we will meet them during our journey (they are particularly useful in neural field models, which play along very nicely with an oscillatory engine for the computation). We will see that the cyclic nature of computations, which derives from the physical notion of *dynamical frustration*, restricts the kinds of functions we can choose from.

Differential equations are customarily used to model the overall behaviour and evolution of dynamical systems, but what about the phase space in which that system evolves? That is, which are the properties of the space in which states that satisfy the dynamics of the system for a certain combination of variables –in simpler terms, 'solutions'- are represented? What are the mathematical and geometrical properties of the objects derived in that space? Here is where *topology* can be of help. We want to know how symbolic representations are derived (how we can 'deform' a particular space to get different emergent properties and objects given physical constraints over deformation possibilities), but also how they behave, and how their own topological properties as well as those of the space in which they interact constrain behaviours. The simplest assumption at this point is that the space is Euclidean, and that the objects are Euclidean as well. While reasonable, this is not a methodological choice as much as it is an empirical issue, to be determined upon close inspection of a particular kind of cognitive representation. For the time being, let us just say we will appeal to the topological concept of a *manifold*, which is an *n*-dimensional object that *locally* displays Euclidean properties, but globally has a non-Euclidean behaviour. A fine example is the well-known Klein bottle (below), a non-oriented manifold derived from the also well-known Möbius strip[3], whose *local* properties we can study as being analogous to what we would find in a Euclidean space of dimension *2* (usually referred to as $\mathbb{R}^2$).

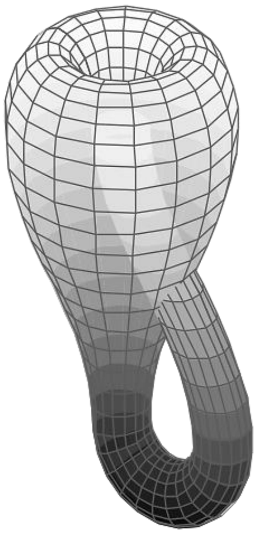


Figure 0: Klein Bottle

The manifolds which we will study here using tools and methodology from differential topology and (multivariable) calculus are constructed by means of computations, which affect the topology of the space in which manifolds interact and relate to each other in many interesting ways to be explored in this work. The focus will be set on the *limits* to such interactions, beyond which certain operations *cannot* happen: crucial -and testable- consequences for linguistic structures will follow from these unambiguous and fully explicit formulations.

We have said that computations cyclically restrict the phase space within which we can find solutions to a relevant equation. Mathematically, this means that we have to devise some tool to capture this process of spaces restricting, and expanding, cyclically (see also the discussion in Saddy, 2018 about the cyclical properties of embodied computation): *integrals* will prove extremely useful in this respect. As a reminder, the integral of a function *f*(*x*) over a closed interval [*a*, *b*] has the form

[3] As the following limerick by the mathematician Leo Moser gracefully shows:

> "A mathematician named Klein
> Thought the Möbius band was divine.
> Said he: 'If you glue
> The edges of two,
> You'll get a weird bottle like mine'."

8)

$$\int_a^b f(x)dx$$

The integral over a function with a single variable gives us the area below the function between points *a* and *b* in the *x* axis. If the function has more variables, i.e., if it is defined in more than 1 dimension, the integral has to be taken for each dimension separately, in a recursive process but with a result of order $n + 1$, where n is the number of dimensions that define the function (e.g., a line is unidimensional, and its integral is an area, thus bidimensional). We will expand (8) as appropriate, when we consider more variables.

0.3 Linguistics

So far we have talked about a 'system', being deliberately vague. Or, rather, non-specific. The reason is that what we have said, and most of what we will say, holds for more than a single 'cognitive system', provided that we have made no reference to system-specific concepts. However, we will set our focus on one of these systems: language. Natural languages seem to fulfil all the requirements specified in (1) for being systems of the kind studied by nonlinear dynamics:

a) they are *open* because whatever cognitive system is appropriate for them must interact with other cognitive systems (in the case of language, those in charge of sound and meaning representations) as well as with the phenomenological world. We would like to stress our position against studying language in the 'mind-brain' (an expression that is commonly used in Generative Linguistics, see Chomsky, 1986b: 29 for an example) completely isolated from other systems –cognitive, external- insofar as those systems impose conditions upon *language design* (in the words of Chomsky, 2005, 2007): we are referring, for instance, to physical limitations on possible neural networks, in turn deeply related to optimization algorithms (designed to complete a given task minimizing the number of steps, the energy required to complete it; in general, maximize the output of a given task while minimizing the cost); factors defining the phase space for cognitive dynamics and the possibilities for the computation to 'point' to certain places within that space, among other aspects which we will analyse in the course of the work.

b) they are *complex* because there is a structure-building computational process that interfaces with other cognitive systems that each impose different requirements, thus giving rise to a tension insofar as these requirements are orthogonal to each other;

c) they are *dynamic*, on the scale at which we are working, because computations proceed in real time and incrementally. This 'derivational diachrony' is an essential feature of an account of a pair of integral characteristics of linguistic derivations and representations, which are *locality* –in representational terms- and *cyclicity* –in derivational terms.

d) they present *emergent* properties. *Emergence* is also a key point to take into account when considering the interpretation of linguistic structures: at each cycle, the interpretation of the derived representation (particularly at the semantic component) is not a linear function of the computations performed at that cycle alone, but is influenced by (i) previous cycles (by means of representational remnants of objects, sets, and ensembles –Feigenson, 2011- in the *episodic buffer*, Baddeley, 2007; a process that is intimately related to the concept of *feedback loops*), (ii) expectations and anticipation processes (involving access to *long term memory*), as well as (iii) interactions between elements within and across cycles, in ways we will attempt to derive from the physical properties of the system as well as the topology of the objects and the workspace in which they are derived;

e) considerations of *hierarchy* and *nesting* will lead us to revisit the concept of *phrase structure* as commonly assumed in linguistics (a task we initiated in Krivochen, 2015a, 2016b; Krivochen and Schmerling, 2016), and to critically examine the relevance of the Turing-based notion of *recursion* for cognitive capacities (Watumull et al., 2014, among many others). Empirically, our testing domain will be *locality conditions* on the establishment of dependencies among syntactic objects, building upon the foundational (and to our mind, still unparalleled) work of Ross (1967) and his formulation of 'island constraints' upon syntactic transformations.

At this point it is necessary to make it clear what we will mean by 'language' in the context of our interdisciplinary inquiry. *We define a **language** computationally, as a set of mutually interacting constraints on an otherwise unlimited productive procedure (which is essentially a normalized L-system; see below); physically, as a complex system evolving in time; and mathematically, as a set of topological transformations (essentially, the imposition of a metric over a field). These are crucial points, for we can only ask questions about 'language' if we know what we are dealing with* (i.e., if we have an answer for the simple question 'what is language?': otherwise, the whole enterprise would be vacuous, in our opinion). *Other uses of the word 'language' are possible*[4] *(and sometimes yield consistent systems), but this is a matter of definition, thus, of methodology.* We will do our best not to take anything for granted.

We will compare and contrast the limitations on structure building and mapping operations that natural language displays with those exhibited by formal grammars as implemented in different automata within the Chomsky Hierarchy (Chomsky, 1959: 143). More specifically, we will deal with Lindenmayer grammars (L-grammars) and their emergent properties, which have been the focus of many cross-disciplinary studies, involving not only linguistics –Uriagereka, 1998; Saddy and Uriagereka, 2004- but also biology -Pruzinkewicz and Lindenmayer, 1990-, computer science, and automata theory -Hopcroft and Ullman, 1969-, among others. We expect L-grammars to reveal the potential of and limitations on formal languages for modeling natural language (and other cognitive capacities), based on the physical-mathematical framework outlined above, which will be developed in detail throughout this thesis. Specifically, we will attempt to derive the notions of *cycle* and *island* for the application of operations from (i) the characteristics of the topological spaces in which manifolds are generated, (ii) the properties of the manifolds themselves, and (iii) the way in which the generative system interacts with (and is limited by) other systems; in **Part IV**. *Straightforward*

[4] For instance, Chomsky famously defined a language as a set of strings: *I will consider a **language** to be a set (finite or infinite) of sentences, each finite in length and constructed out of a finite set of elements. All natural languages in their spoken or written form are languages in this sense, since each natural language has a finite number of phonemes (or letters in its alphabet) and each sentence is representable as a finite sequence of these phonemes (or letters), though there are infinitely many sentences. Similarly, the set of 'sentences' of some formalized system of mathematics can be considered a language.* (Chomsky, 1957: 13. Highlighted in the original)

and explicit predictions pertaining to locality conditions and islandhood phenomena following from formal considerations will be the main empirical contribution of the present thesis.

Overall, we will argue against a computationally uniform template for cognitive processes, and stress the importance of considering interactions between systems when choosing or developing formal tools to model a particular system. The need to build bridges among all three of the main disciplines mentioned here will hopefully be obvious throughout, and we hope our conclusions will be accessible to readers coming from different disciplines, attempting to shed some light on cognitive phenomena.

Roadmap

This thesis has four parts, which correspond to the presentation and development of a theoretical framework for the study of cognitive capacities *qua* physical phenomena, and a case study of locality conditions over natural languages.

Part I deals with computational considerations, setting the tone of the rest of the thesis, and introducing and defining critical concepts like 'grammar', 'automaton', and the relations between them[5]. Fundamental questions concerning the place of formal language theory in linguistic inquiry, as well as the expressibility of linguistic and computational concepts in common terms, are raised in this part.

Part II further explores the issues addressed in **Part I** with particular emphasis on how grammars are implemented by means of automata, and the properties of the formal languages that these automata generate. We will argue against the equation between *effective computation* and *function-based computation*, and introduce examples of computable procedures which are nevertheless impossible to capture using traditional function-based theories. The connection with cognition will be made in the light of *dynamical frustrations*: the irreconciliable tension between mutually incompatible tendencies that hold for a given dynamical system. We will provide arguments in favour of analyzing natural language as emerging from a tension between different systems (essentially, semantics and morpho-phonology) which impose orthogonal requirements over admissible outputs. The concept of *level of organization* or *scale* comes to the foreground here; and apparent contradictions and incommensurabilities between concepts and theories are revisited in a new light: that of dynamical nonlinear systems which are fundamentally *frustrated*. We will also characterize the computational system that emerges from such an architecture: the goal is to get a syntactic component which assigns the simplest possible structural description to sub-strings, in terms of its computational complexity. A system which can oscillate back and forth in the hierarchy of formal languages in assigning structural representations to local domains will be referred to as a *computationally mixed system*.

Part III is where the really fun stuff starts. Field theory is introduced, and its applicability to neurocognitive phenomena is made explicit, with all due scale considerations. Physical and mathematical concepts are permanently interacting as we analyze phrase structure in terms of pseudo-fractals (in Mandelbrot's sense) and define syntax as a (possibly unary) set of topological operations over completely Hausdorff (CH) ultrametric spaces. These operations,

[5] In the context of this work, 'a X string' (where X stands for 'regular', 'context free', 'context sensitive'…) means 'a string that belongs to a language that can be accepted by an automaton of class X'. For example, a 'regular string' is a string that belongs to the set of strings that can be accepted by a FSA.

which makes field perturbations interfere, transform that initial completely Hausdorff ultrametric space into a metric, Hausdorff space with a weaker separation axiom. Syntax, in this proposal, is not ‘generative’ in any traditional sense –except the ‘fully explicit theory’ one-: rather, it partitions (technically, ‘parametrizes’) a topological space. Syntactic dependencies are defined as *interferences between perturbations over a field*, which reduce the total entropy of the system *per* cycles, at the cost of introducing further dimensions where attractors corresponding to interpretations for a phrase marker can be found.

Part IV is a sample of what we can gain by further pursuing the *physics of language* approach, both in terms of empirical adequacy and theoretical elegance, not to mention the unlimited possibilities of interdisciplinary collaboration. In this section we set our focus on *island* phenomena as defined by Ross (1967), critically revisiting the most relevant literature on this topic, and establishing a typology of constructions that are *strong islands*, which cannot be violated. These constructions are particularly interesting because they limit the phase space of what is expressible via natural language, and thus reveal crucial aspects of its underlying dynamics. We will argue that a dynamically frustrated system which is characterized by displaying mixed computational dependencies can provide straightforward characterizations of cyclicity in terms of changes in dependencies in local domains.

Acknowledgements

I would like to express my deepest gratitude to the following individuals (in strict alphabetical order), for reasons each of them knows very well and that would be otiose to detail here (their infinite patience with my inquiries is worth mentioning as one of their common denominators):

> Daniel Fryer, Itziar Laka, Manuel Leonetti, David Medeiros, Massimo Piattelli-Palmarini, Mike Putnam, Haj Ross.

A separate ‘thank you’ goes to Peter Kosta, who believed in me way back then…

The following brothers in arms deserve a special heartfelt mention, for going well above and beyond the call of duty to help me with this enterprise:

> Ana Bravo, Luis García Fernández, Doug Saddy, Susan Schmerling, Juan Uriagereka.

To all of them we owe more than I could start to repay in exponential time, both personally and professionally. I am very fortunate to have encountered them in my journey.

Also, thanks to Thomas G. Bever, Robert Berwick, Mª Victoria Escandell Vidal, Dan Faltynek, Daniel Fryer (again), Petra Martinkova, Vladimir Matlach, Phoevos Panagiotidis, Thomas Stroik, Whitney Tabor for their (direct or indirect) contribution to the development of some of the ideas presented here and in past works. Mine only is the responsibility for any mistakes.

Susan Schmerling, a very dear friend and brilliant colleague, has painstakingly read, edited, and commented on the full manuscript not once, but *twice*! The reader should thank her for whatever degree of intelligibility this thesis has.

I am particularly grateful for having had the chance to have Doug Saddy as a mentor and overall guide. It has been an immense privilege and honor (not to mention a lot of fun) to be his apprentice, learn from him, and work under his supervision. Thank you, Doug.

Part I: Natural Languages and Natural Grammars

1.1 What is a grammar, and why do we need them?

In the present work we will be primarily concerned with the following twofold question:

1) *How do we make sense of external stimuli in the form of objects and relations? And, how do we externalize the internal representations of the phenomenological world?*

The first part of (1) is not a problem concerning sensory perception, as it might seem: every stimulus we are exposed to requires some sort of abstraction in order for us to impose structure on it. Let us consider a couple of very brief but (hopefully) illustrative examples:

a) It would be at the very least simplistic to claim that there is nothing more to vision than light exciting rods and cones in the retina, and so on… Imagine a desk, with a computer on it, a briefcase, and Post-it notes all over the place. Light from the Sun or an artificial source reaches those objects, some frequencies being absorbed, some being reflected and reaching our eyes. That would be pure perception. However, we know there are multiple objects in our field of vision, and what is more, we establish relations between them (of the kind X *on / under / in / ...* Y; relative distances with respect to us as well as other objects…). A crucial point, frequently overlooked, is that photons do not come with diacritics of the kind 'I am light reflected from the computer', or 'I am light reflected from the desk'. We get a single array of light. Distinctness and relations are not part of the stimuli, but we somehow *assign* them to the visual stimuli.

b) The linguistic example has been worked out a bit more, since at least Aristotle (*Poetics*) and Varro (*De Lingua Latina*). More recently, approaches within Mainstream Generative Grammar[1] (MGG henceforth) have developed the argument from an internalist, innatist perspective. The problem is the same: natural language, at the sentential and / or discoursal levels (see Everett, 2005 for a relatively recent reference in this respect), displays hierarchical relations of different kinds. But all we get are waves, layers of air in movement that hit our eardrum (and so on…). Quite safely, we can assume that formal hierarchy (phrasal, discoursal) is not part of the waves, that is, it is not coded as part of the layers of moving air. How do we work out constituency, reference (including binding phenomena), presuppositions, entailments, gap-filling

[1] Following Culicover and Jackendoff (2005: 3), we will use the term MGG all throughout the present work '*to refer to the line of research most closely associated with Noam Chomsky, including Syntactic Structures (1957), the Standard Theory (Aspects of the Theory of Syntax, 1965), the Extended Standard Theory (Studies on Semantics in Generative Grammar, 1972), the Revised Extended Standard Theory (Reflections on Language, 1975), Principles and Parameters Theory (Lectures on Government and Binding, 1981), and the Minimalist Program (1993; 1995)*', as well as those extensions and patches by authors closely related to Noam Chomsky's theoretical position, and whom Chomsky or collaborators have recognized as part of their enterprise. We will extend the concept of MGG to include the so-called 'Biolinguistic enterprise' (e.g., Di Sciullo and Boeckx, 2011; Fujita and Boeckx, 2016) and other attempts to justify core concepts of the Chomskyan theory (Universal Grammar, the existence of an innate and specific Faculty of Language, 'syntax' as an autonomous generative procedure…) from any discipline (e.g., philosophy, see McGilvray, 2013).

(including sluicing, ellipsis…)… out of a phenomenological stimulus that displays no syntactic structure of its own[2]?

It should be at least apparent by now that we do not consider structure is *in* the stimulus. Our answer to the first part of (1) is, thus:

2) *By assigning a grammar to those stimuli*

This is, admittedly, a bit vague, but it will be refined shortly -we still need to define what a grammar is, for example-. Moreover, we will make an ancillary assumption in (3) (whose spirit is in the line of Simpler Syntax, see Culicover and Jackendoff, 2013: 2, but under dynamical local conditions which will be made explicit below):

3) *The grammar we assign to a stimulus is the simplest possible grammar compatible with the input class locally while losing as little information as possible in the process*

(3) introduces the requirement of 'losing as little information as possible', which implies some sort of counter-entropic measure built into the system (and which will be essential in answering the second part of (1), the problem of structure externalization). Moreover, we allow for the possibility that the relevant grammar varies in different local domains: that is, a grammar G might be the 'simplest' for a domain D and might be enough to capture the relations between elements in D, but might not be enough to provide a *strongly adequate* structural description for D', that is, assigning a structural description to D with neither too rich nor too impoverished structure. Unlike Chomsky and Miller (1963) and Ristad and Berwick (1989), *inter alios*, we do *not* assume that the assignment process is in charge of a *deterministic function-based* parser, and we make no *competence-performance* distinction either[3]. It should begin to be apparent that one of the cornerstones of the present thesis is that we are dealing with an interaction-based *dynamical nonlinear system*[4] (a claim we will carefully argue for in **Part II**) which displays some sort of cyclic *entropy – counter-entropy* dynamics, to be refined and expanded on below. Still, (3) is not anywhere near 'satisfactory'. Let us proceed to carefully define and refine the concepts we will use.

In the context of this work (and more generally, of our theory), we define 'grammar' as follows:

---

[2] Needless to say, there *is* acoustic structure, but it does not map to the *syntactic* processes we presumably have in the input preserving selected structure. Thanks to Susan Schmerling for making us aware of the need to include this footnote.

[3] We will define 'function' in **Section 2.4.1** and discuss the implications of adopting a strict notion of 'function' for theories of language and cognition. A working definition, which we will discuss below, is the following: '*A function is a deterministic relation between a set of inputs and a set of permissible outputs with the property that each input is related to exactly one output.*'

[4] We don't want to give away much now (otherwise there's no fun!), but this point deserves some clarification. A dynamical nonlinear system:

a) Changes in time ('dynamical')
b) Produces outputs which are not directly proportional or linearly related to its inputs ('nonlinear')
c) Is composed by sub-units / sub-components which interact to produce said outputs ('system')

> *A finite, maximally explicit set of structural descriptions which are dynamically and locally assigned to stimuli of whichever nature.*

In this context, we take the concepts 'grammar' and 'generative grammar' to be equivalent, if 'generative' is understood in the sense of Chomsky (1965: 4; 1995: 162 fn. 1), as 'maximally explicit' (i.e., mathematically formalized and therefore unambiguous); but not in the sense of 'proof-theoretical derivation of structural descriptions', which is a sense that Chomsky has frequently rejected (see McCawley, 1988: 355-356 for an example of such an interpretation of 'generation' as 'specifying the membership of a set of sentences that is identified with a language', this specification being a recursive function; see also Chomsky, 2007: 6) but also frequently assumed himself[5]. It is to be noted that we shall use the terms 'generate' and 'generation' in a different sense, focusing on real-time computational processes (*contra* Chomsky, 2007: 6), however, as we will see shortly, this does not mean that we assume either an autonomous 'syntactic component' or a 'constructional' approach to what 'syntax' is; as postulated by Schmerling (in press) –in turn a recent rephrasing and reworking of classic European and American structuralist proposals-, 'syntax' *does not create anything that is not 'already there'* in a sense that will be made explicit in **Part III**. Therefore, when referring to a maximally explicit set of structural descriptions, we will simply use the term 'grammar', presupposing the historical / methodological caveat[6].

We depart from most linguistic work here (structuralist, generative, and even some branches of cognitive linguistics), which assumes that a 'grammar' is strictly related to natural language: our approach will be more related to computer science and automata theory when defining a language and its grammar. Thus, for instance, while some second language acquisition researchers argue strongly against the concept of 'incomplete grammar' (often without providing a formal definition of 'grammar'), we claim that *if a grammar is consistent, it must be incomplete*, by Gödel's Incompleteness Theorems, for a grammar in the context of the present work is a formal system like arithmetic or set theory. We put particular emphasis on minimizing the amount of axioms and making inference rules more powerful; a position that is rather close to Generative Semantics as in McCawley, 1968, 1971; Lakoff, 1971, among others. This perspective shift will prove useful as our argument unfolds, and the importance of so-called 'artificial grammars' becomes obvious. Moreover, and unless explicitly indicated, we will not make in **Part I** a systematic distinction between a biological person and a formal automaton.

Let us spell our basic assumption out: the process of 'grammar assignment', to which we will return below, is the set of mechanisms by means of which an input is parsed (note: this is not to say that the grammar is the parser, rather, the grammar can be implemented, and that implementation constitutes parsing). By 'parsing' we mean here that an input is assigned a structural description *and* an interpretation at the relevant system, depending on the information

---

[5] For example: '*The base of the syntactic component is a* ***system of rules*** *that* ***generate*** [sic] ***a highly restricted (perhaps finite) set of basic strings****, each with an associated structural description called a base Phrase-marker.*' (1965: 17. Our highlighting); or '*In its most elementary form, a* ***generative system*** *is* ***based on an operation that takes structures already formed and combines them into a new structure****. Call it Merge.*' (2007: 3. Our highlighting).

[6] An interesting question for studies on the history of Linguistics, and perhaps even its philosophy, is whether generative grammars are indeed 'generative' in the technical sense allegedly intended by Chomsky, particularly under the so-called Minimalist Program (Chomsky, 1995 and much subsequent and related work) and the concomitant considerations about 'perfection' in the Faculty of Language, the role of so-called 'uninterpretable features' in derivations, and so on.

contained in the input (visual, auditory, etc.) and the interactions between those systems. We will use 'parsing' and 'interpretation' interchangeably. In the present work, the characteristics of the systems involved and the workspaces in which operations apply will be analyzed as well, since we will see there are constraints over possible structures that derive from:

- Physical properties of the neurocognitive substratum and the computational system, and
- Topological properties of the mental spaces in which elements are manipulated.

In other words, the characteristics of the space where formal operations apply (including structure building, mapping, and interpretation) constrain the possible outputs of those operations and determine emergent properties of cognitive computations (Saddy, 2018). The kind of emergent properties we get given human limitations will be one of the main objects of inquiry in this work.

At this point, it is useful to compare the definition of grammar we will assume here with some of those more common in computer science or mathematical logics. For instance, in the context of a discussion of formal languages and their corresponding automata, Hopcroft and Ullman (1969) define a grammar as a set (V_N, V_T, P, S), where

> *The symbols V_N, V_T, P, and S are, respectively, the variables, terminals, productions, and start symbol. V_N, V_T, and P are finite sets*. (Hopcroft and Ullman, 1969: 10)

Quite close in time, and influenced by the same mathematical background, Chomsky (1965: 31) characterized a grammar within the discussion of the goals of linguistic theory as follows:

> *We must require of such a linguistic theory* [a generative grammar] *that it provide for:*
>
> *(i) an enumeration of the class $S_{1'}$ $S_{2'}$, … of possible sentences*
> *(ii) an enumeration of the class SD_1, SD_2, … of possible structural descriptions*
> *(iii) an enumeration of the class G_1, G_2, … of possible generative grammars*
> *(iv) specification of a function f such that $SD_{f(i, j)}$ is the structural description assigned to sentence S_i, by grammar G_j, for arbitrary i,j*
> *(v) specification of a function m such that m(z) is an integer associated with the grammar G, as its value (with, let us say, lower value indicated by higher number)*

We could provide more definitions, but we consider these as fine representatives of their time, still very much dependent on the works of Turing (1936) and Post (1944). Quite unsusprisingly for anyone familiar with the history of Generative Linguistics, both definitions have a lot in common. The Base component of a transformational grammar (lexicon + phrase structure rules) manipulate non-terminal and terminal symbols; and structural descriptions for transformations make reference to variables which range over unbounded sequences of both terminal and non-terminal symbols as well (Ross, 1967). They are still present in the form of a set of (well-formed) sentences, {S_1, S_2, …, S_n}, particularly because the root node S is, in Hopcroft and Ullman's conception, a '*distinguished nonterminal*'. Interestingly, the *production* relation, which in rewriting formalisms is denoted by an arrow (such that $\Sigma \rightarrow F$ is a *production* function over an alphabet Σ that produces –possibly unary- strings F as outputs, more on this below), is present in early Generative Grammar in the form of Phrase Structure Rules (PSR) within the *base* component (which also includes the Lexicon, an equivalent to the *alphabet* in purely computational terms), but also in point (iv) in Chomsky's (1965) system. Requirement (iv) assumes that there is a function relating strings to structural descriptions (which has to be specified), such that there is a bijective relation between *kernel sentences*[7] and *structural descriptions*[8]. A grammar G, in the *Aspects* theory, is a set of rules that assign a structural description to each and every well-formed formula in a natural language NL via a function *f*; this function, insofar as it recursively enumerates the sentences of the language, is *generative* (see Post, 1944: 286). Just like in Chomsky's version of the Standard Theory, a language is the set of strings (Hopcroft and Ullman call them *words*, a term that is equivalent to Chomsky's 1965: 18 *basic strings*) generated by the grammar via the projection function applied to the initial symbol *S* (Hopcroft and Ullman, 1969: 11)[9]. Furthermore, the finite character of the V_N and V_T sets is also shared by early Generative Grammar, for lexicalist stances claim that the Lexicon is neither computational nor systematic (see Chomsky, 1970; Fodor, 1970; Uriagereka, 1998: Chapter 6, *contra* Generative Semantics, see e.g., Lakoff and Ross 1967; McCawley, 1968; Shibatani, 1976, among others).

There are two salient notions in both Chomsky's and Hopcroft and Ullman's definitions (also present in other frameworks, particularly Lexical Functional Grammar, see Kaplan and Bresnan, 1982): *functions* defining relations between objects and/or levels of representation, and *finiteness*. In this work, we will be primarily focused on the former, only tangentially touching on the latter (which is nevertheless crucial, see Langendoen & Postal, 1984; Langendoen, 2010 for some discussion). The reason is that assuming a 'generative device' based on number-theoretic functions which recursively enumerates sets (and sentences, as sets of words and phrases) has, as we will see, far-reaching consequences not only for the design of the (linguistic) theory and what we can expect from it (see Pullum & Scholz, 2001; Pullum, 2007), but also for the conceptions of the computational and neurocognitive substratum in which that theory is implemented. Generative linguistics, and its psycho- and neuro-linguistic branches have

---

[7] Strictly speaking, the term was first used in its modern sense by Zellig Harris (1951), but Chomsky reformulated the notion in his *Logical Structure of Linguistic Theory* (1955). It is Chomsky's version we use here. *Kernel sentences* are strings that have a single base Phrase-Marker as basis and only undergone obligatory transformations –e.g., affix hopping-, see Chomsky, 1957: 45; 1965: 17-18. The term was later abandoned in favor of Deep Structure, which was also related to semantic interpretation and soon developed into a so-called 'level of representation' in transformational theories

[8] In a transformational theory, structural changes and the descriptions of the phrase markers after transformations belong to a different component from phrase structure (the so-called *transformational component*, comprising a set of rules to map phrase markers onto other phrase markers via displacement operations, including passivisation, affix hopping, Wh-movement, among others). However, since not every generative grammar is a transformational grammar (take Pollard and Sag's HPSG and Kaplan and Bresnan's LFG as examples), we exclude a thorough discussion of the transformational component from the present work.

[9] Chomsky (1957: 13) defines the *grammar* of a language L as '*a device that generates all of the grammatical sequences of L and none of the ungrammatical ones*' (see also Chomsky, 1965: 9), without clarifying the grammaticality criterion (the linguistic equivalent to the *decidability problem* –see Hilbert and Ackermann, 1928-, to some extent. See Kornai, 1985, for some relevant considerations about grammaticality judgments for center embedding) or the exact nature of the 'device'. The 'generation' process is taken there to be a [Σ, F] *sequential* grammar, a property that crucially distinguishes Chomsky-like rewriting rule systems from L-grammars. Prusinkiewicz and Lindenmayer (1990: 3) explicitly claims that the replacement of the initial sequence with final strings occurs in parallel and simultaneously, as cell division, for instance, occurs in more than a single cell at any time T. Parallel computation will be a crucial part of our proposal, as in previous works, see Krivochen (2011, 2012, 2015a, b). We will discuss here how the simultaneity of L-grammars makes them orthogonal to the Chomsky Hierarchy, which we claim only applies to sequential grammars and the automata that implement them. See also Saddy & Krivochen (2016b); Saddy (2018) for a similar perspective.

assumed more or less strong versions of the Computational Theory of the Mind (CTM henceforth, the idea that the mind *literally* is a digital computer; see Horst, 2015 for a dispassionate overview; Gallistel and King, 2010 for a more apologetic approach) for the last 50 years, and have equated *effective computation* to *function-based computation*, with deep consequences not only at the theoretical level (e.g., the abstract design of neural networks, or the hypotheses about the kind of formal language that can better model a neuron's behavior), but also at the empirical and experimental levels, impacting on experiment design as well as data interpretation.

It is one of the main purposes of the present work to problematize these assumptions, question the theoretical and empirical validity of a uniformly function-based CTM, and propose an alternative model that maintains (and hopefully enhances) descriptive adequacy while improving on theoretical aspects, from the abstract notion of *computation* to the physical properties of the automata we assume the formal system is to be implemented in. Thus, after characterizing what a 'grammar' is, and partially defining its computational properties, we will critically discuss the notion of function-based computation and its ramifications, as well as recent alternative frameworks, in order to find the niche from which to develop our own logically consistent alternative.

Before proceeding further, we need to specify the properties that a grammar must have *qua* formal axiomatic system: in this way we restrict the framework and put our cards on the table. This also allows the reader to draw comparisons with other definitions and characterizations available in the literature (some of which will be reviewed below). In the background of formal language theory, a 'grammar' is a finite set of rules of the general form A → B over an alphabet Σ of terminal and non-terminal symbols, and has the following properties:

- <u>Consistency:</u> As with any formal axiomatic system, we want the set of structural descriptions and the rules generating those not to contain internal contradictions. The set of structural descriptions must thus be finite, and the metatheory of the grammar must contain conditions over what can be a rule of the grammar. This requirement, however, does *not* mean that a single structural description cannot be used for more than a single stimulus, since such re-use is the principle of *generalization* in learning, nor does it mean that a single stimulus cannot be assigned different structural descriptions by different individuals, depending on situational factors (insofar as the situational context is represented in the mind). Consistency is a requirement over the *formal model*, not the modelled object.

- <u>Incompleteness:</u> This point attempts to cover Gödel's objections to Hilbert's axiomatic model for arithmetic (Gödel, 1931). Basically, we claim that a *consistent* grammar is always *incomplete* (the so-called '1st incompleteness theorem') –that is, every grammar has its 'Gödel sentence'; a statement that is true in L but cannot be proved in L- and, moreover, that such a grammar *cannot* prove its own consistency (the so-called '2nd incompleteness theorem') –or, rather, the system cannot contain a proposition asserting its own consistency-[10]. That is, there is *at least* one structural description which does not belong in the grammar, and, moreover, there is no way to incorporate it without making

[10] These are, of course, simplifications of the original mathematical-logical formulations. We refer the reader to Gödel (1931), Theorems VI and XI for the source material.

the system inconsistent. A grammar, as any other formal axiomatic system, cannot provide a solution to the decidability problem for grammaticality unless strings are computationally *uniform* and their computational complexity falls below (or equal to) *context-sensitivity* in the Chomsky Hierarchy[11] (this problem is known as the *Entscheidungsproblem*, see Hilbert and Ackermann, 1928: Chapter III, § 12 for the first reference to the problem; Turing, 1936 is the *locus classicus* for its non-solvability, also Church, 1936). We will make a case for the divorce between the properties of the *object* and the properties of the *formalism*, such that the object may not be uniformly Turing-computable (derivationally and / or representationally), but the formalism (the linguistic metatheory) may very well be.

- Recursion: There are many definitions of this notion which in part differ in their object of focus (such that *recursion* is defined *for an object* rather than for an *operation* or set thereof), but at least those used in mathematical logics and computer science are formally equivalent (which is not the case if we take into account the definitions used in linguistics, often misquotations from mathematical logics papers, as is the case of Watumull et al., 2014; often just taken for granted with no explicit formulation or mathematical discussion, as the case of Hauser, Chomsky, and Fitch, 2002; see Lobina, 2014 for healthy clarifications[12]). Consider the following alternative criterial definitions:

    - Recursion as a property of *systems*: A *formal system* is recursive if one can generate theorems from a limited number of axioms (e.g., the Zermelo-Fraenkel set theory is *recursive* in this sense)

    - Recursion as a property of *languages*: '*A language whose sentences can be generated by a procedure* [a function or set thereof] *is said to be recursively enumerable*.' (Hopcroft and Ullman, 1969: 6)
      A *language* (a *set of strings*) is *recursive* (but not necessarily *recursively enumerable*) if there exists a Turing Machine that accepts every string of the language and rejects every string (over the same alphabet) that is not in the language. A language is *recursively enumerable* if there exists a TM that

[11] In Church's (1936: 41) terms, "*By the Entscheidungsproblem of a system of symbolic logic is here understood the problem to find an effective method by which, given any expression Q in the notation of the system, it can be determined whether or not Q is provable in the system*". Hilbert and Ackermann (1928) formulate it as follows: "*The Entscheidungsproblem is solved when one knows a procedure by which one can decide in a finite number of operations whether a given logical expression is generally valid or is satisfiable. The solution of the Entscheidungsproblem is of fundamental importance for the theory of all fields, the theorems of which are at all capable of logical development from finitely many axioms*." [Trans. by Klaus Sutner]

[12] See also Everett (2009), who presents a brief but illustrative summary of the notions of 'recursion' assumed in the *linguistic* literature (Everett, 2009: 1):

> *There are different formalizations of recursion around, but the two that seem most appropriate to considerations of its relevance to human language are (1) and (2):*
> *(1) Recursion A: Recursion is an operation that applies to its own output.*
> *(2) Recursion B: For any grammar recursion is the property that in principle a machine could determine in finite time, for any arbitrary finite string over the right alphabet, whether the string is in the language or not.*

accepts every string of the language and does not accept every string that is not in the language (but not necessarily *reject* it, the TM can also enter an infinite loop).

- Recursion as a property of *functions*: '[a] *number theoretic function* φ *is said to be recursive if there is a finite sequence f number-theoretic functions* φ_1, φ_2 , ..., φ_n *that ends with* φ *and has the property* that *every function* φ_n *of the sequence is recursively defined in terms of* [...] *preceding functions, or* [...] *is the successor function* $x + 1$' (Gödel, 1931: 159)

- *A subset S of the natural numbers is called recursive if there exists a total computable function f such that* $f(x) = 1$ *if* $x \in S$ *and* $f(x) = 0$ *if* $x \notin S$ (in this sense, *recursive* is synonymous with *decidable*; based on van Heuveln, 2015; Berwick, 1984: 190)

- A function is recursive if it allows the non-circular definition (by means of proof) of formulae in which the function occurs (based on McCarthy, 1960: 4)

Of course, it must be borne in mind that the fact that a grammar as a *formal system* is recursive in any of the senses specified above does not mean the *object it models* is recursive as well: that is to be proven empirically, and independently. In other words, whereas the *structural description* for a string, or the process by means of which that structural description is obtained are formally recursive, it might very well be the case that the *input* for which the grammar provides a structural description is not. A point we will make here is that *natural languages are not recursive* in any relevant sense, but the *formal procedures to generate structural descriptions* might be, if they turned out to be functions in some nontrivial sense. The outputs of natural language grammars are the objects that, *when interpreted*, are assigned a structural description by means of a process that might or might not be *recursive*. Of course, this strongly contrasts with Hauser, Chomsky and Fitch's (2002) claim that *recursion* (in a very vague sense that has provoked severe misunderstandings within the field; Lobina, 2017 offers detailed discussion and clarification still within the 'biolinguistic' camp) is the crucial characteristic of the Faculty of Language in the Narrow sense (FLN): we are highly skeptical regarding not only the existence of an 'innate' Faculty of Language (a most obscure notion, since definitions have changed over the years becoming more vague and including biological jargon to further complicate the picture; see Behme, 2015b; Postal, 2009, *inter alios* for a critical review of the rhetoric behind UG and the lack of hard evidence for its existence), but also the claim that recursion (caveats aside) is a necessary condition for natural language to have arisen evolutionarily and to 'work' as a computational (read: 'generative' –Post's 1944: 286 *recursively enumerable*- and 'function-based') engine. Consider, for instance, tessellation (i.e., the tiling of an *n*-dimensional space without leaving any gap or empty place –think of a tiled floor, for instance. See **Figure 1**, below-). While the *procedure* of dividing a plane using a finite number of geometrical figures (e.g., two, in the case of the well-known Penrose tiles 'kite' and 'dart' but usually more if the tiling has a decorative function) *is* recursive in the senses specified above, it is not clear in which sense we can say that the *object* (i.e., the tessellated plane) is recursive itself.

Figure 1: tessellated wall at the Alahambra, Spain. "Tassellatura alhambra" by gruban - http://www.flickr.com/photos/gruban/11341048/ Licensed under CC BY-SA 2.0 via Commons

The same caveats can be made with respect to a sentence: whereas the question about whether it has been *derived* by means of recursive procedures like Post-style rewriting rules is, although not obvious, legitimate (see Everett, 2005 for some critical discussion about the role of recursion in language; also Jackendoff and Wittenberg, 2014[13]). However, we do not think that asking 'is a *sentence* recursive?' makes any sense: on the one hand, just like nobody would say a plane is recursive before tessellating it; a sentence, existing as either a neurocognitive entity describable in terms of patterns of firing neurons, or as an external object describable as drops of ink on a paper, or layers of air in movement, is hardly describable as *recursive* before 'someone' (or 'something', we are not distinguishing people from automata for the time being) attempts to 'parse' or otherwise analyze (provide a structural description and an interpretation model for) it. On the other hand, but in a related point, it is an empirical question whether the procedures used to weakly generate the sentence are recursive themselves. In the sense that *generate* has in Post (1944), this is only true if this procedure recursively enumerates the sentences in a language; this view assumes that explicitation can only come with enumeration. But if we analyze the nature of structural descriptions that are assigned to strings, the question of 'what operations are necessary to yield a string *s* as output' becomes relevant, and essentially an empirical question. In Krivochen (2015a; 2016b, c; 2018a); Bravo et al. (2015), a.o. we argued that phrase

[13] It should be noted that these authors make the mistake, in our opinion, of equating *syntax* to *recursion*, thus tacitly accepting Hauser, Chomsky and Fitch's (2002) definition of FLN. They say '*In our view, syntax is not the only generative linguistic system*' (Jackendoff and Wittenberg, 2014: 66), expanding on other 'generative' systems in natural language (see also Jackendoff, 2002), but without contesting the 'generative' character of the 'syntactic component'; this implies as narrow a conception of what 'syntax' is as MGG's: '*it* [syntax] *is a formal system that abstracts away from semantic categories, that labels constituents in terms of abstract categories such as nouns and verbs, and that imposes formal structure on them*.' (Jackendoff and Wittenberg, 2014: 67).

markers exhibit computationally mixed dependencies, which means that a single phrase marker could display a finite-state dependency between some of its parts, and a higher-level degree of complexity between some others[14]. In past works we proposed that it is the change of computational dependency that delimits local cycles in such a 'mixed' system. By saying that a system is 'computationally mixed', we mean that the structural descriptions assigned to substrings in L(G), for L a finitely enumerable set of strings and G a grammar, *need not all be formally identical*. In this sense, a computationally mixed system assigns a substring *the simplest structural description that captures and represents the formal and semantic relations between syntactic objects*, essentially deriving the so-called *cyclic principle*: instead of appealing to designated nonterminals (e.g., NP / S), the assignment of structural descriptions and the formulation of admissibility conditions over representations pays attention to the shift in computational dependencies within 'extremely local' units (in the sense of Putnam, 2010: objects introduced in the derivational space by the immediately following rule). Should we concede that the *derivation* of a sentence is an example of a recursive procedure, we are faced with the question whether *all* the operations involved in any such derivation can be characterized as 'recursive' and exactly how so: this involves a problem noticed, among others, by Juan Uriagereka, about the syntactic / semantic representation we assign to a phonological string like (4), for X representing different occurrences of a given symbol:

4) […X…X…]

There are at least three ways in which we could assign (4) a structure (omitting the dots, which correspond to irrelevant structure):

5) a. [[X] [X]]
b. [X [X]]
c. [[X] X]]

In Kosta & Krivochen (2014a: 249) we offered conceptual and empirical evidence in favor of not considering (b) and (c) mirror images of each other (as has been suggested in the literature), but basically the same phrase marker, since branching side should be irrelevant for a neurocognitive plausible model of syntax. Consequently, a situation like (4) can be solved in two ways:

6) a. Assume a linear (e.g., finite-state) dependency between instances of X
b. Assume a hierarchical (e.g., phrase-structural) dependency between instances of X

Since the number of possible structural descriptions for *n* instances of X is *n*!, should we have […X…X…X…] (a longer version of (4)), there are 6 possibilities, of which only 3 are semantically relevant. The others are mirror images of these 3, only relevant if branching side is taken to be a significant factor (the picture gets more complex with each further occurrence of X). This question, in turn, involves assumptions about the

[14] Sampson (2009) and Gil (2009) also challenge the often implicit assumption that natural languages are structurally uniform, but their view is essentially cross-linguistic. In that respect, their work could be seen as complementary to some aspect of the present thesis. They also put more emphasis on language as part of human culture than on its cognitive, physical, and formal aspects; without denying the importance of the latter.

topological nature of the space in which derivations take place, to which we will return in depth in **Section 3.2** below:

7) a. [[X] [X] [X]]
b. [X [X [X]]]
c. [[X X] X]

Of these, (b) is the only one that displays pure, uniform phrase-structural (Type 2) dependencies, being derivable by successive and constantly growing ('monotonic') applications of a *concatenation* operation (say, 'Merge' or 'Unification'; at this point they are not empirically distinguishable) to instances of X. The others involve either pure finite-state dependencies within a *flat* structure (in (a)), or mixed dependencies – combining parataxis with hypotaxis- (in (c)). The crucial part here is that no interpretative hypothesis is any better than the others *a priori*, without us knowing the semantic contribution of each instance of X (note that this takes us back to the possibility of having to resort to *different grammars* for different local domains that we left open in (3), in a sense that will be made fully explicit in the remainder of the thesis). Thus, the options in (5) are all available as interpretative hypotheses when we start to parse the input (which therefore has no inherent structure), the final decision being a product of the interaction between independent systems (a claim we will explore in depth below).

- <u>Productivity:</u> (also often referred to as 'creativity' in the Cartesian sense, see Behme, 2014 for discussion) We claim that the set of structural descriptions is not fixed *a priori*: unlike Chomsky (1965: 31), we do not consider that a linguistic theory should contain an enumeration of all possible sentences: such a set has been proven to be *uncountably infinite* (Lewis & Papadimitriou, 1998; Langendoen & Postal, 1984). This is so because there is a potentially unlimited number of sentences (Chomsky, 1965: 4), and it is also a popular view within MGG (particularly on its first incarnations) that any sentence can be infinite in length due to the possibility of embedding (this take on 'infinity' has some concomitant learnability problems that we will not tackle in this work, but which should be taken into consideration when evaluating a theory of grammar). The set of structural descriptions, on the other hand, is *finite*. In the view defended here, we propose that the process of *grammar assignment* is dynamic and adaptive, in the sense that a particular interpretative hypothesis can be replaced in mid-parsing if a disambiguating / correcting element is introduced (the same happens in cases of semantic / pragmatic *coercion*, see Escandell and Leonetti, 2002). Consider, for example, the famous case of 'Garden Path Sentences' (example taken from Bever, 1970: 316):

8) The horse raced past the barn fell[15]

[15] Bever qualifies the sentence with ?, as marginal or unacceptable. However, if presented to subjects in a non-self-pace-reading manner (e.g., if the sentence is heard in casual speech), its acceptability increases considerably. The medium (in the sense of 'properties of the physical channel through which the message is transmitted', following Lyons, 1977: 103, ff.) seems to be at least partially relevant in the identification of parsing difficulties.

It is known that, until ‘fell’ is presented to subjects, the preferred interpretation is NP + VP (see Bever’s ‘interpretative strategies’). However, after ‘fell’ is introduced, the representation must change to NP + Relative Clause + VP. Graphically:

a) [NP VP] (before the introduction of ‘fell’)
b) [[NP [RC]] VP] (after the introduction of ‘fell’)

We will see that the dynamic, online character of the model we argue in favor of has empirical differences with respect to predictions derived from models of function-based computation, which is basically linear, sequential, and insensitive to external influence.

The effects where (8) is concerned are clearer when the sentence is linearly expanded, as in (9):

9) The horse raced past the barn fell after tripping over.

From a theoretical point of view, the computation of functions cannot be tampered with: once one set the (α-)machinery in motion (i.e., the relevant Turing Machine TM), one external controller cannot influence it (e.g., change the rules, favor the application of one rule over another, etc.). This means that interpretative procedures, either considered as a whole or as sets of trial-error loops, are impenetrable (just like traditional Fodorian vertical faculties and their way of processing data, see Fodor, 1983). The direct consequence of this is that we cannot know whether the system has chosen a ‘bad’ interpretative path until we have the results. If the interpretation is not compatible with some measure (say, *accuracy* or *simplicity*, to take a classical opposition in the psychology of perception, see Pomerantz and Kubovy, 1989), then the process must be repeated *from square one*. An interactive approach makes quite different claims from those made by a function-based approach: as we will see in **Section 2.4** when analyzing function-based vs. interaction-based computation, the latter allow multiple sources of information to influence the computation at non-aprioristically defined points, rather depending on the requirements of the system when processing a particular input and the kind of available information. Sub-derivations delimiting accessible domains for the purposes of specific operations (so-called *cycles*) are thus porous, open to external influence (thus resembling *c*-machines more than α-machines, in Turing’s 1936 terms).

We will look at this in more detail when comparing *function-based* and *interaction-based* models of computation.

- <u>Internalism:</u> This is one of the crucial parts of our proposal. As a radical departure from (often tacit, but) widely held assumptions in cognitive science, we will argue that the phenomenological world *has no structure itself*, but *we assign* structural descriptions to the phenomenological world by means of categories like *sortal* and *eventive* entities, where the latter can be defined in terms of relations among the former. We defend the idea that interpretation *is* grammar *assignment* (as opposed to *recognition*), in the sense in which we defined it above. This claim has relevant consequences for cognitive science in general, but we will focus on the linguistic implications of adopting such a view. The immediate consequence is that no linguistic object *has* a structure, that is, there is no structural description associated with a string before that string is interpreted. Moreover, in this *structure-assignment* process, more than a single structural description can be initially considered as a working hypothesis. This multiplicity of possible structures gives rise to a *competition* dynamics among them that, relevantly, need not result in the same candidate for all subjects, so-called 'structural ambiguities' illustrate this eloquently. Consider the following example, appeared in a 1930 issue of the magazine *Boys' Life*:

10) *Scoutmaster*: Time flies.
Smart Tenderfoot: You can't. They go too fast

Relevantly, the 'scoutmaster' has assigned the string the structure [$_{\text{Subj}}$ Time [$_{\text{Verb}}$ flies]], whereas the 'smart tenderfoot' acts as if he takes it to be an imperative sentence with the structure [[$_{\text{Subject}}$ *you* [$_{\text{Verb}}$ time [$_{\text{Obj}}$ flies]]]. Considering humorous and situationally–dependent interpretations (which take us, computationally, significantly beyond a 'last-element-in-the-stack' based kind of automaton, since non-literal –non-natural, in Gricean terms, see Grice, 1957- meaning often requires near-unrestricted search space, perhaps only constrained by procedural elements).

A more complex example, like (11)

11) Time flies like an arrow

has (at least) the following possible interpretations, with different degrees of 'far-fetchedness' (taken from http://en.wikipedia.org/wiki/Time_flies_like_an_arrow;_fruit_flies_like_a_banana)

12) a. (*imperative*) measure the speed of flying insects like you would measure that of an arrow
b. (*imperative*) measure the speed of flying insects like an arrow would
c. (*imperative*) very quickly (i.e., like an arrow) measure the speed of flying insects
d. (*imperative*) measure the speed of flying insects that are like arrows (includes an abridged restrictive relative clause)
e. (*declarative*) all of a type of flying insect, "time-flies" collectively enjoy a single arrow (i.e., $\forall$(flies) > $\exists$(arrow))
f. (*declarative*) each of a type of flying insect, "time-flies" individually enjoys a different arrow (i.e., $\exists$(flies) > $\forall$(arrow))
g. (*declarative*) each of a type of flying insect, "time-flies" individually enjoys an occasional arrow (i.e., $\exists$(flies) > $\exists$(arrow))
h. (*declarative*) "time", as a sortal entity, moves metaphorically in a way an arrow would
i. (*declarative*) a copy of the magazine *Time*, when thrown, moves in a similar manner to that of an arrow (admittedly far-fetched, but possible)

Those different interpretations, some varying in Logical Form properties only (e.g., e / f / g), some in constituency boundaries (e.g., a / b / c / d), and some in interface properties (relating both semantics-pragmatics and syntactic structure; e.g., h / i), seem to provide at least a starting point to think of the structural description associated to the string [time flies like an arrow] as something *not inherent* to the string. If this is so, then our task, as interpreters, is not to *uncover* the real structure, but to *assign* the string a structure that fulfils certain requirements, which we will deal with in detail below. The internalist proposal we put forth here also has another consequence, and that is that *there is no* *structural description if there is no mind to assign it*[16]. Of course, we are not restricting the possibility to assign a structural description to humans, as it seems obvious that animals also assign a structural representation to their environments, their own actions, and even their social relations[17], even though they do not externalize it in the form of stings of symbols and it is likely the kind of computational dependencies available to them is more restricted than in the case of the *Homo* genus. Specific properties, we will argue, arise as *emergent* properties of the interaction between complex systems.

- Generative character: Chomsky (1965: 4, 1995: 162) has asserted that a *generative grammar* is nothing more than an *explicit* grammar (Borsley and Börjars, 2011: 1-2; but see McCawley, 1988 and Postal, 2003: 603, who explicitly show that Chomsky's definitions of what 'generative' is have changed over the years, yielding some inconsistencies. The reader may compare Chomsky, 1995: 162 with Chomsky, 1957: 13; Chomsky, 1959: 138; or Chomsky, 2007: 3), abstracting away from the real-time process of structure generation and processing in the mind. In fact, he claims that '*a generative system involves no temporal dimension. In this respect, generation of expressions is similar to other recursive processes such as construction of formal proofs*' (Chomsky, 2007: 6). It seems clear that no such 'generative' system (in the 2007 version and similar incarnations, a system whose origins are to be found in Post's 1944 definition of *generative* in the context of sets of positive integers) can account for physically realized –i.e., embodied- cognitive processes (and this claim will be essential in the adoption of a certain kind of time-dependent equations for the mathematical formalization of our theory while acknowledging the essentially *continuous* and *analogic* character of mental computation, see Spivey, 2007 for discussion), and, in fact, it seems to be at odds with the alleged 'biolinguistically' oriented enterprise that currently dominates a large part of the generative market (see Di Sciullo and Boeckx, 2011; and Fujita and Boeckx, 2016 for a recent overview), in the sense that all biological *systems* evolve in time, most frequently in the form of non-linear systems.

  A time-independent theory of syntax is, by definition, severely undermined in its implementational adequacy. Such a theory allows for stipulations to emerge without the limits that could be imposed if one considers, for instance, that all operations have to apply in polynomial time (i.e., language processing has to be a problem of the class P; specifically, an algorithm is said to be solvable in polynomial time if the number of steps required to complete the algorithm for a given input is upper-bounded by $O(n^k)$ for some positive integer k and where n is the length of the input in *bits*): it is hard to imagine how all feature-checking / valuation / deletion / inheritance operations (to name but a few, see Chomsky, 2000, 2001; Pesetsky and Torrego, 2007; Gallego, 2010; Ouali, 2010) *plus* the structure building and mapping operations Merge and Move (in the simplest picture, without even considering the materialization and linearization operation Spell Out applying several times during a derivation, following the seminal proposal of Uriagereka, 2002a) could be implemented in a system with limited working

---

[16] For the solipsistic reader, riddle me this: if a syntactic objects moves out of an island and there is no one to parse it, is the sentence still ungrammatical?

[17] This claim is based on recent experiments in which non-primates, like certain species of crows, have displayed the capacity of planning actions with a goal, and establish a hierarchy between needed actions, see for instance Taylor et al. (2010); Taylor and Cayton (2012).

memory and in polynomial time given the fact that there are no principled limits for what can be said to belong to UG. The traditional architecture for the grammatical system assumed in the Minimalist Program is the following (*apud* Chomsky, 1995; Marantz, 1995; Kitahara, 1997, among many others):

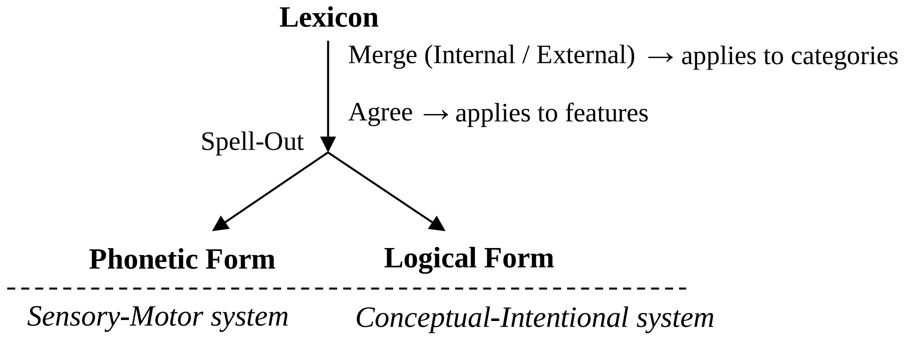


Figure 2: The architecture of the Language Faculty

This architecture has no (obvious) neurocognitive correlates or correspondences (see Poeppel and Embick, 2005; Embick and Poeppel, 2014; and our discussion of their take on the matter in **Section 1.4**), nor is it derived from limitations imposed by physical aspects of the substratum in which said grammar *emerges*. However, it should be useful to keep it in mind, as we will discuss a related architecture below, and our own proposal can be seen as a reaction to the unidirectional, function-based nature of the architecture assumed in the Minimalist Program. We will discuss aspects of this architecture in the remainder of this thesis, particularly, structure building and materialization and their corresponding physical interpretations.

Since our goal is to develop aspects of a neurocognitively implementable theory of language (in connection to the physical properties of cognitive systems), it is clear that 'generative' *in the sense of 'fully explicit'*, while being crucial, is not enough empirically: we require of a theory to account for the relations between Conceptual Structures CS (Taylor et al. 2007; Uriagereka, 2008, 2012; Jackendoff, 2002, Lobina, 2012b; among others) and linguistic structures, as well as how (and why) hierarchical structures are 'flattened' for purposes of their externalization as sound waves. Moreover, we also require of such a theory to establish reasonable limits for the time in which computations take place, in turn limiting the properties of what is actually computable in a classical sense (e.g., problems that can be computed in *exponential* time are not computable in *polynomial* time), and revisiting it with experimental predictions about what can be computed in polynomial time within an interactive model of computations. To summarize this point, the sense in which we understand 'generative' properly contains the orthodox Chomskyan sense ('fully explicit'), and complements it with a 'structure assignment in real-time' sense. We will also see that there are theoretical and empirical arguments to reject the notion of syntax being a 'generative' mechanism in the 'step-by-step structure building' sense (i.e., the *proof-theoretic* sense) from a field-theoretic perspective: syntax can be thought of as a topological operation that disrupts the ultrametricity of the conceptual and phonotactic spaces, and parametrizes them in the mathematical sense. In this respect, we will argue in favour of syntax as a *partitioning* rather than as a *generative* mechanism (see also

Schmerling, in press for a related view; and Hjelmslev, 1961 for an example of formal analysis under 'syntax-as-partitioning' assumptions). Syntax, in this work, will not operate over discrete *objects*, but rather over *spaces*.

- No domain-specificity (Cf. Fodor, 1983 and much related work): In short, this property of the concept of 'grammar' we assume amounts to saying that grammar assignment processes are not specific to language, being present in other faculties. As we said above, the process by means of which a structural description is assigned dynamically to relevant portions (in the technical sense of 'relevant', see Sperber and Wilson, 1986; Wilson and Sperber, 2004) of the phenomenological world is perhaps *the* defining property of perception, from a computational point of view (see, e.g., Pomerantz & Kubovy, 1989; Chater, 1996, among others). As such, the process requires mechanisms that are explicit and underspecified enough to be applicable to different faculties, which work with different substantive elements: the algorithm cannot be sensitive to the inner characteristics of the objects it manipulates, a feature we have referred to in past works as a 'blind' syntax (Krivochen, 2011 et seq., see also Boeckx, 2014 for a different take on the matter, closer to MGG). A blind syntax, that is, a mechanism of structural description assignment that is insensitive to the inner characteristics of the objects it assigns a structure to, is, we think, a better way to approach a theory of computational systems and their interfaces with other systems (e.g., meaning and sound) than going the other way round, which is the common practice in MGG.

  Thus, we propose, the computational mechanism that assigns a *figure-ground* dynamics to the cup of tea and the table it is on in front of me is, *in a non-trivial way*, the same one that assigns a linguistic stimulus a description in terms of what we customarily refer to as *phrase markers*, despite the obvious variations in the relevant alphabet. Differences in the form of structural decriptions depend on external conditions over the computations that operate over cognitive spaces (see Saddy, 2018 for details about this process). Non-domain specificity also means, crucially, *no Faculty of Language*, since the set of *elements* that are *both* innate and specific to language is the empty set (we have made a case for this in Krivochen and Kosta, 2013): here we have summarized previous discussion by saying that specificity is, if anything, an *emergent property* of the *interaction* between systems, which allows us to resort to the explicit and highly developed techniques used in dynamical systems theory, among other formal explanatory tools. 'Specificity' is an *interface property*, an *emergent* property, if you will (even though the concept of 'interface' will be challenged in this work, it is familiar to linguists). In a word, *there is nothing 'special' about language as a cognitive capacity or as a formal system, and that is desirable*[18]. We will see that such a claim has deep consequences not only at the theoretical level, where we are forced to abandon intra-syntactic stipulations, like the existence of purely syntactic filters or principles governing relations between elements, as well as features and Agree operations driving the syntactic derivation. This claim has also deep consequences at the empirical level, where we are forced to pay attention to a wider number of variables when considering a particular phenomenon and attempting an explanation. We explicitly claim that *no*

[18] Most emphatically: 'not special' does *not* mean or entail 'not worthy of attention'. There is nothing 'special' about, say, Hydrogen. It is the most abundant element in the Universe. Yet, without it there would be no life.

phenomenon in the human mind can be satisfactorily explained by resorting to a single, very specified, mental faculty in substantive isolation (*contra* the growing specification of syntactic mechanisms within MGG). Rather, the focus should be set on the interactions between the relevant systems, insofar as systems constrain each other in their operations and outputs: as we will see below in detail, the dimensionality flattening which, we think, is essential for externalization, is actually an interface phenomenon arising at the interaction of orthogonal planes within a complex nonlinear system.

Now that we have made explicit the features we assume for our grammar, which differ significantly from customary assumptions in MGG (as well as other formal, yet non-transformational models, like HPSG or LFG, see Pollard and Sag, 1997; and Kaplan and Bresnan, 1982; Bresnan, 2001, respectively) involving structure building as a *formal model*, the relation between grammar and its respective inputs and outputs, as well as the neurocognitive processes underlying the formal operations we assume; it is time to move forwards and formalize exactly what we mean by 'structural description', which is the other obscure term in our hypothesis, 'grammar' being the first. In short, we will propose the following:

> *A structural description SD of an input I is a formally explicit, abstract representation of I in terms of objects and relations in I built in real-time.*

Of course, more explanation is in order. On the one hand, we require the notion of 'representation' to be clarified and explicitly characterized in this context. On the other, we need to specify the alphabet (i.e., the finite set of symbols) with which these representations will be formed (technically, *generated*) and the operation or operations by which this generation takes place, as well as the properties of the space where those operations take place and how they influence the relevant computations. We will now proceed to do that, narrowing down our scope to natural language, the object of our inquiry for the rest of the thesis.

1.2 On the alphabet of linguistic structural representations[19]

Following the system outlined in Krivochen (2011; 2012a), in turn heavily based on the Relevance-theoretic approach of Sperber and Wilson (1995), and the Relevance-generative proposal of Escandell and Leonetti (2000), we distinguish two kinds of elements present in the modelling of structural descriptions for natural language strings: (1) *roots* (a.k.a. *conceptual elements*) and (2) *procedural elements*. The difference between these kinds of elements is given not by their format or inherent syntactic properties (we argue that there are no such intrinsic properties, *contra* Chomsky's 1995 et seq. *formal features*), but by their semantic properties, which in turn influences their possibilities of syntactic combination (that is, the syntagmatic relations they can enter into) in the context of a model in which syntactic structure is essentially semantic structure (see also McCawley, 1968; Dowty, 1979 for related views):

(1) *Roots* are pre-categorial linguistic instantiations of generic concepts (see Panagiotidis, 2010, 2013; Borer, 2014; among many others; we will return to the properties of the

[19] Warning for the non-linguist: this section contains a fair share of jargon, which has been kept to a minimum to the extent that circumlocutions were possible. The non-linguist may skip this section altogether, if it feels too dense. Or, he can consult Uriagereka (1998) for a very accessible introduction to generative jargon, which has been adopted and/or adapted, to different extents, by other frameworks (HPSG, LFG…). In any case, we will summarize the section's main points before **1.3**.

conceptual space and the characterization of concepts in terms of attractors in dynamical systems in **Part III**). Generic concepts of the kind we will assume are used to build conceptual structures (which might or might not be linguistically instantiated) are "severely underspecified" with respect to their extension as well as their format (which determines the formal properties of the structures in which they appear, including graph-theoretical and computational properties, both of which we will analyze below). This underspecification makes them unusable for the narrow purposes of building linguistic structures by themselves[20], so they can only be instantiated in a derivation under the scope of a procedural node (an assumption that has similarities to Distributed Morphology's 'categorization assumption' which requires all roots to appear in a local relation with a category-defining functional head by the time they are Spelled-Out, see Panagiotidis, 2013: 5 for a recent reference, although our version is *semantically* rather than *morphosyntactically* based). Roots convey generic conceptual instructions, and their potential extension is maximal (expressible by the superset that properly contains all actual, past, and potential referents for a given root, including non-factuals and counterfactuals), given their semantic underspecification: bare roots have no (spatio-temporal) anchor, and are thus not referential in any non-trivial sense: we cannot 'pick out' an element, or a set thereof, from the (mental representation of a) world, because we have no instructions with respect to what to look for.

(2) *Procedural elements* convey instructions for the system pertaining to how to manipulate, interpret, and relate semantic substance. Procedural instructions play two main roles for interpretative routines:

- They restrict reference to a proper subset of a root's extension. Each element restricts the set in different ways, for example:

  $\surd = \{\alpha, \beta, \gamma, \lambda, \delta \ldots \omega\}$ (each Greek letter represents a member of the extension of the root)

  a. $\{x, \surd\} = \{\alpha, \beta, \gamma\}$

  b. $\{y, \surd\} = \{\gamma, \lambda, \delta\}$

Where *x* and *y* are procedural elements having semantic scope over the root.

- They provide instructions as to:
  - where to retrieve information from, in the phase space defined by the dynamics of the system at a specific derivational point (whose characteristics we will make explicit in **Part III**)

[20] In theory-neutral terms, pure roots cannot denote either sortal or eventive entities (consider the root √water, for instance…in isolation, we cannot know whether it is used to denote a substance, or the event of pouring that substance on some affected object) nor can they serve as property predicates. These semantically interpretable functions are only available for structures of the kind [*procedural element*(*conceptual root*)].

- what kind of information to retrieve, relational or non-relational [(21)]

Procedural elements thus convey an essentially *locative* meaning in the sense that they relate a figure (i.e., the root) to a ground (a set of properties / sortal entities), and they are thus predicators (i.e., functors) which have to have semantic scope (i.e., a local hierarchical dependency at the component assumed to be in charge of semantic interpretation) over their (logical) argument. The mutual dependency between these roots and procedural elements is very eloquently illustrated by means of the well-known Kantian *sententia* '*Gedanken ohne Inhalt sind leer, Anschauungen ohne Begriffe sind blind*' (*Kritik der reinen Vernunft*, A 51/B 75). While we are well aware that the quote originally refers to the relation between sensorial information and *a priori* formal categories, we want to use Kant's words to emphasize that procedural instructions must operate over conceptual content, otherwise, they are 'empty' (e.g., a preposition with no complement). Likewise, conceptual content with no structuring instructions (e.g., a bare root) is 'blind'. In both cases, unless they combine thus limiting their respective phase spaces, they are *linguistically* useless.

Moreover, we will make a further distinction within conceptual elements *just for the time being*, between *Lexical NPs* and *sortal variables* (which we will denote by Δ, a notational device used already in Fiengo, 1977 and Wasow, 1979). Following the type-token distinction worked out in Krivochen (2015b), we will use numerical subindexes (Δ_1, Δ_2, …) to denote different tokens of the same type, and prime notation (Δ', Δ"…) to denote different types. Thus, [… Δ_1 … Δ'_1] is a structure in which we have a single token of two different types (see Bromberger, 1993 for a discussion of types and tokens in linguistics from a –primarily- philosophical perspective, which needs to be distinguished from the strictly syntactic approach taken here: both types and tokens are formal, not sensory, entities in the present work). In cases like this, in which there is only one token of each type, we will frequently omit the numerical subindexes (which would be just '1') for simplicity. Variables, which are non-rigid designators (including reflexives, reciprocals, and pronouns), enter the derivation comprising all possible outputs as far as their phonetic form and semantic interpretation are concerned, which will be contextually determined (in the formal language theory sense of *context*): an anaphoric or a pronominal form will surface *depending* on the presence of a Lexical NP, identified with proper names and, possibly, indexicals (following Russell, 1911) in a local domain. Therefore, it is not the form that determines distribution (as in traditional Binding Theory[22]), but quite the opposite: it is the distribution that determines the phonological form in a way to be specified shortly. For instance (using traditional bracketing for expository purposes):

13) John shaved himself
14) [$_{vP}$ John [$_{v'}$ v_0 [$_{VP}$ shaved Δ]]]

The VP-*v*P structure form a single domain (see also Grohmann, 2003, who calls the domain within which thematic relations are established 'θ-domain'), semantically, an eventive domain

[21] These points are related to the notion of 'conceptual localizer' in Relevance Theory, see Sperber and Wilson (1986: 86-92), also Escandell and Leonetti (2000: 367-368).

[22] This 'module' of the grammar is composed by the following principles (Chomsky, 1981: 188):

Principle A: an anaphor is bound in its governing category
Principle B: a pronominal is free in its governing category
Principle C: a referential expression (R-expression) is always free

including an 'initiator' or 'agent' argument (licensed by the causative node *v* heading a 'little *v* Phrase' *v*P)[23]. Therefore, the sortal variable Δ in the Complement position of V is given a morphophonological exponent –'pronounced', or 'materialized'-, as an "anaphor", a term that we consider an interface epiphenomenon when referring to the phonological form of a syntactic object (as opposed to a set of semantic properties, with which there is not always a perfect superposition, as the existence of long-distance anaphors as in Latin proves). In a more general spirit, we will consider all phonological forms epiphenomenal aspects of syntactic-semantic construal, as there is no one-to-one link between phonological form and syntactic distribution or semantic interpretation (for an empirical study concerning Case from the present perspective, see Krivochen, 2012a: 77, ff.; also Trejo, 2013: Chapter 2; see also Schmerling, in press for a fully explicit Categorial grammar built from the morphophonology up). The possible phonological outcomes of Δ are apparently the superposition of the following two states (Case / theta roles interpretations being -non deterministically- read off the configuration containing the variable and the structurally closest procedural head, Tense for *nominative*, *causativity* for *accusative*, Preposition for *dative*; see Krivochen, 2012a for details):

a) Anaphor (reflexives, reciprocals)
b) Pronoun (e.g., *him*, *her*, *it*)

However, it would be a stipulation to claim that *Lexical NP* is not a possible state of the referential sortal system, and it would give NPs a special place within the theory: something we would like to avoid, unless data should so require. A theoretical revision of the proposal above, including NPs into the Δ system, considers only phonological variables (that is, elements whose phonological exponent, or 'materialization' is dependent on that of some other element), which are logically referential expressions *all the same*. Consequently, the structure of (13) is actually –something like- (15):

15) $[_{vP}\ \Delta\ [_{v'}\ v_0\ [_{\mathrm{VP}}\ \text{shaved}\ \Delta]]]$

If hierarchy actually plays a role in reference assignment when interpreting a string, it does so *along with linearity* (Culicover, 2013): since top-bottom relations in a phrase marker does not *necessarily* correspond to left-to-right relations when that phrase marker is linearized (see, e.g., Uriagereka's 2012: 56 Mirror Linear Correspondence Axiom, which maps c-command[24] into follow-on relations instead of precedence), there must be some independent mechanism if syntactic structure is to be inferred from phonological order in terms of a linear function (a methodology employed widely in Generative Grammar, most notably since Kayne, 1994). However, if we look at things the other way round, departing from a Conceptual Structure CS and considering morpho-phonology to be a subsystem that forces the collapse of all possible

[23] We will exclude lexico-syntactic proposals like Hale and Keyser's (2002) for the time being, because it is not clear that the lexical structure (or l-syntactic structure) actually affects the binding properties of variables. In the case of a lexical decomposition model (e.g., Hale and Keyser, 2002; Mateu Fontanals, 2002; Jackendoff, 2002), the structure for the change of state predicate would be something along the lines of [John [DO [CAUSE [BECOME [John, shave]]]]]. The exact formulation varies from author to author, our sample structure is closer to the original decomposition approach in Generative Semantics (McCawley, 1968 and much related work).

[24] Following Reinhart (1976) and much related work, let us define that A *c-commands* B *iff* the first branching node that dominates A also dominates B, and neither A nor B dominate each other. *C-command* is said to be *asymmetric iff* A *c-commands* B but B does not *c-command* A: this qualification excludes sister nodes, which mutually *c-command*.

interpretations of referential variables to one of their possible outcomes to materialize them, the resultant system is quite different. This proposal is clearly different from the usual 'phonology-free syntax' position that is assumed in MGG, and disrupts the independency of components in the Y-model (see, e.g., Chomsky, 1957 for the first arguments in favour of the independency of syntax). It is crucial to bear in mind, though, that a deterministic system for the materialization of variables faces difficult problems: consider (16), from Bach (1970: 121)

16) The man$_i$ who shows he$_i$ deserves it$_j$ will get the prize$_j$ he$_i$ desires

Bach (1970) used (16) as part of an argument against the transformation *pronominalization* as defined, for example, in Lees and Klima (1963: 23). In effect, as Bach notes, a *pronominalization* approach woud yield an infinitely complex Deep Structure, with infinitely many embedded NPs. However, there seems to be arguments in favor of some structural sensitivity for variable materialization (which could take the form of preferences or gradients rather than deterministic rules): for instance, examples like (16') are ungrammatical:

16') *He$_i$ who shows the man$_i$ deserves the prize$_j$ he$_i$ desires will get it$_j$

That is, crossing dependencies (computationally *context-sensitive* local configurations) pose problems for a transformational theory of pronominalization (at least in the strongest version, in which all pronouns are equally transformationally derived, which as Postal, 1969: 202 among others points out, need not be the case), but as Jacobson (1977) argues in depth, the syntactic configurations in which crossing reference is possible (that is, patterns like [i…j…i…j…] for $i \neq j$) is severely restricted, and interpretation is not always straightforward (sometimes requiring relaxing referential identity to 'sloppy identity'). She takes these structural and semantic constraints to be arguments in favor of a theory in which surface pronouns do *not always* correspond to NPs at Deep Structure. In our terms, the argument can be recast as one against strict deterministic rules relating occurrences of variables and their materialization. So whereas an approach based on *c-command* (or *command*, as in Langacker, 1969) between referential variables before and after mapping operations might have shortcomings if interpreted as a set of obligatory and deterministic rules, language-specific cyclic conditions do seem to influence the possibilities for variables to surface as pronouns or anaphors in specific configurations where there is a competition for preferences in referential chains (see, e.g., the discussion of the pronoun/anaphor alternance in Dutch by Rooryck and Van Wyngaerd, 2011; also the rich inventory of examples in Culicover & Jackendoff, 2005: Chapter 10, among others. The concept of 'competition' in chains is developed in Martin & Uriagereka, 2014). If cycles are not aprioristic entities but emergent properties of derivations, inter-linguistic variation is only to be expected. Moreover, if binding is not a primitive of the theory of grammar, but an interpretive effect of cross-cycle relations at the semantics-pragmatics interface, putative 'exceptions' like the morphophonological exponents in Malayo-Polynesian languages noted in Reuland (2011: 23) could fall into place. We will not develop a dynamical syntax approach to binding here, but it is to be noted that there are enough objections to canonical Binding Theory to pursue alternatives that are not purely syntactic in nature (for some such alternatives, see Culicover & Jackendoff, 2005; Reinhart, 2006, among others).

Now, an elegant theoretical solution to modelling the lexicon would set formally uniform (yet crucially *not* static) criteria for referential variables, be they *sortal* or *eventive* (in Borer's 2005a, b terms). This is what we will propose now, as an extension of the system sketched above containing procedural elements and Δ sortal variables: the subset of the lexicon used to derive a given sentence is an array that consists of

- Δ variables, corresponding to sortal entities
- Γ variables, corresponding to eventive entities
- Procedural elements

Let us work with the traditional idea that there is a lexicon separate from the syntax, and that the syntax manipulates elements from that lexicon in order to build structure for a while longer. The question arises of what the nature of the elements in the lexicon is, and how they are instantiated in specific derivations. Here we will assume that the set of elements used to build linguistic structure in a particular instance contains abstract *types*, not *tokens* (see Krivochen, 2015b for an extended analysis of this idea and its application to displacement phenomena). Moreover, *types* do not correspond to full-fledged lexical items with featural specifications which determine their syntactic, semantic, and phonological properties (as would be the case in non-transformational lexicalist frameworks), but to variables in the sense specified above, whose category, reference, and morpho-phonological exponent will depend on the local relations that it establishes with other nodes. A *token*, in this sense, is simply a pair *type, context* (that is, a *type* inserted in a structure or otherwise manipulated by a rule). We will maintain the distinction between Δ and Γ as if it was a primitive of the theory, but it should be borne in mind that these themselves can be decomposed: recent approaches to *roots* within the Minimalist tradition (Marantz, 2007; Panagiotidis, 2014; Borer, 2017) assume a-categorial roots and a set of 'categorizer' functional heads (*v*, *n*, and maybe others) which determine that a single root -say, √water- would be interpreted as a part of speech or another –*water*$_{\mathrm{N}}$ vs. *water*$_{\mathrm{V}}$-; this perspective owes much (more than it seems to be willing to recognize) to the Generative Semantics enterprise in the late 60s and early 70s (McCawley, 1968, 1973; Lakoff, 1970, among others). Transformational rules, then, make reference to *tokens*, not *types*. In more general terms, which suffice for the purposes of the present work, the simplification of the lexicon we proposed above can be made more schematic, expressing a fundamental opposition between *relational* and *non-relational* elements, which goes deeper into levels of description than the Δ, Γ version: we could conceive of Γ variables (with the exception of some very primitive eventive entities, mainly presentational and stative unaccusatives, like [appear] and [be] respectively) as the result of of local dependencies between semantically underspecified roots related by means of a very narrow number of procedural elements, whose variability we will represent via binary features *in a purely descriptive manner*:

17) P: a terminal comprising locative instructions in terms of *central / terminal coincidence* relations (Hale, 1986)
V: a terminal comprising eventivity in terms of [+/- dynamic] (Mateu Fontanals, 2002)
v: a terminal comprising causativity in terms of [+/- affectedness][25]

[25] Mateu Fontanals (2002) adopts a *v* head for verbs like [have], in which it is not clear that there is even an external initiator (since the semantics of possession can be decomposed by means of just figure-ground dynamics with a central coincidence relation, as in Harley's, 2003 '$\mathrm{P}_{\mathrm{HAVE}}$'). However, it would not be quite correct to eliminate *v* altogether, since verbs like [have] do not behave like unaccusatives or ergatives, particularly when it comes to diathesis. We thus follow Mateu Fontanals in this point, much discussion pending. See Krivochen (2015c) for discussion about the procedural value of *v*.

D: a terminal comprising *sortality* in terms of specificity and definiteness (in combination with aspectual, modal, and temporal information).

Let us summarize the main idea of this section: every grammar operates over an alphabet. In the case of natural language, that alphabet contains (at least) *two* kinds of elements: conceptual elements (which provide semantic substance), and procedural elements (instructions pertaining to how to interpret the relations between substances). Those conceptual elements receive their morpho-phonological exponents partly depending on their distribution and local relations with other elements, and partly depending on cost-benefit relations at the time of building a representation of the meaning of the construction that those elements appear in. Materialization options for procedural elements are quite more restricted, for they belong to closed classes. Moreover, we have suggested that these elements are located in a mental space, whose properties and characteristics we will expand on in the remainder of this thesis. In this view, procedural instructions pertain to where to look for semantic substance and how to establish relations among substantive elements and between substantive elements and models of the world.

1.3 On generative operations[26]

A *generative grammar* for natural languages is an explicit formal system (Chomsky, 1965: 4; 1995: 162, fn. 1) based on an algorithm that recursively enumerates structural descriptions of well-formed strings in the form complex structures made up of discrete atomic elements in the manner of a proof (see Pullum, 2007). Essentially, the construction of structural descriptions proceeds in the following way: 'take simple things and combine them to make a more complex thing'. So far, a point on which transformational and non-transformational theories of syntax agree (and even pre-generative forms of structuralism), is that *syntax is computational* (derivations are developments of proofs), and *computation is discrete combinatorics*, operating on *discrete objects*. Let us focus on MGG for the time being. This computational system, in short, takes lexical items from an unordered set (the so-called Numeration) and combines them step-by-step binarily, establishing hierarchical relations between *pairs* of *distinct* objects. Current developments within transformational versions of generative grammar, from Chomsky (1995) on, have led to the proposal of a single, 'free' generative algorithm called Merge, which '*takes objects X, Y already constructed and forms a new object Z.*' (Chomsky, 2013: 40). Needless to say, stipulations aside, X and Y can be of arbitrary complexity, either terminals (i.e., lexical items, which do not branch) or non-terminals (branching nodes, i.e., sub-trees) built via Merge, which, in Uriagereka's (2002a) terms corresponds to the distinction between (a) Monotonic (which make the tree grow constantly and at a constant rate; expressible in finite-state terms[27]) and (b) non-Monotonic (non-regular) Merge.

[26] Most of what follows in this section is borrowed from Krivochen (2015a), although the discussion has been shortened and adapted to the present context. We refer the reader to that publication for details about phrase structure building under the present assumptions. Here, we will focus on alternatives rather than discuss the problems of current assumptions.

[27] Consider that finite-state models allow *head-tail* recursion, insofar as loops can consist of arbitrarily long chains of symbols, linearly ordered (i.e., excluding hierarchy and embedding). If this is so, as Uriagereka (2008: 233) proposes, then the Markovian character of monotonic Merge (or "merge to the root") follows straightforwardly: the monotonically assembled sequence [he [saw [the [book]]]] (and any other like it) can be expressed by means of a Markov chain, oversimplified as 'he → saw → the → book',

Merge was devised as an attempt to unify two defining properties of natural languages in the MGG lore, *hierarchical structure* and *displacement*. The former (External Merge) includes a theory of proper containment, dominance, or non-connectedness relations between syntactic objects; the latter (Internal Merge, or Move), a theory of how syntactic objects are interpreted in syntactic locations that differ from their phonological location. There are some crucial notions in the previous paragraph: one of them is the 'new object Z' part. Within the Chomsky Hierarchy CH, an inclusive classification of formal grammars (see Chomsky, 1959: Theorem 1[28]), External Merge (EM) (where X and Y belong neither to the same object nor to each other, but are taken from the 'Numeration', a subset of the mental Lexicon that is used to derive a particular expression) is regarded by Uriagereka (2008: 234) as a *context-free* operation, because it creates Z from any two X, Y available at a time in a derivational space. Note that Merge(X, Y) = {Z, {X, Y}} is the bottom-up equivalent of the PSR Z → X, Y. Thus, EM is available to Push-Down Automata, which allow *true recursion*, i.e., free center embedding (as opposed to Finite State Automata, which allow only *head-tail* recursion)[29]. Note, however, that this Merge operation can be further decomposed as follows:

18) a. Concatenation (join X and Y together)
    b. Labeling / Projection (form Z)

Labeling implies taking {X, Y} as a unit (or, alternatively put, identifying the categorial specification of Z and its distributional properties) for the purposes of further computations, which in turn determines its syntactic behavior (can it be an argument? can it *take* arguments?, etc.) and interpretation at the sound and meaning interfaces with the Sensory-Motor (SM) and Conceptual-Intentional (CI) systems, respectively. Such decomposition has been attempted, among others, by Boeckx (2009: 48), Hornstein (2009: 57-58), and Hornstein & Pietroski (2009), under mainstream Minimalist assumptions.

Of those steps, only the first is computationally context-free, insofar as Labeling requires the algorithm to peer into either X or Y to find categorial features to project (following orthodox assumptions, expanded on in Chomsky, 2013, 2015); this probing can go further than the last element introduced in the derivational space, thus, a 'last-in-first-out' stack is not enough. The operation proceeds in such a way that if X = V and Y = N, then Z = VP via percolation of V's categorial features; this operation determines that this object VP, as it is, cannot be an argument, but can take an argument (e.g., an initiator or 'effector', following Van Valin and Wilkins' 1996 proposal), can be anchored in time, etc. Some recent proposals on the Chomskyan side assume that other features than categorial features can determine labels, including agreeing ϕ features among other intra-theoretical stipulations (e.g., Chomsky, 2015).

with each lexical item configuring a state of the system. We did not include loops here, but it is equally simple to add *n* loops to the finite-state automaton, as in [he saw the old, boring, heavy book].

[28] '*Theorem 1: for both grammars and languages, Type 0 ⊇ Type 1 ⊇ Type 2 ⊇ Type 3*' (Chomsky, 1959: 143); where Type 0 = unrestricted; Type 1 = Context-Sensitive; Type 2 = Context-Free; Type 3 = regular.

[29] Also Medeiros (2015a, b) analyses dependencies traditionally considered to involve movement within a stack-perspective (i.e., implementable via a PDA): his stack mechanism, which we will briefly introduce in **Section 3.2.5** below, correctly predicts superiority effects as the ones arising in multiple Wh- questions like [who saw what?] and [what did who see?]. It remains to be seen if a stack model based on the Linear Correspondence Axiom (Kayne, 1994) as a linearizing algorithm can also accommodate phenomena like anti-superiority effects, which are known to arise in languages like Japanese and Korean (Jeong, 2003); not to mention non-monotonic or cross-cycle dependencies. These areas are currently under research.

Thus, Concatenate might be computationally context-free at most, as it merely 'puts things together', but Labeling must be as (mildly) context-sensitive as the structure mapping operation Internal Merge IM (formerly Move-α). This is the case if IM's (mild) context-sensitive nature (in turn, requiring *at least* an extended Push-Down Automaton, PDA+ to be computationally implemented, see Joshi, 1985) depends on IM *searching* anywhere within the limits of left and right boundary symbols (Uriagereka, 2008: 227-229) within an already constructed object W to extract / copy a syntactic object of arbitrary complexity $X \in W$ and merge it to W (this constitutes what is called *reprojection*, see Matushanski, 2006; also Uriagereka and Hornstein, 2002), or to $Z \supset W$, always *extending* the phrase marker in a bottom-up fashion (the so-called *Extension Condition*, see Chomsky, 1995: 189-190; Kitahara, 1997: 5-8) without modifying it ("tampering with it") internally, by definition. We will see that non-transformational models differ with respect to the assumptions they make about structure building, but they all implement *normal grammars*[30], thus, they all need some notion of *nonterminal*, which is always labeled in some way: differences in this respect are mostly notational (see, e.g., Bresnan, 2001, 2011: 117 for an LFG perspective; Steedman and Baldridge, 2011: 185-186 for discussion within Combinatory Categorial Grammar), and do not affect the core idea[31]. We beg the reader to keep in mind the concept of *label* in the sense of 'what determines how a certain unit is to be interpreted for the purposes of future computations', for it will prove useful when comparing different kinds of formal grammars: we will suggest it is *the* defining property of *normal* grammars: only those can operate over labels.

As opposed to Merge (and, in general, the format of Phrase Structure Rules that prevailed in the Standard Theory and its extensions), Jackendoff (2011) argues in favor of a different kind of structure building operation within a generative framework: Shieber's (1986) *Unification*. The formalism put forth by Shieber, and adopted by HPSG, LFG, and other non-transformational grammars, is heavily based on the concept of *feature structure*, assuming a very rich and specified Lexicon (see, e.g., Nordlinger and Bresnan, 2011 for a presentation of LFG; and Green, 2011 for an overview of HPSG; both LFG and HPSG operate over feature matrices via *Unification*). In fact, it does not apply to undetermined X and Y objects, but, by definition, applies to *feature structures*, which are defined as a partial function of *features* (*dimensions*, in Minimalist syntax) to *values* (see Pollard and Sag, 1994: Chapter 1): for

---

[30] As a reminder,

Chomsky-normal grammar: every context-free language is generated by a grammar for which all productions are of the form $A \to BC$ or $A \to b$. (A, B, C, nonterminals, *b* a terminal, including ε)

Greibach-normal grammar: every context-free language is generated by a grammar for which all productions are of the form $A \to b\alpha$, where *b* is a terminal and α is a string of nonterminal variables.

Unless explicitly indicated, we will use 'normal' to refer to either kind of formal grammar, for they are equivalent in most relevant cases. See also Post (1943: 198-199).

[31] As Postal (1964: 7) correctly points out, traditional grammar assumes the same basic 'phrase structure grammar' generative grammars do, with a terminological change: instead of identifying the immediate constituents of S(entence) as NP and VP, they are Subject and Predicate. A nontrivial observation to make is that headedness is *not* a *sine qua non* condition for labelling, for we can have in natural language exocentric constructions that are nevertheless labeled (including non-sentential units, see, e.g., Emonds, 1976: 52-56). Formally, the requirement for labeled nonterminals is a constant in all the frameworks we have reviewed so far. It is crucial to bear this in mind, for it will be significant when dealing with non-normal formal grammars.

instance, we can have a mapping from the feature [person] to the value [third] (which is a partial function insofar as there are other possible values), as in the following example:

19) $$\begin{bmatrix} \textit{cat}: & \textit{NP} \\ \textit{agreement}: & \begin{bmatrix} \textit{number}: \textit{singular} \\ \textit{person}: \ \textit{third} \end{bmatrix} \end{bmatrix}$$

Such a structure is abbreviated as D_{NP3Sg}. Feature structures vary in their specificity, and in the number of features they contain (here, we have only *categorial* and *agreement* features, but different formalisms have adopted more, including specifications for grammatical functions and subcategorization frames). Unification applies to feature structures in the following way (Shieber, 1986: 14):

20) In formal terms, we define the *unification* of two feature structures *D*' and *D*" as the most general feature structure *D*, such that $D' \subseteq D$ and $D'' \subseteq D$. We notate this $D = D' \cup D''$.

Some further clarification is necessary here. Within Unification grammars, ⊆ is used to symbolize a *subsumption* relation between feature structures, in which a feature structure, abbreviated D, contains *part* of the information of another D', such that D' ⊆ D. In more complex terms, the concept of *subsumption* is based on that of *dom(D)* (the *domain* of a feature structure, namely, the features it includes, regardless of their mapped values), such that D' ⊆ D *iff* $\forall(x) \mid x \in dom(\mathrm{D'}), x \in dom(\mathrm{D})$. A more concrete example will clarify the differences between Unify and (Chomskyan) Merge (taken from Jackendoff, 2011: 276, ex. 10 a, b):

21) a. Unification of [V, +past] and [V, 3 sing] = [V, +past, 3 sing]
(not *[[V, +past] [V, 3 sing]]*, as with Merge)
b. Unification of [VP V NP] and [V, +past] = [VP [V, +past] NP]
(not *[[V, +past] [VP V NP]]*, as with Merge)

We can recognize three distinct possible results of the operation, based on Shieber (1986: 15):

- Unification *adds* information (e.g., feature structures are not identical, but compatible)
- Unification *does not add* information (e.g., feature structures are identical)
- Unification *fails* due to conflicting information (e.g., same features, different values)

Merge could be formulated in such a way that derivations *must* be counter-entropic (that is, the informational load for the interfaces is always increased, at least if one interprets the *extension condition* and the *full interpretation principle* strictly). But this is only *one* of the possibilities with Unification, which makes the derivational dynamics, in our opinion, more interesting (insofar as the computational consequences of each possibility have to be taken into account at each derivational point, if Unification is to be implemented in a derivational model). Jackendoff (2011: 276) claims that Merge can be regarded as a particular instance of Unify, in which feature structures are reduced to two, in the following way:

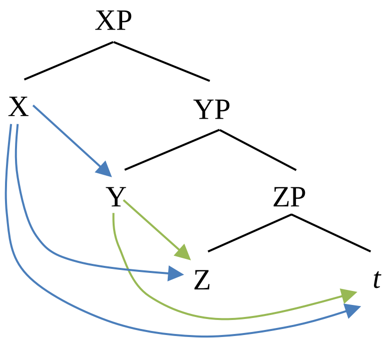
22) Merge (X, Y)
Unify (X, [*x y*]), being [*x y*] a feature structure containing unspecified terminal elements = [X, *y*], as $X \supset x$.
Unify (Y, [X, *y*]) = [X, Y], as $Y \supset y$.

Unification-based grammars do not consider a structure-mapping algorithm, since they are by definition *non-transformational*: there is no operation derivationally relating phrase markers, nor are there different, subjacent levels of structure (or conditions applying to them). Thus, these models are called *monostratal* (since there is no mapping between levels or components, even though there might be different components expressing different information, as in *c*- and *f*-structures in LFG).

Merge-based and Unification-based models of structure building include most major syntactic theories (MP –including Stabler-style Minimalist Grammars-, HPSG, LFG, Survive-Minimalism, Simpler Syntax). We have reviewed pros and cons of both classes of grammars in Krivochen (2015a; 2016c), here we will go further, assuming the discussion there.

It is crucial to bear in mind that, despite what some (widely assumed) theories imply, *models* of structure building are nothing but *models*, not the actual object of research. That is, a binary-branched tree is *not* a sentence or its structure, it is only a *model* of a sentence or its structure, just as Bohr's atomic model was *not* an atom. For non-linguists, this clarification might seem otiose, but within generative linguistics it is most important: MGG's theories of displacement and linearization, for instance, *depend* on linguistic structures being *actually* 2-D trees in the graph-theoretic sense (which has not been thoroughly analyzed, with the exception of the works of Roberts, 2015 and the slightly more topologically-oriented proposal in Uriagereka, 2011). Non-transformational theories are also, although to a lesser extent, committed to graph theoretic assumptions. While the properties of a hydrogen atom can be made explicit regardless of the model used (that is, a hydrogen atom will always have a proton and an electron, regardless of whether we are graphing it using Bohr's model, Schrödinger's, Rutherford's…), the properties assigned to a particular string of language largely (if not entirely) depend on the theoretical framework we are submerged in and the model they assume. For the sake of concreteness, let us consider Kayne's (1994) Linear Correspondence Axiom (LCA), which states that linear order is a function of asymmetric c-command relations between terminals[32] (see Section **2.9** below for clarification). LCA-compatible objects are *always* and *only* of the following form:

23)



X's asymmetrical c-command path (i.e., all the nodes X c-commands but that do not c-command X) is marked in blue and Y's, in green. Z does not asymmetrically c-command anything and nor does its sister (they mutually c-command, which is referred to as a 'point of symmetry'), but since the most deeply embedded element in the structure is a trace *t*, it need not

[32] The LCA formulation is as follows (we will come back to it in **Section 2.9**):

'*Linear Correspondence Axiom: [LCA] d(A) is a linear ordering of T.*' [A a set of non-terminals, T a set of terminals, *d* a terminal-to-nonterminal relation, defined as asymmetric c-command] (Kayne, 1994: 6) be linearized (as it receives no phonological exponent), and the potentially problematic point of symmetry situation is directly dismissed. The transitive c-command relations we diagrammed in (23) are translated into linear order as follows:

24) $X^\frown Y^\frown Z[^\frown t]$

Note that the linearization axiom (to which we will return below, as it is an example of function-based syntax, against which we will argue) rests on two fundamental, far-reaching stipulations (there are others, like the assimilation of specifiers and adjuncts, or restrictions over labeling, but they all derive, in our perspective, from these two):

- Structural uniformity under X-bar theoretic assumptions (regardless of structure building directionality, that is, the LCA is not necessarily tied to either *top-down* –PSR- or *bottom-up* –Merge- approaches, if linearization is strictly post-syntactic)

- The most deeply embedded element in a structure *must* always be a trace (or any other kind of phonologically null element) in order to break a symmetry point

However, there is an essential confusion here: while the linear character of the *signifié* is an observable phenomenon, theory-independent; the syntactic structure that is inferred from it is quite the opposite, a highly theory-dependent, purely abstract set of stipulations. The confusion between levels has lead MGG theorists like Adger (2012, p.c.), or Boeckx (2014) (see also Chomsky, 2013, 2015; to mention just the most recent references) to claim that every road leads to Merge (in different forms, always part of UG and frequently stipulated as an operation over feature bundles), and you cannot have an explanatory theory of language without assuming something like it (going so far as to say that '(…) *it would require stipulation to bar either of them* (…)' [External and Internal Merge], Chomsky, 2013: 40). Particularly after the explosion of the so-called 'biolinguistic enterprise' (see Chomsky, 2005; Di Sciullo and Boeckx, 2011; Fujita and Boeckx, 2016), MGG proponents seem to be under the impression that they are actually describing and 'explaining' something external to their particular formalism, in the most extreme cases, a biological reality (failing to note that, as Postal, 2004: 298 –paraphrasing Everett- does, that '*if linguistics were what the author* [Chomsky] *claims, syntactic trees would be visible in CAT scans*'). This is a problem for epistemology, and falls outside the scope of our work. We do stress, however, that the best we can aspire to for the time being is having a *model* of the object that can capture and faithfully represent the properties of mental processes and representations while attending to the proviso that we are *not* claiming linguistic structures are in fact the way we model them[33], a position that is not shared by most practitioners of MGG (who follow a Kaynean approach to phrase structure). After all, if physics still uses *models*, at this point in history, we cannot expect much more from linguistic theory.

1.4 On Linguistics and Neurolinguistics, Reductions and Unifications

We propose in this work an alternative model to transformational generative grammar and associated 'biolinguistics' in the present work; which, in our opinion, has the advantage of capturing some properties we expect to verify empirically. While models for phrase structure

[33] In Krivochen (2015a; 2017) we have resorted to a modified version of the well-known tree model purely for expository purposes, but the same considerations apply. We are quite confident that the problems we tackle (e.g., the structure of coordination, adjunction, iteration, and their consequences for computational models of linguistic competence) exist independently of the specific framework we utilized, and arise in other frameworks as well (HPSG, LFG, Simpler Syntax, Construction Grammar…).

building and apparent 'mapping' have been thoroughly presented in past works (Krivochen, 2015a, b), there is a missing step, which is in fact crucial: how do we link those models with neurocognitive conditions and set up an empirically plausible program? In this respect, Poeppel and Embick (2005); Embick and Poeppel (2014) find an essential discontinuity between the objects of study in linguistics and neuroscience, which they call a 'granularity mismatch':

> ***Granularity Mismatch Problem*** *(GMP): Linguistic and neuroscientific studies of language operate with objects of different granularity. In particular, linguistic computation involves a number of fine-grained distinctions and explicit computational operations. Neuroscientific approaches to language operate in terms of broader conceptual distinctions.* (Poeppel and Embick, 2005: 104)

This mismatch, in their terms, prevents the formulation of both computationally and biologically plausible links between linguistics and neuroscience. There is a fundamental incommensurability between the units of linguistics and neuroscience, and this 'hiatus' arises at each of the different levels of analysis and involves the relevant units at each level. Their (2014) piece repeats the case, with very little to add.

It is crucial to note that, when Poeppel and Embick (henceforth P&E) write 'linguistics', the reader should recover *generative* from the context. That is, non-trivially, P&E's case argues for the incommensurability of primitives and operations assumed *in MGG* and those studied in neuroscience (meaning, there is no *direct, one-to-one correlation* between objects and operations). This is evident from their comparative table (updated 2014 version, p. 3):

	Linguistics	Neuroscience
Objects	Distinctive feature	Dendrite/spine
	Timing slot	Neuron
	Morpheme	Cortical microcircuit
	Phrase	Cortical column
Operations	Feature spreading	Long term potentiation (LTP)
	Merge	Oscillation
	Concatenation	Adaptation
	Semantic composition	synchronisation

Table 1: Examples of hypothesized primitive objects / operations.

It is obvious that the assumed elements in 'linguistics' are largely theory-internal and narrowly concerned with syntax (and in passing with propositional semantics): the dependence on Merge (which, by the way, is not formally distinguished from alternatives like Unification or Concatenation in either of P&E's papers), or feature spreading in the syntactic sense, is an exclusive characteristic of the Minimalist Program. Other notions, while more widespread, are not exempt from problems, like 'phrase' (why not 'constituent' or 'nonterminal'? Do languages like Piraha or Riau Indonesian, according to the analyses of Everett, 2005, and Jackendoff and Wittenberg, 2017 and Gil, 2009 respectively, have anything resembling 'phrases' in the MGG sense, or 'constituents' in the structuralist sense?). Recently, operations of feature spreading and reassembly have also undergone healthy criticism and revision from a Harmonic Grammar perspective (Putnam, 2017), in which candidate competition and satisfaction of mutually conflicting constraints play a crucial role. There is also no mention to Item-and-Process grammars (including Montague grammar and variants of Categorial Grammars), whose form is radically different from Item-and-Arrangement grammars (which include MGG). Nor are important critical discussions about the untenability or inconsistency of the generative-biolinguistic ontology of syntactic objects and primitives that are just assumed in **Table 1** even acknowledged (Katz & Postal, 1991; Postal, 2009; Behme, 2015a). In more general terms, we argue that the nature of fundamental linguistic objects and operations has to be problematized rather than taken for granted.

After presenting this object / operation mismatch between linguistic and neuroscience, Embick and Poeppel (2014: 3) say that

> (…) *crucially, without any straightforward way of aligning and connecting the inventories that the two types of theories identify, it is not possible to speak of any sort of unification.*

We think that the GMP is not a matter of principle (as they claim), but of methodology. The discussion, however, is necessary. For starters, it is not obvious that we want a *unification* of neuroscience and linguistics, and it is even less obvious that the only way to achieve mutual understanding and non-trivial collaboration is through unification (or reduction, a term that is often mentioned in the biolinguistic literature, in an inexplicable attempt to do with linguistics and neuroscience what was once done with *two well-defined scientific fields*, *physics* and *chemistry* attending to scale issues). With this criterion, mathematicians and physicists should not be able to understand each other, let alone collaborate, for primitives of mathematics and physics are not strictly parallel. However, we do have very precise mathematical models of physical phenomena (after all, the Standard Model is a collection of numbers, just to give an example), see Tegmark (2007) for the strongest position in this respect, namely, that the physical reality *is* a mathematical structure (the so-called 'Mathematical Universe Hypothesis'). Following P&E's criterion, the relevant units in mathematics for the purposes of characterizing each particle in the Standard Model are numbers, whereas the units in physics would be (for the sake of concreteness, leaving aside much discussion about fields, to which we will return in detail below) particles / forces. What is more, there is no physical entity that corresponds to a number, nor can there ever be: numbers are abstract entities *par excellence*, and particles or their associated fields are subject to conditions that do not affect numbers (the Pauli Exclusion Principle for fermions, for instance). The present argument might seem a *reductio ad absurdum*, but the reader should first consider if P&E's own argument is not a *reductio ad absurdum* of their own position (as P&E build their argument from an extreme position), or alternatively, of MGG biolinguistics (see Postal, 2004: Chapter 11 for very critical discussion of some aspects of so-called 'biolinguistics' as presented by Chomsky; also Postal, 2009). The spirit of the GMP, the idea that syntax and neuroscience operate over objects that are essentially different, is hardly controversial. The devil is in the details: linguistics as MGG syntax, unification as the goal, etc. These substantive issues notwithstanding, the core of P&E's work, and the spirit of the GMP, need to be seriously considered by 'biolinguists'. A methodological warning is certainly due, as there have been recent attempts within MGG to indeed identify (establish an isomorphism between) different brain oscillatory frequencies with Merger of syntactic objects to strong (Transitive) / weak (Intransitive) phases. In these works, to give but an example, the highly theory-internal notion of *feature Agree* is in some recent works mapped *directly* to measurable brain oscillation frequency coupling without even considering aspects of incommensurability between observables and stipulations. Thus, we get things like

> *A review of the cognitive neuroscience literature will suggest that non-Phase Heads (PHs) may oscillate at gamma, that transitive PHs* […] *would do so at* [frequency] *beta2, and that intransitive PHs oscillate at beta1. In fact, this triangle represents the three elements of phrases/phases: complement, head, and specifier/edge*. (Ramírez Fernández, 2015: 78)
>
> (…)
>
> *A first division can be done by claiming that non-PHs are sustained by gamma oscillations and PHs by beta oscillations. Furthermore, among PHs, two beta sub-bands may implement transitive and intransitive PHs: beta2 and beta1, respectively.* […], *intransitive PHs bear appealing resemblances to specifiers and internally merged elements, which prompts one to consider them as a single elementary category*. (Op. Cit.: 79)

The problems with this approach should be obvious: as we will discuss shortly, there is simply no reason to assume that primitives of brain dynamics correspond to primitives of *a particular linguistic theory* (in this case, MGG): Ramírez Fernández's position (shared by Murphy, 2015, 2016, among others, sometimes with variants, making MGG 'phases' correspond with patterns of coupled oscillations: in all cases, there are no field equations provided, nor is there an explanation of why this putative correspondence should be the case) is the mirror image of the strong icommensurability hypothesis of P&E, and falls short for the same reasons. The notions of Phase Head, complement, head, and specifier/edge not only have no natural correlates in non-phrase structure based models (or even phrase structure models that do not assume the traditional X-bar skeleton), they are strictly intra-theoretical insofar as they do not represent anything *in the data*, but *in a model* (one, of many possible models) *of selected data* (what constitutes the 'core' of the grammar –and mostly, the grammar of English-, more on this below). In principle, a deep analysis of the brain dynamic oscillations involved in language processing is of course desirable, but without a critical analysis of the data, without an adequately powerful theory of computation (where 'adequacy' is to be understood as in Joshi, 1985: 208), and without a psychologically plausible theory of 'language' (including syntax, semantics, morpho-phonology, pragmatics, and their interrelations), this position is nothing short of untenable (or trivial). It is also crucial to bear in mind that theories of processing and neurobiology of language must not be confused with theories of grammar: psychological experiments or brain imaging do not shed light on the properties of specific constructions (say, gerunds vs. gerundives; or the non-recursive properties of non-restrictive relative clauses; or…), which falls under the scope of *grammar* see Langendoen & Postal, 1984 for some discussion about the ontology of grammatical objects). In summary, the unification way equates theoretical constructs with observables, and P&E's argument refutes this position by adopting a polar opposite stance. In our opinion, neither is justified either epistemologically or empirically.

That said, and overlooking some more fine-graded gaps in P&E's argument, it is useful to bear in mind their tripartite approach, as follows, to the relations between linguistics and neuroscience, if for no other reason, because they assume these disciplines to be fundamentally different, and consequently, the (essentially programmatic) relations between these disciplines which these authors propose take this essential difference into consideration (Embick and Poeppel, 2014: 4):

- Correlational neurolinguistics: Computational-Representational (CR) theories of language are used to investigate the Neuro-Biological (NB) foundations of language. Knowledge of how the brain computes is gained by capitalizing on CR knowledge of language.

- Integrated neurolinguistics: CR neurolinguistics plus the NB perspective provides crucial evidence that adjudicates among different CR theories. That is, brain data enrich our understanding of language at the CR level.

- Explanatory neurolinguistics: (Correlational + Integrated neurolinguistics) plus something about NB structure/function explains why the CR theory of language involves particular computations and representations (and not others).

Again, the equation ‘linguistic theory = computational-representational theory’ does not come for free, and there is no discussion about the nature of the computations involved (but the authors do cite Gallistel and King, 2010 when discussing computation, so we can quite safely assume they are on the ‘digital-function-based computation’ side). While we will not address our questions in terms of these levels of neurolinguistic integration, they are more likely to be expressed in theory-neutral terms, with the appropriate caveats (e.g., about the nature of computations and representations). While still incompatible at fundamental levels with our proposal, a reformulation of the work in the neuroscience of language in terms of P&E’s distinction would constitute a great improvement over existing theories of ‘biolinguistics’, and perhaps even a whole research program on its own.

Coming back to the ‘incommensurability problem’, let us assume for the sake of the argument that P&E’s position is not, as we have suggested above, a *reductio ad absurdum* of ‘mainstream’ biolinguistics ignoring issues of scale and emergent behavior. Is *unification*, an example of which we saw in Ramírez Fernández’s extract, the only way to go? Or, is it even desirable? Does ‘commensurability’ imply such unification? Let us address these questions in connection to alternatives proposed from within cognitive neuroscience.

Reduction and *unification* are meaningful concepts if we are dealing with fields or objects that are essentially the same and can be mapped onto one another in a straightforward way, while losing as little information as possible; otherwise, there is simply no point in unifying, as the loss would be much greater than the gain (in terms of specificity, depth of insight, and definition of the object). P&E assume that, if CR and NB cannot be put in correspondence, they cannot be related at all. That is, in our opinion, overlooking half the picture: what if we are not looking at linear systems? What if we can gain insight on computation by looking at the biological substratum of computation, at microscopic, mesoscopic, and macroscopis levels? What kind of formal theory is allowed by, and consistent with, conditions of neurological plausibility? (see Piattelli-Palmarini & Uriagereka, 2008; Saddy & Uriagereka, 2004 for some perspectives on these questions, somewhat different from –much closer to MGG than- the perspective adopted here) A whole set of possibilities, questions, and concepts becomes available, including the essential notion of *emergence* as a staple of nonlinear dynamical systems. McClelland and Vallabha (2009), from the perspective of connectionist networks, make a useful distinction between *mechanistic* and *emergent* dynamics. This is useful for laying out our own conception and contrasting it with P&E’s (and also Gallistel and King’s) take on linguistic computations and representations, which impact on their incommensurability. McCelland and Vallabha (2009: 2) define *mechanistic* dynamics as a set of rules or equations that stipulate the behavior of a system, more specifically, a set of *differential* equations. The evolution of the relevant system is completely deterministic, there being no room for stochastic effects or non-linear behavior. *Emergent* dynamics, on the other hand, are related to '*several mechanistic subsystems*' (McCelland and Vallabha, 2009: 2) interacting with each other. Connectionist approaches and dynamical systems theory do share some basic assumptions:

> *(a) mechanistic dynamics are best described by differential equations that are rooted in biology and*
>
> *(b) psychological phenomena such as categorization and selection reflect the emergent dynamics of the system.* (McClelland and Vallabha, 2009: 4)

The computation is, in this view, an emergent property of the neurocognitive substratum, an idea with which we wholeheartedly agree: there is no 'incommensurability' problem if the relation between computation and physiology is one of emergence rather than of unification.

For the purposes of the present work, and particularly because the derivational system assumed by P&E (that is, the theoretical apparatus of the Minimalist Program) is function-based and describable entirely in mechanistic terms (particularly in a strongly feature-based system, like Pesetsky and Torrego, 2007; Di Sciullo and Isac, 2008; Panagiotidis, 2011; Stroik and Putnam, 2013, in which having the feature matrix of each element, we can safely reconstruct the derivation step-by-step). We will take mechanistic systems to be describable (although perhaps not exhaustively) by means of recurrence relations, and non-linear dynamical systems, to be describable (although perhaps not exhaustively) by means of differential equations. A system, viewed at in the right scale and with a global perspective, can display computationally mixed dynamics; however, if we focus only on a single level of organization and look at it locally, we will usually be stuck with structural and computational uniformity. Consider, for instance, one of the three arguments in Kornai (1985) in favor of finite-state models of natural language stringsets, the one concerned with aspects of neural computation:

> *As individual neurons can be modeled by finite automata (McCulloch-Pitts 1943), and a finite three-dimensional array of such automata can be substituted by one finite automaton (see Kleene 1956), NLs must be regular.* (Kornai, 1985: 4)

Such argument assumes structural uniformity *all the way down* (or up) levels of organization, so that mechanistic dynamics arise trivially, because the assumptions determine so[34]. P&E do a good job in this respect, showing there are differences at different levels of analysis (but their assumptions lead them in what we consider the wrong way, towards an impossible unification).We adopt McClelland and Vallabha's distinction, and link both kinds of dynamics assuming the object in question is a complex non-linear system. Relevantly, the concept of *emergence* does not need either reduction or unification; properties that are not directly predictable from the individual behavior of units arise when those units / actants interact (Boccara, 2002; Ladyman, Lambert, and Wiesner, 2012). Assuming there are neural bases for mental computations (that is, rejecting a dualist position with respect to the mind-body problem,

[34] Interestingly enough, Kornai's arguments are the mirror image of Chomsky's (1957): Chomsky argues that, since there are portions of English that are not finite-state modelable, then English is not a finite state language. That is, at a minimum, a leap of faith: the existence of portions of a language that go beyond Type 3 complexity does not mean that the language itself (understood, as in computer science, as a set of strings) is *all* beyond finite-state complexity, as we have shown in Krivochen (2015a), Krivochen and Schmerling (2016), and Bravo et al. (2015) with empirical arguments from English, Spanish, and Latin.

contra Chomsky, 1966: 40), a possible research program consists on looking for the neural structures that generate linguistic (or, more generally, symbolic) representations as *global emergent* properties of their internal dynamics, which can be, at a very *local* level (e.g., neuron-to-neuron), mechanistic in nature. *Neither is reducible to the other*, as they have a different computational and physical nature, but we can link them in a non-trivial way if we give up the unmotivated belief in the linear relation between physical substratum and computational properties. That linearity is exclusive of certain kinds of *hardware-software* relations (as in a PC), but claiming that the same reasoning is valid for the mind-neuron relation is a leap of faith that is not backed up empirically: there is no proof that the brain is 'wired' like a computer, or that it operates like one (that is, following digital, function-based principles, as argued by Gallistel and King, 2010). Nor is there evidence that cognitive systems are linear, quite on the contrary (Beim Graben et al. 2007, 2008; van der Mass and Raijmakers, 2009; Beim Graben et al. 2004; Saddy and Uriagereka, 2004; Schöner, 2009, among many others). Crucially, dynamic systems are characterized by a 'loss of stability' (Schöner, 2009: 2), as the system is permeable by external perturbations (multiple sources of information, 'noise' in a strict information-theoretic sense). The tension between stability and instability is at the core of the notion of *dynamical frustration*, which depends, precisely, on the conflict between opposing or mutually incompatible tendencies (requirements, constraints…) remaining unsolved:

> *The essence of complexity may lie here: A system without dynamical frustration will either settle to equilibrium or grow without bounds.* (Binder, 2008: 322)

Two of the *leitmotifs* of the present work are linked in this quotation: complexity and dynamical frustration. And now we introduce a third: *field theory* and its relation to neurocognitive dynamics. From a brain dynamics perspective, Schöner (2009: 2) claims that Dynamic Systems Theory (DST) '*is a specific set of concepts*', which are worth reproducing here:

> (1) Behavioral *patterns* resist change; that is, they are stable. This may be mathematically characterized by considering behavioral patterns as the attractor states of a dynamical system.
>
> (2) Behavioral *change* is brought about by a loss of stability.
>
> (3) Representations possess stability properties, as well, and can be understood as the attractor states of dynamic fields, that is, of continuous distributions of neuronal activation.
>
> (4) Cognitive processes emerge from instabilities of dynamic fields.
>
> (5) Learning occurs as changes in behavioral or field dynamics that shift the behavioral and environmental context in which these instabilities occur.

Given this set of concepts, can we show that we can *construct* an explanatory program for linguistics (without loss of generality, for these considerations apply to cognitive sciences as a whole) stemming from conditions defined in dynamical field theory, given specific limitations over the system? Proceeding with the utmost care, we will attempt to do just that. First, we will describe the theory, and the predictions it makes, here and in **Part II**. Discussion and details about field dynamics and their relation to computation will follow in **Part III**, and empirical linguistic evidence will be analyzed in **Part IV**.

1.5 Grammar assignment or grammar recognition?

Assume the following scenario: we are presented with a piece of data and a structural description. What is the relation between them? Does the structural description refer to something that is *within* the data? In more philosophical terms: is there structure in an input if there is no one to interpret it? An example of a mild *structure recognition* stance can be Shirley's (2014: 12) claim that '*not only can humans **locate patterns** in complex data, but that the drive to do so can lead us to find patterns everywhere, even in random data*' (our highlighting). This entails that the data *has* a structure, internal and inherent to it. Whatever cognitive operations we carry on when interpreting (function-based or not, automatic or not…), they are aimed at *extracting* somehow the structural description that is *within* the data, patterns that can then be generalized. Therefore, we *find* patterns (i.e., structure) in strings, instead of *assigning* those strings a provisional structural description in real time, which is progressively 'polished' (in roughly the lines of Townsend and Bever, 2001). A view of language in which a given natural language is a set of strings (Chomsky, 1957, 1965) entails a strong version of *structure recognition*, insofar as the speaker needs to extract the structural description out of a stimulus and match it with the set of well-formed structural descriptions SD, by means of a function that matches a sentence S to SD by a grammar G (Chomsky, 1965: 31). There are, we think, many problems with this 'structure location' approach, which underlies many theories not only of language as a cognitive capacity (including MGG as well as those versions of LFG that assume an Augmented Transition Network parser, see Kaplan, 1972, 1995: 11 for an example – although we must note that Kaplan is very much aware of the limitations of the approach, and explicitly restricts the 'comprehension' capacities of ATN to syntactic relations between constituents-), but of general cognition (e.g., Hauser, 2009; Fitch, 2013; Zink, et al., 2008) and computational modelling of natural language grammars (Berwick, 1984; Ristad and Berwick, 1989). To begin with, the assumption that humans *locate* patterns in data (i.e., that patterns exist *in* the data) is controversial: when moving out of symbolic strings (either numerical, linguistic, or some other nature) to consider wider perception issues, the model does not apply so clearly (this does not entail an endorsement of *structure recognition* applied to language, it just states that the models are explicitly worried about linguistic stimuli but not about other kind of perceptual stimuli which display syntactic properties all the same). Do the cup and the desk it is on at the moment, for instance, configure a figure-ground dynamics *inherently*? Or is it the case that, since the most salient object for the purposes of my perception is the cup (as it contains the tea I need to stay awake and write this), I assign that static, atelic event a figure-ground dynamics in which the cup configures the *figure*, thus the perceptive foreground, and the desk is the *ground*, the background or spatial context, both related by means of a *central coincidence* spatial relation (Hale, 1986)? The question is not trivial. Not only does it call into question the nature of the *objects* involved (objects, in a wide sense, comprising both eventive and sortal entities, and, among the former, central- and terminal-coincidence locative relations), but it also raises further questions about the computational properties of the structure-assigning system (call it a *mind*), its versatility, and the way in which it makes sense out of non-obviously related entities. Studies on the visual capacity have argued in favour of an innate capacity to distinguish figure from ground, perceptively salient objects from the background in which they appear, and, more generally, relevant features from non-relevant features in an array of sensory information (e.g., in the case of facial recognition in newborns, see Slater and Kirby, 1998; also Blake and Sekuler, 2006 for a more general perspective on perception and its relation to the so-called 'Plato's Problem', a.k.a. 'poverty of stimulus argument'). The extent to which those arguments pass over to abstract structures (e.g., arithmetical / geometrical) or language is far from clear.

A second, but not less important problem is the assumption that there exists something like *random data* (see Matlach & Krivochen, 2016 for discussion from representational and derivational perspectives). If one adopts a *grammar assignment* position, the problem just becomes more serious: is there such a thing as *random* data? Strictly speaking, there is a limit to data variation, which is given by limits on possible physical structures. But let us be more practical: assume we have an algorithm to generate structure with a finite number of variables $\{x, y, z, \ldots, n\}$; an example of such an algorithm would be good old Merge, which takes at most *two variables* at each derivational point in the traditional formulations, to be bound by lexical terminals. All one can do is try to control as few variables as possible, always under the belief that the procedure is linear (i.e., outputs are proportional to inputs and the system is closed). If the system should be non-linear and / or chaotic, things would work very differently from square one (as we will see in detail below). However, in a linear system, since each variable can take a finite number of values (if the system is required to be physically implementable, we cannot take ℕ, say, as the set of values for our algorithm, because it would never halt), the number of possible local combinations is finite. Sure, possible combinations are increased if the number of variables is increased as well, but the set of local combinations is always finitely enumerable. This certainly relativizes the concept of 'random' data, at least from a mathematical point of view. In a system where you have only two production rules and two possible values (0-1, or [bi]-[ba] syllable types in the case of Shirley's and Saddy's work), the 'random' character of a string (which might be so, widely considered, for experimental purposes) is quite dubious under strict theoretical considerations. What looks 'random' representationally might not be derivationally, and vice versa: equal transition probabilities between adjacent symbols in a string (representational randomness) does not correlate with equal weights assigned to nondeterministic expansions for a nonterminal symbol in the grammar that generates said strings. The representational and derivational properties of the stimulus must also be distinguished from whatever a parser can 'learn' about that stimulus: if the transition probability between adjacent symbols in a string remains equal and constant throughout experimental trials, for instance, the 'randomness' of the string is formally constant…but if the system in charge of interpreting that string is capable of adjusting internal weights and *learn and adapt,* the transition probability for this system will change, even though that in the generative mechanism (the grammar that generates the input) will remain the same.

The last problem with the *grammar recognition* approach that we will consider here is quite more practical: there is *always* more than a single structural description compatible with the input, both in language and in other cognitive capacities (for specific discussion about language, see Ross, 1969a; Heinz, 2007; Townsend and Bever, 2001: 86-87). This goes against theoretical claims of *structural uniformity*, which attempt to make all inputs fit a single structural description skeleton. A good example of strong *structural uniformity* is given by the binary branching axiom in MGG, best expressed in the form of X-bar theory (Chomsky, 1970; Stowell, 1980) and its revisions and extensions (none of which has addressed core structural assumptions, see e.g. Chomsky, 2013, 2015, who rephrases binarity and endocentricity under a slightly different rhetorical guise), but which was already present in early forms of generative grammar and even pre-generativist approaches to syntactic structure (e.g., Tesniére, 1959: 14-17).[35] Interestingly enough, such uniformity pretension was not featured in pre-generative and

[35] It is to be noted that, against common conceptions, binarity is *not* a recent thing: it was present since the first days of generative grammar, as part of the definition of generalized transformations. For example, Katz & Postal (1964: 12) claim that '*The recursive power* [of a generative grammar] *resides in*

primarily structuralist theories, see e.g. Wells (1947); Hockett (1958); Lyons (1968), which acknowledged the necessity of, for instance, distinguishing endocentric and exocentric constructions (Lyons, 1968: 231), paying attention to both morphology and semantics. There was no formal criterion (although Wells does follow the methodology of 'definitions' also used by Bloomfield, 1926), but we think structuralist insights should not be abandoned if they can be adapted within a wider explanatory framework.

Some of the problems and consequences of adopting a single template for syntactic structure (the 'principle of invariance of linguistic complexity' in the terms of Sampson, 2009) are introduced by Culicover and Jackendoff (2005: 113):

> *One might wish to argue that binary branching is maximally simple on grounds of Structural Uniformity. The argument would be that, first, there are many relations that are binary, and second, we would gain maximum generality by assuming that all relations are binary. However, since everywhere in the universe as well as everywhere in language there will be binary relations, it would follow from this reasoning that only binary relations exist everywhere. It is difficult to see the value of such reasoning, unless it were to turn out that there were positive empirical consequences of generalizing binary relations uniformly, and doing so required no additional complications of the theory. To our knowledge this has not been demonstrated for language, let alone in general.*

It is not trivial to note that almost any complex formal object can be modeled in terms of binary relations, from abstract patterns of plant growth to the Fibonacci or Lucas sequences, via L-grammars (Prusinkiewicz and Lindenmayer, 2010; Uriagereka, 1998: 192-193). Moreover, other kinds of generative devices (e.g., *concatenation* interpreted as *n*-ary addition) can also give us mathematical 'monsters' like π, decomposing multiplication and division in series of additions (with positive integers for multiplication and negative integers for division). However, this does not mean that the (computational) *nature of the object* is itself binary, something we can see in linguistic objects if we assume derivations are semantically motivated (in a related vein to Generative Semantics as well as the Simpler Syntax hypothesis, but strongly derivational unlike the latter, see Krivochen, 2012, 2015a, b, 2016c, d). However, is binary branching always the simplest option? And, 'simplest' is defined with respect to which criterion or set thereof? What about adequacy (in the sense of Joshi, 1985: 208)? Computationally, if our grammar is semantics-sensitive, there are elements that belong to the set of binarily branched structures which do not belong to the set of finite-state expressions, refining Uriagereka's distinction. Recall that, according to Uriagereka (2002a, 2012: 53), any exhaustively binary branched tree is expressible in a finite state fashion (the so-called 'Finite-State limit on phrase structure'), a point previously noted by Greibach (1965: 44)[36]. If the system is insensitive to

Generalized Transformations, i.e., those which operate on a set of P-markers [phrase markers] *(**probably always two**) to produce a single new derived P-marker* (…)' (Our highlighting).
This restriction over the format of structural descriptions (the requirement that they be exhaustively binary branching trees) has had far-reaching consequences at both theoretical and empirical levels, some of which we reviewed in Krivochen (2015a) and Krivochen and Schmerling (2016).

[36] The relevant quote is the following:

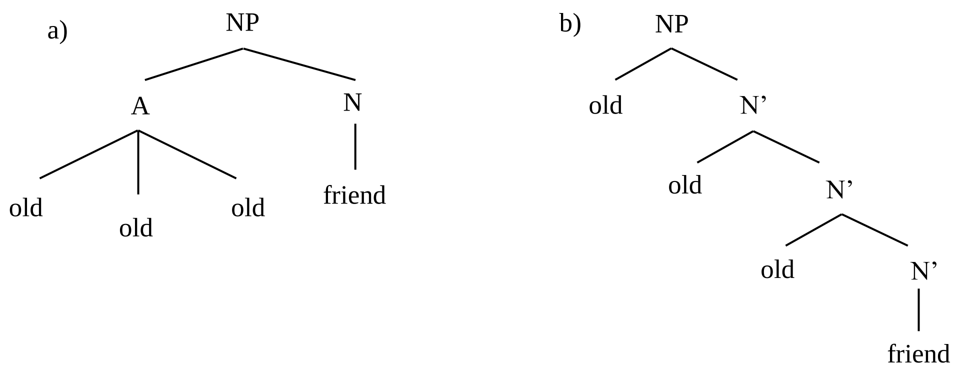
> *a given finite-state language L can be generated either by a psg* [Phrase Structure Grammar] *containing only left-linear rules: Z → aY, Z → a, or by a psg containing only right-linear rules: Z*

semantics, that *is* true. However, if the semantic relations between the substantive elements involved in the structure are taken into account, the results are quite different. Consider, for instance, the case of iteration, to which we will return below (see also Schmerling, 2017 for a similar perspective on reduplication):

25) An old old old friend

As discussed in Krivochen (2015a), there are (at least) two ways to represent the relevant portion of the structure in a phrase structure grammar, the part involving iteration:

26)



Binary branching is good for graphically representing predicate-argument relations, such that given a syntactic object SO = {a, {b, c}} (where {} mark unordered relations), #a# has semantic scope over {b, c} as a whole, and over #b# and #c# as terminals because #a# c-commands #b# and #c#: May (1985: 5) defines that '*The scope of a is the set of nodes that a c-commands at LF*' (see also Ladusaw, 1980). The relevance of binary branching for scope relations has been highlighted, among others, by Kayne (1984, esp. Chapter 7), where it is stressed that binding relations depend on *unambiguous paths* between operators and variables, of the kind that can only be achieved through binary branching. Actually, since the syntax feeds the semantic component in MGG[37], we can safely say that it is precisely binary branching that creates predicate-argument (and operator-variable) relations. If this is the case (and it *is*, for MGG), then in (26 b) there is a scope relation between the instances of [old], such that the highest instance takes [old old friend] as an argument, the second highest takes [old friend] as an argument, and so on. However, the semantic interpretation of the string is *not necessarily scopal*, i.e., monotonically hierarchical (certainly, not *uniformly* scopal!), but most likely linearly *incremental* (very old), something we can see in many examples: consider Led Zeppelin's 1971 'Rock and Roll' recurrent line 'It's been a long [lonely, lonely, lonely, lonely, lonely [time]]', which is interpreted as 'very lonely' rather than the monotonic '[a lonely time [which was lonely [which was lonely]]…]' (the latter being the interpretation we would expect to be read off a monotonically assembled phrase structural description following MGG-LCA

→ Ya, Z → a, and a psg containing either only left-linear rules or only right-linear rules will generate a finite state language (Greibach, 1965: 44)

[37] Importantly, other theories (HPSG, LFG) make a strong statement about the relation between lexical information and syntactic structure, but often remain at the margin of 'design' discussions, basically, 'what feeds what'. Moreover, while different structuralisms gave different importance to different aspects of linguistic theory (e.g., Bloomfieldian focus on morphology and phonology; Saussurean focus on the sign as the unit of analysis…), there was no ontological claim with respect to 'derivational timing'.

assumptions)[38]. Therefore, (26 b) is not an appropriate syntactic representation. But, what is more, it is not even finite-state compatible: finite state languages cannot encode scope, because the relations are strictly linear, at most, they can be taken as additive (e.g., a⁀b⁀c = a+b+c…). This means, in short, that the limit Uriagereka suggests is correct in purely formal terms (i.e., a finite-state grammar can indeed be expressed by means of binary-branching trees and *vice versa*) but this is only so if no semantic considerations are added to the picture. Note that [old, old, old friend] is *rhetorically* equivalent to [oooooold friend] (Newman, 1946; Schmerling, 2017), which would also be fitted into the binary template in MGG completely overlooking its semantics. Therefore, in contrast to a popular position among reviewers, (26 a) and (26 b) are *not* equivalent for natural languages, where semantic considerations cannot just be left aside.

Even if this matter has been (at least hopefully) clarified, there is still a problem, in principle: both structures in (26) are *formally* possible for the relevant example, and indeed there is a (third) possible reading in which at least one instance of [old] has scope over the others, an interpretation roughly paraphraseable as ‘a friend who is old in age and whom I have known for a very long time’, corresponding to the structural representation

27) [old [old old [friend]]]

So, we have three syntactically possible structures, two of which yield semantically plausible readings as well, and there is no criterion or set thereof to choose among them: both (26 a) and (27) –the latter, a ‘mixed’ representation- are in principle consistent with the data. If the string [an old old old friend] *had* an inherent structure that subjects (or a ‘performance model’) deterministically *recognized* in logarithmic or polynomial time (see, e.g., Ristad and Berwick, 1989: 389-390), this multiplicity of structural descriptions for a single string should not be a problem (there would be only one structural description *per* string, description that would be somehow encoded in the string and that hearers / interpreters recognize), but it clearly *is*. In relation to this, it is interesting to consider Petersson and Hagoort’s (2012) take on the interpretation of Artificial Grammar Learning (AGL) experiments:

> *A common assumption in the field is that if participants, after exposure to a grammar, are able to distinguish new grammatical from nongrammatical items, then they have learned some aspects of the underlying grammar. However, there is sometimes a tendency to assume more that participants process the sequences according to the grammar rules and strong claims are made about the representation acquired. However, this need not be the case.* ***The use of a particular grammar does not ensure that subjects have learned and use this, instead of using a different and perhaps simpler way of representing the knowledge acquired.*** *Several AGL studies have not sought to determine the minimal machinery needed to account for the observed performance, often leaving open questions about the nature of the acquired knowledge* (Petersson and Haggort, 2012: 1972. Our highlighting)

This passage touches on the practical problem of interpretation (that is, how the experimenter interprets the results of the relevant test) as well as the more abstract problem of grammar competition, within an *assignment* framework. The possibility that a ‘simpler’ way (or merely a ‘different’ but computationally equally complex way; a different segmentation of a string) is

[38] We are overlooking extensions of the binary-branching proposal, like Cinque (1999), in which prenominal adjectives are specifiers in functional projections FPs, stacked over the NP. The semantic interpretation corresponding to these stacked markers is the same as in (26 b).

used to parse a certain input (or substrings within an input) is, as these authors point out, frequently overlooked. In AGL, when performing a grammaticality task, one should not simply assume that the grammar being used to classify the input is actually the grammar participants were exposed to. This assumption, however, is very difficult (if possible altogether) to dissociate from the experiment itself in *grammar recognition* frameworks: after all, if a grammar was used to generate a certain input, and the stimulus *has* a grammar, it cannot *have a different grammar than that which was used to generate it*. For instance, if we assumed {0, 1, 1, 2, 3, 5, 8, 13, 21…} *has* a grammar, the only possible formal device that we as experimenters would accept as a correct hypothesis about the structure of the string is a Fib grammar of the kind we have reviewed. As we will briefly discuss below, that would be at least too strong an assumption, given the fact that there is never a single way to represent the structure of a string, for there is very rarely a single possible structural description compatible with a given string, *and*, for any grammar G we can come up with G' whose weak generative capacity is equivalent at least for the relevant example or set thereof (i.e., their weak generation capacity overlaps, such that a finitely enumerable *subset* of strings can be weakly generated by either one). Informally, we will say that the relevant grammars are weakly equivalent for surface strings s_1, $\dots s_n$, $n \geq 1$. [39] For L-systems, Rozemberg and Salomaa (1980) have claimed that there is no way of proving that a string s_x has been generated by G or G' unless by brute force. This means that, given s_x, there are infinite grammars that are weakly equivalent for the purpose of generating s_x. As should be at least apparent, this is a problem for grammar recognition approaches.

Another example of the necessity to (at least) relativize or weaken the *recognition* hypothesis is what we call 'Veritasium effects'[40]. The relevant experiment is quite simple, yet interesting. Subjects were presented with a sequence of numbers, '2, 4, 8' that, they were told, followed a rule, and were asked to (a) produce a different sequence that also followed the rule, and (b) enunciate the underlying rule. Unsurprisingly, all subjects (7 in total) produced sequences of their own which, according to the experimenter, followed the rule (say, '4, 8, 16', or '3, 6, 12'), but when they enunciated what they thought was the underlying rule to generate the relevant sequences, their hypothesized rule was not what the experimenter had in mind. All 7 subjects assumed the rule was something like 'take a number –even or odd-, multiply it by 2, and then again by 2'. Basically, a geometric progression in which the relation between the n^{th} and $n^{th}+1$ element in the sequence is defined by a 'doubling' function. However, the rule the experimenter had in mind was quite simpler: 'numbers in increasing order' (such that '1, 2, 376' would have followed the rule). Needless to say, the numbers the experimenter used were chosen to generate a structural ambiguity, so that more than one grammar was compatible with the specific input '2, 4, 8'. It would be incorrect to say that the subjects were in any way *mistaken* when they enunciated the rule, because it is the case that the sequence can be generated by the geometric grammar (talking about a 'mistake' would entail setting a specific, arbitrary frame of reference in a privileged position without any principled reason to do so). The thing is, it is not

[39] See also Patel et al. (2015), who define *weak grammar equivalence* as follows:

> ***Definition 5.2 (Weak grammar equivalence).*** *Two grammars G1 and G2 are weakly (grammar) equivalent if the following two conditions are satisfied:*
> *1.* $\rho_\infty(G_1) + \rho_\infty(G_2) = 1$ *where* $\rho_\infty(G_1) \neq \rho_\infty(G_2)$
> *2.* $A(G_1) = A(G_2)$ [for ρ_∞ density of ones in infinite strings, *A* the autocorrelation value for a grammar *G*]

[40] Because the idea is based on a video by the YouTube pop-sci channel *Veritasium.* URL: https://www.youtube.com/watch?v=vKA4w2O61Xo [Accessed on 10/11/2014]

the only grammar that generates that string, and there are many others (e.g., 'multiples of 2 that are not multiples of 3'). As there are no semantic conditions over numerical functions, we do not have the advantages we had when considering the iterative pattern in (25): there is no interpretative guide to lead the grammar assignment process, and based on limited data (a single string), there is not enough information to assign a unique structural description with complete certainty. As the number of interface systems imposing 'legibility' and 'usability' conditions decreases, the conditions over grammar assignment are increasingly nonspecific, and more structural descriptions can fit the data (if 'Kolmogorov complexity' was not an issue, we could have potentially infinite SDs for a single string; the Kolmogorov complexity of an object refers to the the length of the shortest computer program that produces the object as output).

The examples above are intended to raise awareness with respect to the consequences of adopting a *structure recognition* model, assuming structure is inherent to the object. Furthermore, if *structure assignment* is combined with *structural uniformity* assumptions, the results are increasingly problematic, both theoretically and empirically, as the format for the rules of our grammar become more and more strict, to the point of forcing the data into their own patterns instead of attempting to resolve the structure assignment problem based on either *simplicity* or *accuracy*. These concepts are not exempt of problems, but, as we will argue throughout this thesis, the possibilities of integrating mutually incompatible requirements into a comprehensive model of cognitive computations –viewed as emergent properties of the physical substrate- under a perspective that tries to *take advantage* of the tension between physical-computational conditions instead of just eliminating that tension in favour of one of the terms, are quite promising.

1.6 The rise of the machines

In **Section 1.1** above we said we would not be making a difference between an automaton and a real person yet, and since we have not introduced the relevant considerations derived from the physical constraints over computations, we still have to define what exactly we mean by 'automaton'. We will see that some aspects of the following definition, particularly the discreteness of states and the relevance of functions (which is based on the voluminous literature on the topic; see Hopcroft and Ullman, 1969; Lewis and Papadimitriou, 1998, among many others) will change when we make the progression from formal function-based digital automata to embodied computational systems under interactive-based computation assumptions[41].

> Automaton (based on references mentioned above): discrete-state, function-based deterministic computational system operating over an alphabet Σ in finite time, with finite input and output sets for any time T_x.
>
> - State: the *m-configuration* of the automaton, comprising the current representation D_T (Turing's 1936 '$\mathfrak{G}_r$') and the rule(s) involving / applying to D_T (Turing's condition q_n from the possible conditions over the *a-machine*).

[41] In the present work we will not address cultural influences on cognition, or the permanent interplay between common knowledge and real-time computation, including guesses about what other participants in an interaction might know or guess, etc. For a recent discussion of the cognitive relevance of cultural factors, see Everett (2016).

Unlike Arbib's (1987: 24) and Turing's (1936) definitions, the definition of 'automaton' above does not define the members of the output set a priori, there being a learning process involved which can adjust mappings between inputs and outputs, and external conditions influencing the derivation at points that cannot be foreseen (and thus cannot be incorporated into a function-theoretic set of instructions for the automaton). Moreover, we need our toy automaton to *not* be time independent, as we focus on the interpretative and generative procedures applying at each derivational point D_T and their mutual influence. We do artificially quantize time and states, for the purposes of the exposition, and will keep doing so in the rest of the present thesis. This does not mean, however, that such quantization is intended as a hypothesis about the nature of the object being modeled, quite on the contrary, we argue for a continuous, analogic approach to cognition and cognitive interfaces (more on this below), and the *field* theoretical approach we adopt is based on the idea of *nonlinear dynamical continuity* with entangled states in an open procedure, rather than linear digital discreteness for a closed function ot set thereof.

In a 'free Merge' view (Chomsky, 2004; Boeckx, 2014; Krivochen, 2011), the generative procedure itself can be blind to representations, but in our revision, structure building and interpretative routines cannot: thus, the automaton as a whole is not insensitive to input-output (co)relations, which can be adjusted in real time by appealing to information external to the machine's *m*-configuration. Note that this possibility has great impact on the computational complexity of interpretation: the assignment of structure to a string can be broken down into *uneven* pieces (*contra* standard Divide and Conquer algorithms; see Krivochen, 2016c), based on different kinds of cues (e.g., phonological, semantic, pragmatic…), and then 'solved' by means of procedures that are permanently influenced by (a) the environment in which the interpretative process take place, and (b) the intermediate outputs of computations being carried out *in parallel* (see Goldin and Wegner, 2007 for discussion about this last point). Information encapsulation is not always the way to go.

Arbib's (1987: 24) definition of an automaton by its input, output, and state sets is compatible with (although not equivalent to) Chomsky's (1965: 31) 'goals' for a linguistic theory, since states and outputs are defined beforehand (sentences and structural descriptions respectively), and the derivational procedure that produces outputs is independent of time. More recent proposals acknowledge the limitations imposed by not taking the time variable into account. However, if we consider time, the P-NP (-NP hard) problem arises for all processes computed by a neural network (e.g., a finite-state based Hopfield network): are the relevant grammar-assignment computations performed in (deterministic / non-deterministic) polynomial time? Is it possible that the structural description outputs of a grammar G_x can be easily checked, but not so easily generated? (this is an informal formulation of the P vs. NP problem; see Hartmanis, 1989 for historical context and a partial transcription of Gödel's foundational letter to von Neumann, 1956). The time-dependency factor will be essential in the presentation of our physically based formulation, since we will resort to differential equations to model cognitive processes as nonlinear dynamical systems. But we still have to discuss some things further, including a development of the theory of functions within cognitive science, and more discussion about the automata that each theory licenses.

Let us now consider the proposal made by Arbib (1987: 26) regarding the relation input-output, which is one of the major points we will set our focus on when comparing functions and interaction as models for neurocognitive processes:

> *for a brain* […] *the current output of the network need in no sense be a response to the current input regarded as stimulus. Rather, the firing of those neurons that provide the output can be influenced by activity within the network reflecting quite ancient history of the system*

This formulation of input-output relations is not quite compatible with strict interpretations of function-based computation, in which the output is a function of the input (even if stochastic). However, for this proposal to work, the definition of an automaton taking into account output sets should be modified, because the output set is, by definition, incomplete and undecidable, provided any input and any computational operation. Moreover, the decidability problem was proven not to be solvable for Deterministic and Nondeterministic Turing machines, although decidability is possible for regular, Context-Free, and Context-Sensitive languages within superpolynomial time (at most).

In the present thesis, a *representation* is defined simply as a model-internal fully explicit description of the state of a system minus the operations applying at the relevant derivational point. Thus, strongly derivational though our model might be (since there are no internal conditions over representations, nor is there any principle stating that all and every step in a derivation is to be well formed, *contra* classical GB assumptions), it is impossible by definition to get rid of representations. It is important to note that even those approaches that claim to be 'representation-free' (e.g., Epstein and Seely, 2006; Surányi, 2010) assume a generative operation that, in the simplest case, takes two atomic objects α and β and creates the ordered set <α, β>, which by the pairing axiom of Zermelo-Fraenkel set theory is equivalent to {{α}, {α, β}}. Given this scenario, we would need to resort to some special stipulation to claim that the atomic elements are not representations, or that the resultant object of combination in whichever form (call it pure *Concatenation*, as in Krivochen, 2015a; *Merge*, as in Chomsky, 1995; *Unification*, as in Shieber, 1986…) is not a representation, labeling issues apart (see below for some more discussion about the notion of representation in cognitive science, albeit from a completely different stance). In a cognitive model in which there are several systems, and specificity does not reside in substantive elements of one or the other system, but in the particular emergent properties from the interaction between different systems, it is essential to *conserve* information when interacting / interfering: completely destructive interference between any number of fields is in principle incompatible with the cumulative nature of mental operations, even if some information is bound to be lost and some energy is bound to be dissipated. In more traditional terms, this amounts to saying that cognitive interfaces, which arise by necessity in a modularist framework, should optimally be as little entropic as possible (a thesis we will revisit under the light of Uriagereka's 2012, 2014 CLASH model). At this point, a crucial question arises: what is it that we should conserve? This has been a relevant question since the first models of transformational linguistics (in which syntax, semantics, and morpho-phonology are three different components), and there are basically two possibilities: either we conserve structure, or we conserve content. There is a logically possible third option, namely, we conserve both. However, that option assumes a complete isomorphism between representations belonging to different cognitive systems, which operate under different conditions, in terms of the manipulated elements as well as the computational constraints over what counts as a string of the language generated by that system. Such a position is in fact quite rare, and to the best of our knowledge there is no theory that assumes *transparent* cognitive interfaces, at least not as the rule. The proposition of *opaque interfaces* (and, more generally, *opaque interaction*, without implying that there is domain specificity) makes sense for interactions *between* systems (thus the *interface* part). However, are purely syntactic operations supposed to follow a general conservation principle as well? Emonds' (1970, 1972) *Structure Preserving Hypothesis* specifies that constituents cannot be moved *into* certain structural configurations[42], a mirror image of Ross' (1967) islands constraints, which specify the conditions under which elements cannot be *moved out* of certain structural configurations (Complex NPs, adjuncts, direct objects within coordinate structures, among others). Early MGG was completely focused on *structure* preservation (i.e., maintain the integrity of structural descriptions under transformations, see Chomsky, 1986a for a reinterpretation of structure preservation which is not quite in the line of Emonds' original formulation: Chomsky's version was completely dependent on the X-bar template, and made the Structure Preservation Constraint on transformations '*next to contentless*' –J. Emonds, p.c.-), the content of such structural descriptions (i.e., the specific symbols involved in a phrase marker) being left aside. Assuming a different architecture, LFG's focus was set on mappings between *c*- and *f*-structures, derived by functional annotations to *c*-structure rules; see Kaplan (1995: 16-18). Nodes N in the *c*-structure tree were mapped onto units of the *f*-structure space F by means of a *structure correspondence* function $\phi\ N \rightarrow F$ within a Context-Free or, at most, mildly Context-Sensitive grammar (see also Bresnan et al., 2016 for details): configurational issues were given privilege here.

The Minimalist Program, unsurprisingly, has not gone far from the earlier MGG perspective: their strong commitment to Structural Uniformity means that a syntactic operation must not change the basic graph-theoretical properties of a representation: the Extension Condition makes sure every operation extends the tree from the bottom up, whereas the very definition of the generative operation Merge guarantees that the result of either External or Internal Merge always results in an {α, β} situation (where α is a head and β an XP, which, according to Chomsky '*is virtually everything*', in discussion with Cedric Boeckx, 2009: 52). Any syntactic operation will result in such a configuration. However, there is a catch to it: not every {α, β} configuration, even of the {H, XP} kind, will be interpretable. The problem is, following a strict 'digital', function-based conception of the computational system; there is no way of finding that out until the relevant object is transferred to the interfaces, be it a *phase-based model* (Chomsky, 2001, 2008) or a *Multiple Spell-Out based* model (Uriagereka, 2002a, 2012). While syntactically well-formed, the result might not be convergent. Thus, preserving structure does not guarantee anything, really: a structure-preserving transformation can give rise to an interface-illegitimate object, a claim that leads us to at least question the role of structure-preservation in natural language grammars (cf. Gallistel and King, 2010: 55, who assume a strong *homomorphism* holding in mapping operations; Williams, 2003 for a narrow-linguistic perspective on structural homomorphism between several levels of representation; see also Emonds, 1970, for the original proposal about structure preservation as a constraint over transformational rules in generative grammar).

It is useful at this point to compare and contrast natural language phrase markers with non-linguistic phrase markers generated by the same kind of grammar: let us consider the case of the class of L-grammars generating Fibonacci trees, which we will call 'Fib grammars'

[42] More specifically (partially assuming the Minimalist machinery), '*A structure-preserving transformational operation is one in which α substitutes for β, where β cannot be specified for a feature differently than α.*' (Emonds, 2007: 347). The original version (from 1970) involved the requirement that an element α must have been able to be base generated at position γ if γ is intended as a target for Movement of α.

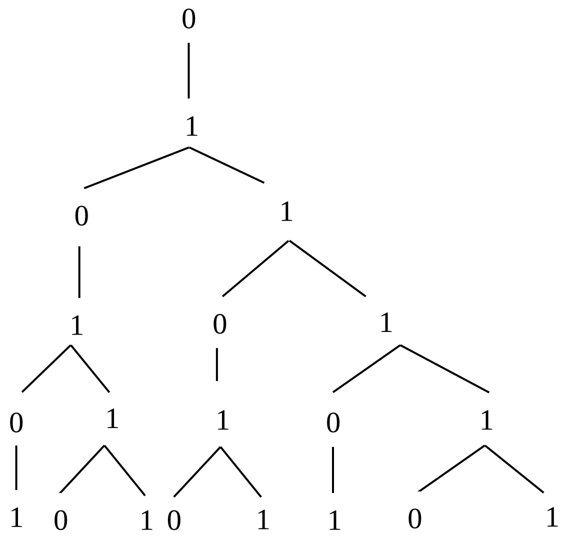
(Uriagereka, 2014: 383, ex. 32 a; see also the sample grammar offered by Prusinkiewicz and Lindenmayer, 2010: 3, which is –coincidentally?- the simplest Fib grammar. We will come back to L-systems below). Let us provide an example of the grammar and a partial derivation:

28)

Axiom: 0

0 → 1

1 → 0 1

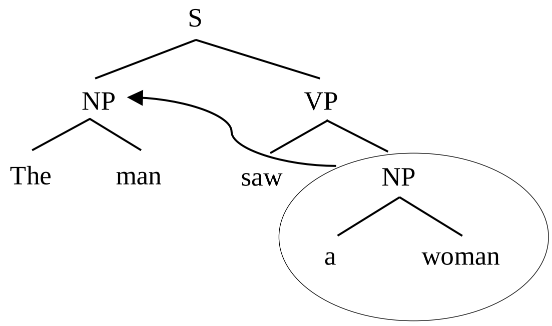


We can formulate an operation here which targets a ‘1’ and everything it dominates, and replaces any other instance of ‘1’ in the tree with the initially targeted one, which is identical in constituent structure as they are both the ‘projections’ of the ‘weak’ member of the alphabet (0, which rewrites as a single element). Within a Fib tree, we can apply this substitution operation freely. We can also, for instance, replace every instance of ‘1’ which is immediately dominated by ‘0’ but not the instances of ‘1’ which are immediately dominated by ‘1’. Or, change the left-adjoining properties of the grammar and make them right-adjoining. Sure, the resulting string might be a mirror image of the original, but we are not (unlike Uriagereka) concerned with linearization issues here, and even if we were, independent evidence in favor of the LCA (or its theorematic formulation LCT which assumes c-command in the base step; Uriagereka, 1998: 202-204, 2012: 83-84) would have to be provided[43]. The point is, any such transformation leaves the emergent properties of the grammar untouched. Let us see what would happen in a very simple (and simplified) linguistic tree built following a sequential Chomsky-normalized context-free rewriting grammar (see Martin, 2010 for some discussion about the concept of Chomsky-normal grammar):

29) S → NP + VP
NP → Det + N
VP → V + NP
Det → {the, a}
N → {man, woman}
V → {saw}

[43] We do not mention frameworks like HPSG or LFG here because they usually encode precedence relations in lexical attributes (Pollard and Sag, 1994: Chapter 1) or ‘linear precedence statements’, rather than in an operation that applies *after* PSR. This is to be expected, since non-transformational theories more often than not assume only a single level of syntactic description (and sometimes a further level for semantic tiers).

30)



Let us now review the original definition of *structure preservation* within linguistics:

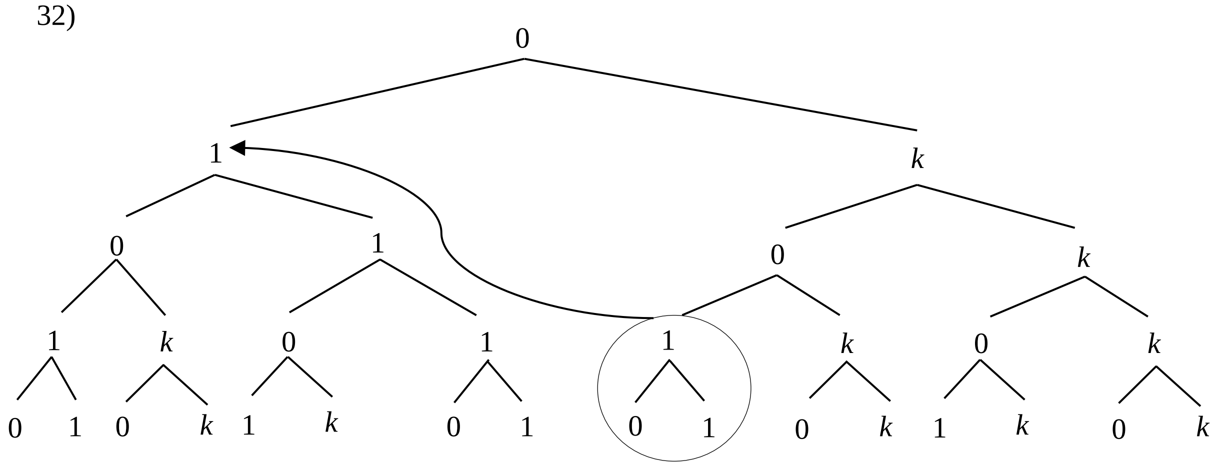
> *A phrase node X in a tree T can be moved, copied, or inserted into a new position in T, according to the structural change of a transformation whose structural description T satisfies, only if at least one of two conditions is satisfied: (i) In its new position in T, X is immediately dominated by the highest S or by any S in turn immediately dominated by the highest S. (A transformation having such an effect is a root transformation.) (ii) The new position of X is a position in which a phrase structure rule, motivated independently of the transformation in question, can generate the category X. (A transformation having such an effect is a structure-preserving transformation)* (Emonds, 1970: ii. Highlighted in the original)

A transformation replacing [$_{\text{NP}}$ The man] with [$_{\text{NP}}$ a woman] (whatever that could be) would undoubtedly be structure-preserving (as would be any of the tweaks over the Fib tree we suggested above), but while the result of the transformation applied to the Fib tree is strongly equivalent to the pre-transformation phrase marker (we still get the Fib sequence as an emergent property if adding the 1s and 0s *per* generation), we certainly cannot say the same about the linguistic X-bar tree, even if structurally they share fundamental properties (they are both rooted, oriented, labelled trees; see McCawley, 1968 for discussion). Where is the difference? Well, the reader might think we are cheating because we are comparing structures generated with different alphabets, {0, 1} and {the, man, saw, a woman}. However, it is not the case that the *number* of elements in the alphabet impact on the formal properties of the graph. Let us picture the following scenario involving a deterministic L-grammar (we will come back to them below):

31) Axiom: 0
$0 \rightarrow 1\ k$
$1 \rightarrow 0\ 1$
$k \rightarrow 0\ k$

Here we have added a non-null (i.e., $k \neq \varepsilon$) element to the alphabet, a constant k. Moreover, we assume that a symbol cannot rewrite as two identical terminals (e.g., $0 \rightarrow 1\ 1$ is forbidden) and the F part of the grammar cannot be the same for two different Σ (which is in fact a simple counter-redundancy statement), to add further conditions and see where they get us. If we develop the tree up to the 5$^{\text{th}}$ generation, we get:

32)



The sequence that we get is not quite Fib: the grammar in (31) yields the sequence 0, 1, 1, 3, 5, 11, 21, … by adding 1s at each generation, and 0, 2, 2, 6, 10, … by adding all non-zero symbols at each generation (i.e., 1s and *k*s). This means two things: on the one hand, that Fib is an *emergent* property of *certain* L-grammars under *specific* conditions (which makes them all the more interesting, and creates the necessity of making the conditions explicit in order to understand the properties of the emergent pattern, something we will come back to below; see also Saddy & Krivochen, 2016b). On the other, that structure preservation *alone* (which holds, as we can see) does not contribute much to our purposes; the *content* of the rules (the symbols that appear on the right-hand side of the transition function) is also relevant for the purposes of accounting for emergent behavior. Non-trivially, it is to be noted that *any node is a root node in a Fib-tree*: that is, any node is either identical to the highest node -0- or identical to the node immediately dominated by the axiom -1-. The crucial part is this: having no semantic component, there is no way of telling pre- and post-mapping phrase markers apart, and this makes a great difference in the comparison between L-grammars and natural language grammars. We thus disagree with Gallistel and King (2010) (G&K hitherto) in their claim that:

> *The mapping* [between representing and represented systems, the mind and the world respectively] is ***structure preserving*** [read: isomorphic]*: The mapping from entities in the represented system to their symbols is such that functions defined on the represented entities are mirrored by functions of the same mathematical form between their corresponding symbols*. (G&K, 2010: 53)

A conception of the brain as a ‘collection of representing systems’, in the terms these authors present it, should give us the same effects for natural languages (or parts of them generated via L-grammars) and Fib (and pseudo-Fib, that is, strings that have the same quantitative properties as a true Fib string but have not been generated using the Fib L-grammar; see Patel et al., 2016; Krivochen & Saddy, 2016b) grammars. However, since the relevant properties (the Fib sequence, for instance) are not *in* the structure (i.e., in the development of the L-grammar itself) but are *read off* the structure as an *emergent* property, we cannot just aim at strict structure preservation *qua* isomorphisms, or think that it is the only mechanism in play. In fact, it might very well be that, if phrase structure rules are subject to semantic requirements (as in the system explained in Krivochen, 2015a), *structure preservation* is not even an issue as a condition over mappings (and even less so is *isomorphism*). For the time being, and for the purposes of the Fib-approach to language structure, let us say that, optimally, both structure and meaning (or configuration and properties of the elements involved in such configuration) are to be preserved under mappings *to the extent that the other allows it* (compare this informal proposal with Jäger's 2002: 435, ff. definition of *X-Optimality*). Such architecture seems more in tone with dynamical non-linear systems than with function-based, linear systems theory; we will analyse non-linear systems in detail below. However, for the time being, there is a more basic question to be answered: from an informational-theoretic perspective, do we *want*, as a theoretical assumption, neural computations to be digital or analogic? (and, more deeply, does this apparent dichotomy exhaust the possibilities?) This is a very real question, insofar as not only the theoretical consequences, but also the empirical predictions are significantly different in each model. G&K argue for a digital, function-based approach to computations which, as we will see when we analyse in detail each kind of formal automaton and the kind of languages each can generate and interpret, has far-reaching consequences for a theory of natural language and its neurocognitive substratum. A look at the mathematical modelling that would suit each option will shed some light on the problem.

1.7 Are cognitive operations digital or analogic? (And... is this the right question to ask?)

Shannon's (1948) digital-based information theory (a major reference for G&K's work, see, e.g., the discussion about information in pages 8-25) builds on clearly digital –discrete-, mechanistic assumptions not only with respect to the nature of the transmitting and receiving ends and the noise source, but also the mathematical tools to model such a system, and so should any proposal that wants to arise as an alternative. Formalizing computation as a dynamical process makes us face a choice of mathematical tools, which partly depend on the focus of one's inquiry. With respect to mathematical modelling for information theory, it would seem that there are two options:

- One option is to resort to *differential equations*: these equations typically model systems with a continuous independent variable (for Ashby, 1947, *inter alios*, the independent variable would be *time*) and its dependent variables, the equation establishing the relation between them given a rate of change which is hypothesized to be 'infinitesimal' (a term due to Leibniz, and not particularly cherished nowadays). The applicability of differential equations to the study of biological systems (and complex systems, more generally) is not free of controversy: while Ashby (1962: 261) claims that '(…) *a representation by differential equations is* […] *too restricted for the needs of a science that includes biological systems and calculating machines, in which discontinuity is ubiquitous*'; Breitling (2005) argues in favor of analyzing biological systems with differential equations, opposing the apparent non-linearity, multi-component nature, and feedback of biological systems to the linear, essentially limited nature of human 'thinking'. Curiously enough, no evidence (or theoretical argumentation) is presented in favor of the strong claims made for 'human thinking', whatever the author considers that to be, nor is it conceded that discontinuities in a function are *not always* problematic for differential equations, particularly if the discontinuity is removable[44]

[44] Consider, for instance, the following function:

$$f(x) = \begin{Bmatrix} x^2 \text{ for } x < 1 \\ 0 \text{ for } x = 1 \\ 2 - x \text{ for } x > 1 \end{Bmatrix}$$

The function is discontinuous at $x = 0$, but since the limits as $x \to x_0$ exist, from both the positive and negative sides, the discontinuity can be removed by defining x_0 as the limit and replacing that in the (sure, the function is not differentiable at the point at which it is discontinuous, but if the discontinuity is *removable*, either linear or quadratic approximations might do). Ashby points out some limitations of modelling using differential equations, but provides no alternative mathematical framework: he generalizes a matrix operation over finite-state automata to get a function-based 'machine' composed of a set I of inputs, a set S of internal states, and a function f mapping $S \times I$; not unlike Turing's formalism for α-machines, as we will see in **Section 2.1.4**. While *purely* information-theoretic arguments in favor of ordinary differential equations are, in our opinion, too restricted or sometimes even misled, so are many arguments *against* such dynamics.

- The second option is to resort to *difference equations* (often referred to as *recurrence relations*), which model discrete-state systems in which for any state S, S_n is defined as a linear function $f(S_{n-1})$ –let us leave the nature of f undefined for the time being-. The model should be familiar to the reader, as it is the basic template to generate the Fibonacci series under Lindenmayer sysmtems, for f a two-place addition function taking S_{n-1} and S_{n-2} as arguments (as well as other series, like Lucas', if a constant k is incorporated, see Uriagereka, 2015). Biology and data signal processing have taken advantage of recurrence relations, as they allow simple, step-by-step models of time-dependent processes, allowing for a neat mathematical description of derivations, if each derivational step (i.e., the application of a structure-building / structure-mapping operation) corresponds to a stage at time t. Formally, a derivation seen under a recurrent relations perspective would be defined as a set D = $\{(d_1, t_1), (d_2, t_2), \ldots, (d_n, t_n)\}$, where d is the relevant representation being manipulated at t_x, and the mapping between d_x and d_{x+1} is a linear function $f(d)$. For the sake of concreteness and simplicity, let us call that function *Concatenate*[45] (assuming, with Uriagereka, 2008; Stroik and Putnam, 2013; Krivochen, 2015a, b among others, that the property of displacement in natural languages is ultimately a context-sensitive interpretation of a structure building operation. We will return to structure mapping as an operation over fields in **Sections 3.11** and **3.12**). The range of possibilities is quite more restricted here, and it is often very difficult to account for the behavior of nonlinear dynamical systems by means of these relations alone. While the final object of a grammar based on recurrence relations might be very complex (say, a Mandelbrot set or a Sierpinski fractal), recurrence relations can only partially account for the step-by-step process.

Prima facie, it would seem that these options are mutually exclusive within a consistent model of mental computations: what is more, it *would seem* that the problem 'digital cognition vs. analog cognition' proposed, among others, by G&K (2010: 24) actually reduces to choosing between difference and differential equations for our mathematical model. However, things are not that simple. A common characteristic of both types of equation is a fixed input, and a function-based development (Barrett et al., 2006: 354), and it is not obvious whether either of them can be expanded to include dynamic inputs influencing the derivation in a non-aprioristic

function above. Thus, we get a $g(x)$, that equals $f(x)$ for $x \neq 0$ and L (the limit from the positive and negative sides as $x \to x_0$) for $x = 0$.

[45] Since all we want is an operation that 'puts things together', we might as well call it *Glue-α*, for α an *n*-ary set. The name is not important, as long as we bear in mind that we need *not* assume *headedness*, *binarity*, or *projection* (the X-bar axioms, inherited by Merge) to have a formally consistent and empirically adequate theory of language.

way, at least if implemented in a Turing-like α-automaton (we will come back to differential equations in **Section 3.8**, under completely different assumptions). Moreover, it is worth noticing that even Shannon's information theory must resort to the interference of factors external to the narrow computation (i.e., the transmitter-receiver narrow channel), since the noise source *can* be system-external[46]. In this sense, redundancy would be a way to counter-balance entropy, a way in which the system self-adjusts to optimize. Of course, we would not expect this kind of behavior in a closed, function-based model of computation implemented in an α-machine: at some point, there will be a choice to make (e.g., increase the attention threshold when the channel is too noisy, so that noise does not trigger unwanted responses): this amounts to saying there are elements from c-machines that are probably relevant for information theory. Of course, the extreme function-based proponent could resort to probabilistic / stochastic (i.e., non-deterministic) TMs, but there is actually no algorithm that, *ceteris paribus*, prevents a 'luckiest guess' situation when faced with multiple possibilities (unless assigning extra weight to a particular possibility from the beginning, but that is precisely the kind of situations contemplated by the *ceteris paribus* clause). It is essential to bear in mind that all TMs are equivalent with respect to *what* can be computed (there might be differences with respect to *how quickly*: the P vs. NP problem becomes relevant here), and we see no computational or neurocognitive advantages in adopting a non-deterministic TM uniformly. In a more general sense, we have argued in Krivochen (2015a; 2016c) (partially following and expanding on argumentation and data presented in Culicover and Jackendoff, 2005) against uniform templates for grammar assignment to a certain phenomenological input: our main claim in the present work is that the cognitive system (or, better still, each particular faculty, individually considered as the dynamic intersection of very basic and general cognitive systems) dynamically assigns a structural description to objects to be parsed (sensory stimuli, linguistic stimuli, and more abstract symbolic structures alike). That assignment, we have said, is not *a priori* defined as either the simplest or the most faithful, but results from the resolution between the tension (technically, the *frustration*) between those requirements, expressible in terms of economy of derivation and richness of representation respectively. Narrowing our scope down to natural language, we claim that it is not the case that a subject *recognizes* the grammar of a sequence, but rather, he assigns that sequence a structural description, building on information from more than a single source. Computational power, then, *might* correlate with countable potential (weakly) *generated sequences* (which is actually not clear, because even for the simplest computational system, a Finite-State Automaton, there is no upper bound on the length or number of weakly generated sequences, which is the whole point of the 'creativity' issue in Generative Grammar, see Chomsky, 1966; Behme, 2014, 2015a for discussion. Also, Langendoen and Postal, 1984, for formal discussion and a more dispassionate perspective), but it is not clear how that correlates with the *recognition* of sequences (be they base-generated or not): if any weakly generated language is an infinitely countable set of strings, then the *recognition power* of any automaton is an infinitely countable set. This said, it is true that some infinite sets can subsume others (as proven by Cantor's diagonal argument, and illustrated by Hilbert with his Grand Hotel paradox), but this has not been an issue in the mentioned references, or in the MGG literature about generative power in general.

G&K (2010: 24) assume that information transmission and processing in the brain is ultimately *digital*, because, they claim, this kind of computation optimizes noise reduction,

[46] A source of noise is said to be 'system-external' when the noise is neither a property of the code itself nor is it a property of the system. This is the case if we are dealing, for instance, with an open nonlinear dynamical system in which there can be positive feedback when presented with a perturbation.

efficiency of transmission, and control. Information is, for them, essentially digital (no continuity or dis-continuity in information transmission, but a whole lotta discreteness), and the probabilistic model associated with the transmission of that information is *function-based*, with the function of the posterior probability of a state of the world holding being directly proportional to the product of two other functions, namely, prior probability[47] and likelihood[48]. This assumption makes their thesis stronger, since not only is the processing and transmission of information in the brain *digital*, it is also strictly *function-based* in its probabilistic aspect. Taking into account that functions are by definition linear in the application of derivational rules, and closed to external influence, this model is compatible with orthodox assumptions in MGG, from Chomsky (1965) onwards, and also further developments within non-transformational theories (e.g., Kaplan and Bresnan, 1982; Green, 2011; Pollard and Sag, 1997). G&K assume an isomorphism between digital computers and minds, such that, since digital computers carry out tasks efficiently, there is no need for a theory of analog computation, which, in their terms, might not even accomplish everything digital computation (which is function-based) can. That is, in their view, analog computation (or, more generally, non-classical computation) constitutes a subset of digital computation in terms of strong and weak generative power (but see Lasnik and Uriagereka, 2012 for a generative perspective that rejects classical discrete computation as the model for operations that are taken to be 'psychologically real'). However, efficiency, non-redundancy, uni-directionality,… are properties of a formal model, not necessarily of the modelled object itself.

Part II: Functions, Formal Languages, and Automata Theory

2.1 Automata and computation

A proper reworking of the concept of 'grammar' needs to take into consideration the analysis of artificial languages and automata theory, which have been used as a model for the analysis and modelling of natural languages particularly since the 1950s. We will devote this section to expanding on formal languages and their relation to automata, an essential point in understanding the function-digital approach to computation, its strong points and, crucially for our own ends, their limitations both theoretical and empirical. For the sake of clarity, as well as methodological rigor, we will follow the standard practice in computer science of defining automata as *n*-tuples, which will allow us to compare the properties assigned to each automaton. In the presentation, we will proceed in the order assumed in the Chomsky Hierarchy (Chomsky, 1959: 143), whose applicability to natural languages (and even some formal languages) we will then criticize (along the lines of Krivochen, 2015a, 2016b, c; Bravo et al., 2015, among other works) on both theoretical and empirical grounds.

[47] The prior probability term usually does not entail complete uncertainty (as also noted, in a different context, by Spivey, 2007), since, as G&K acknowledge, we use information from different sources to obtain evidence with respect to the likelihood of a particular event. However, this resort to information from different sources is not, in principle, compatible with the essentials of function-based computation, unless it is specified in the finite input set. Once it is conceded that more than a single source of information influences the computation at the input end, it is difficult to prevent multiple sources of information (system-external) from influencing the computation *during* the derivation, unless by resorting to some stipulation (e.g., MGG's *Inclusiveness Condition*).

[48] *Probability* is used when *describing a function of the outcome* given a fixed parameter value. Conversely, *likelihood* is used when *describing a function of a parameter* given an outcome.

The basic *formal* elements all automata share are an *alphabet* (Σ), a set of states (including finite sets of allowed *initial* and *final* states), and a transition function to proceed from state to state, usually some sort of *mapping* function, although other kinds of relations can also hold between states (e.g., mere succession, as in an automaton generating the set of positive integers ℕ via Peano's axioms). For the sake of uniformity, we shall consistently use S_0 as an axiom when exemplifying a formal grammar, and → to symbolize the rewriting of the symbol on the left as the string of symbols on the right (i.e., X → Y is to be read as 'rewrite X as Y'), also known as the 'transition function' or 'mapping function'.

Automata are made of (at least) two *functional* elements: a moving head, which can just read, or read *and* write, and a (Long Term) memory tape, on which reading and writing operations take place. Differences arise between automata with respect to their ability to write, and the space within the tape that they can access at a particular derivational step: it is important to point out at this stage that the *time* variable is usually not taken into account in formal automata theory, and thus time might be defined just by the function relating subsequent states of the automaton[49]. Another important issue is that, for the time being, we will not require that our grammars should ever *halt* (Post, 1936; Turing, 1936): we will later see that there are natural ways to implement periodic halting as an emergent behavior within a system given certain conditions upon its physical instantiation. Moreover, we will try to unify the terminology in our presentation, such that the comparison between automata is more straightforward, even if that means changing the notation used in the original sources. When this happens, we will make the necessary clarifications.

2.1.1 Finite State Automata

A *finite state automaton FSA* over an alphabet Σ is a 5-tuple $M = (S, \Sigma, \delta, s_0, F)$, where S is a finite, non-empty set of states, Σ is a finite input alphabet, δ is a mapping function of $S \times \Sigma$ into S (that is, every application of the mapping function results in a state of the system); $s_0 \in S$ is the *initial* state, and $F \subseteq S$ is the set of *final* states. The alphabet is not fixed *a priori*, but it is customary to use the binary set {1, 0} in computer science for illustrative purposes. Here, we will replace *numerical terminals* with *lexical items*, somewhat following Chomsky (1957) and related work. In the context of this work, *alphabet* and *lexicon* will be used interchangeably.

A string x is said to be accepted by M if $\delta(s_0, x) = p$ for some state $p \in F$. The set of all x accepted by an FSA (thus, the *language* that an FSA can generate and manipulate) is designated as *T(FSA)* (Hopcroft and Ullman, 1969: 23; see also Martin, 2010: 54). That is,

33) $T(FSA) = \{x \mid \delta(s_0, x) \in F\}$

Any set of strings accepted by a finite automaton is said to be *regular*, and T(FSA) is called a *regular language*. Let us exemplify what a non-deterministic Finite-State Grammar FSG looks like, for Σ = {A, B, a, b}:

34) S_0 → A

[49] Such a conception makes it difficult to implement these automata in a brain, since mental processes are time-dependent. However, it is possible to model time in formal automata theory by means of recurrence relations (i.e., difference equations), and add a constant *t* to the automaton's definition, denoting the *time lapse* between subsequent states. Of course, this entails that any transition between any two subsequent states will take the same amount of time, which is not, in our opinion, a good approximation of actual neurodynamics.

A → *a* B
A → *a* A
A → *a*
B → *b* A
B → *b* B
B → *b*

A sample string *x* from this language (omitting nonterminals, as is customary) is the following:

35) *x* = *aabbaaab*…

An FSA can be graphically represented as a *chain* of states, such that there are two possible relations between states: deterministic and probabilistic. However, since every sentence accepted by a probabilistic (or non-deterministic) FSA is also accepted by a deterministic FSA (a proof we will not get into here, but see Hopcroft and Ullman, 1969: 31-32; Martin, 2010: 106-107), the label 'FSA' is used to denote both kinds of automaton. A binary Σ finite state chain (Σ = {a, b}) for (35), including state change and loops (and assuming all rules have an equal probability associated to them), looks like this:

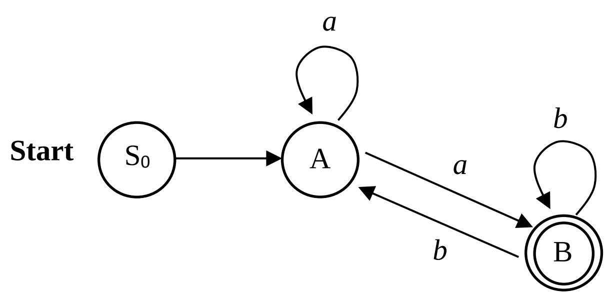


Figure 2: Finite State Automaton

Finite state automata are known to have the so-called Markov Property: they have no memory. That means that their search space is limited to the current state: only one symbol is read at each derivational step and there is no access to previous states. In more technical terms, stochastic processes (like probabilistic FSAs) are said to have the Markov property if the conditional probability distribution depends exclusively on the 'present' state of the system, with no access to previous states (or the ability to *look ahead* to future states, that is, no access to the {*F*} set). Since probabilistic FSA (e.g., that in **Figure 2**) are stochastic, but can be subsumed to deterministic FSAs, then the whole reasoning is abbreviated to saying 'finite state automata have no memory'. Moreover, finite state-compatible strings, that is, strings belonging to a *regular language* display strictly sequential dependencies, without the possibility of relating states s_x and s_y if there is a state s_z such that $\delta(s_x) = s_z$, since the search / probing space is limited to the present state. The active read-write head customarily moves only from *left to right*, except in the case of so-called 'two way-FSAs', which are necessary if there is an instruction that takes us back to the initial state (particularly if the set of allowed states *S* is a small set, as in **Figure 2**). It is important to notice that more 'freedom of movement' does not impact at all in either active memory or acceptable strings, since every string that can be modeled by a two-way FSA can also be modeled by a one-way deterministic FSA. That is, all FSAs are equivalent, insofar as they accept the same set of sentences.

With respect to the acceptable strings, all finite sets are acceptable by (i.e., can be modeled by) a FSA. In formal language theory, generation and recognition are not separated, but if we are dealing with *natural* languages, the distinction becomes relevant. In natural languages, there are sets of finite strings which cannot be *generated* by an FSA: finite strings with discontinuous dependencies are one example (as Chomsky, 1957 noted; we will come back to this briefly). In the realm of formal languages, strings of symbols derived using the Fib-grammar are another example of finite string that resists FSA computability; it is not possible to create a FSA that will generate all and only Fib-strings. However, and against what is now linguistic common sense, these limitations do *not* entail that Markov chains should be completely abandoned as models for *some* finite (sub-)strings, as we will briefly see below (we refer the reader to Lasnik & Uriagereka, 2011; Krivochen, 2015a for extensive discussion). As a matter of fact, we will see, they are quite useful, *locally*.

Within the theory of syntax, finite state models have had quite an impact, both before and after Chomsky's (1956, 1957) case against finite state descriptions for natural languages. The inquiry about the nature of the computational system underlying human language was somewhat revived, within MGG, by Howard Lasnik and Juan Uriagereka. Lasnik (2011) addresses Markovian properties of 'lower level syntax', which had already been spotted by Chomsky and Miller (1963), under the rubric 'true coordination':

36) The man comes/The old man comes/The old old man comes

For sentences like (36) (which kind of sentence we discussed above when referring to the *structure assignment* vs *structure recognition* approaches), a phrase structure grammar of the kind [Σ, F] (Σ a finite set of initial strings and F a finite set of rewriting rules) like the one argued for in Chomsky (1957) -to which we will come back below-, imposes too much structure over adjective stacking, assuming a uniform system of binary projection in which notions like 'c-command' and 'domain' are entailed by the axioms of phrase structure grammars within MGG. Chomsky himself (1963: 298) recognizes the structural overgeneration of phrase structure grammars (or, in terms of minimal description length, the excessive complexity of phrase structure descriptions):

> *a constituent-structure grammar necessarily imposes too rich an analysis on sentences because of features inherent in the way P-markers are defined for such sentences.*

Curiously, there was no attempt within MGG to improve the phrase structure engine in order to include finite-state dependencies (e.g., by making it more dynamic or semantics-sensitive), but these examples were increasingly overlooked from the '80s on: the rise of X-bar structure (Chomsky, 1981a; Stowell, 1980), and its *binary branching* requirement (Kayne, 1984) directly excluded finite-state dependencies from the picture. Minimalism's operation 'Merge' (Chomsky, 1995) maintained the essentially binary character of phrase structure markers, which was 'deduced' from apparently independent principles like antisymmetry (Kayne, 1994, 2011; Di Sciullo, 2011) and feature valuation operations (Pesetsky and Torrego, 2007; Wurmbrand, 2014).

Following up on Chomsky (1963) and Postal (1964), Lasnik (2011) acknowledges the problem imposing 'too much structure' on structural descriptions for strings if a *uniform* 'moving up' in the Chomsky Hierarchy is performed (that is: 'FSGs are inadequate for some substrings, then we proceed to CSGs; these also have limitations, thus we go further up…'). Little –if any- attention has been paid to such warnings. Even after acknowledging the problem of imposing extra structure on a syntactic object, Chomsky and Miller (1963: 304) insist on the binary character of phrase markers, making it a part of the definition of *generalized transformation* (Chomsky, 1955, 1995: 189; Kitahara, 1994: 27):

> *The basic recursive devices in the grammar are the generalized transformations that produce a string from a* ***pair*** *of underlying strings.* (Chomsky and Miller, 1963: 304. Our highlighting)

Lasnik proposes to weaken the binarity requirement, which is also a huge part of the definition of *Merge* (Chomsky, 1995: 189; Kayne, 1994) so that the structure building algorithm incorporates a generalized transformation mechanism that maps *n* phrase markers into a new phrase marker (see also Chomsky and Miller, 1963: 299), which would yield the required flatness for Markovian dependencies. Along the same lines, Lasnik and Uriagereka (2012) summarize the computational properties of the grammars in the Chomsky Hierarchy while acknowledging that portions of natural languages display Markovian behavior (see also Lasnik, 2011), and thus we must require of a psychologically implementable grammar to be able to provide us with 'dynamic flatness', which is '*incomprehensible in a classical computational view*' (Lasnik and Uriagereka, 2012: 21)

> *In a manner of speaking, what we really want to do is* ***move down*** *the* [Chomsky] *hierarchy. Finite-state Markov processes give flat objects, as they impose no structure. But* ***that is not quite the answer either****. While it would work fine for coordination of terminal symbols, phrases can also be coordinated, and, again, with no upper bound.* (Lasnik, 2011: 361)

The possibility of going up and down the Hierarchy in local domains follows straightforwardly from a mixed approach to linguistic computations of the kind we will develop here and have presented in Krivochen (2015a, b, 2016d), Bravo et al. (2015); García Fernández et al. (2017), and Krivochen and Schmerling (2016). The kind of computational system that can implement such 'dynamic flatness' cannot thus be uniform: here lies one of the crucial aspects of the present work.

Assuming a strong computational uniformity thesis for natural language strings, Chomsky (1957: 21) claims that it is *impossible* for a Markov system to produce *all and only* the grammatical sentences of the English language (a clear constructivist desideratum, see Lasnik, Uriagereka & Boeckx, 2005 for developments of Minimalism under constructivist desiderata). Thus, he arrives at the following conclusion:

37) *English is not a finite state language* (Chomsky, 1957: 21. Ex. 9)[50]

He provides examples of discontinuous dependencies like the following as evidence of his claim (1957: 22):

38) a. If S_1, then S_2
    b. Either S_3 or S_4

[50] Cf. Kornai (1985: 1) for the converse thesis: "*natural language stringsets are regular* [i.e., Finite-State]". Unlike Chomsky, Kornai draws arguments from language data and acquisition, apart from considerations regarding theoretical simplicity. While his bases are wider, and more thoroughly argued for than Chomsky's, the selection of data regarding nested dependencies is limited, and no evidence is provided regarding discontinuous dependencies (as in Wh-interrogatives or parasitic gaps / multiple gap constructions).

c. The man who said that S_5, is arriving today

We have to disagree with (37), even in the light of examples like (38) including discontinuous dependencies of different kinds (*if-then*; *either-or*; *the man-is*). All that the discussion in Chomsky (1957) leading to (37) proves is that *some portions* (i.e., substrings) *of the English language* (considered as a set of well-formed formulae, as was the use in the late '50s, particularly within transformational grammar) are not generable by a finite state grammar, when a dependency between α and β is not established via adjacency and thus requires some *memory* -which systems displaying the so-called 'Markov property' do not have-. Chomsky's claim, clearly, is a plea for structural uniformity (i.e., the generative engine can produce only one kind of dependency) which is, we think, a stipulation over the limitations of the automaton in which a formal procedure is implemented: if several grammars (read: interpretative procedures) are available (from finite state grammars to mildly context-sensitive grammars, in the case of natural languages), why limit the generative-interpretative capacities to a single step in the hierarchy if such limitation is neither required nor supported by the data[51]? Note that Chomsky's concern does not pertain to whether Type 3 grammars can generate structural descriptions for natural language strings (because they can), his concern pertains to the *adequacy* of said structural representations (which is, coincidentally, also our main concern). We will address this issue in the remaining of this work, since it is an essential part of our 'grammar assignment rather than grammar recognition' theory, particularly within the 'mixed computation' approach we have developed in past works.

2.1.2 Push-Down Automata

A Push-Down Automaton (PDA) over Σ is formally defined as a 7-tuple: $PDA = (S, \Sigma, \Gamma, \delta, s_0, Z, F)$, where *S* is a finite set of states, Σ is a finite alphabet, Γ is a finite stack alphabet set (i.e., an inventory of symbols that go on the tape for storage-retrieval), δ is the transition relation between states, $s_0 \in S$ is the *initial state,* $Z \in \Gamma$ is the initial *stack symbol,* and $F \subseteq S$ is the set of *final* states. A schematic version of the automaton is the following (taken from Lewis and Papadimitriou, 1998: 130):

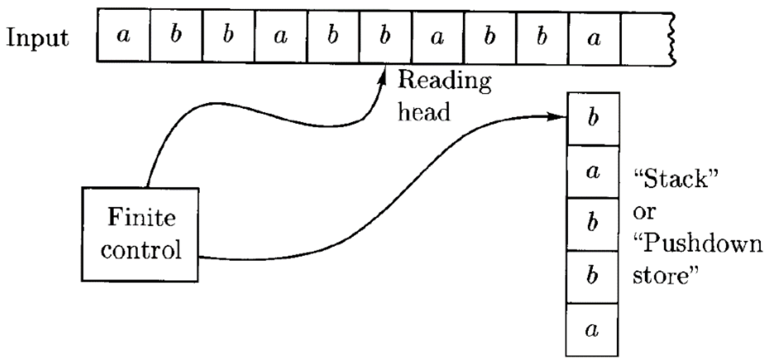


Figure 3: Push-Down Automaton

[51] This problem holds for uniformity proposals in general, of the kind 'language as X [and not as Y]', where X and Y are specific kinds of computational devices (for more examples, see Nordström, 2013; Watumull et al. 2014). All these proposals (Chomsky's, Kornai's, Nordström's, Watumull et al.'s…) make a case based on limited data, a specific set of examples which is particularly problematic for some other equally procrustean theory or set thereof. What we would want to advance here (as we hope to have done in Krivochen, 2015a; Bravo et al., 2015; García Fernández and Krivochen, 2018; Krivochen and Schmerling, 2016) is that there is no need to choose *either-or* if we let the computational substratum of cognitive capacities be *non-uniform* (a.k.a., 'mixed').

Unlike FSAs, PDAs can accept not only sentences of a regular language, but also sentences of non-regular context-free sentences. That is, PDAs can accept sequences of the kind AB^n (acceptable by finite-state automata), as well as A^nB^n, which, by virtue of being non-regular, cannot be accepted as strings for a FSA (see Martin, 2010; Hocroft and Ullman, 1969: Chapter 5) The kind of languages accepted by PDAs require some memory, which is defined for a PDA as the most recently stored string of symbols (Uriagereka, 2008: 226-227; 2012: 230-231) and which can establish nonlocal dependencies as long as they are not crossing dependencies: the *search space* for a PDA is thus equal to the length of the stack input at a derivational point D_x, which 'stacks' over previously stored information. This system of stacking implies that the last material stored is used first, as it is the most accessible element in memory (see Medeiros, 2015a, b for an application of a stack-sorting architecture). Such a conception is compatible with phrase structure grammars, and captures certain notions of impenetrability that are common in current syntactic theorizing (e.g., the notion of *phase*, see Chomsky, 2008; Gallego, 2012), in the sense that earlier inputs are not available once a new symbol has been introduced (in consonance with 'every phrase is a phase' proposals, see Epstein and Seely, 2002, 2006). A PDA is thus strongly derivational, in the sense that, when implementing a [Σ, F] grammar, the terminals that constitute the rightmost part of the rule *replace* the previous symbol, as in the sample (stochastic) context-free grammar in (40) –from Krivochen and Matlach (2015)-, which displays center-embedding for all rewriting possibilities (cf. the FSG in (35) above):

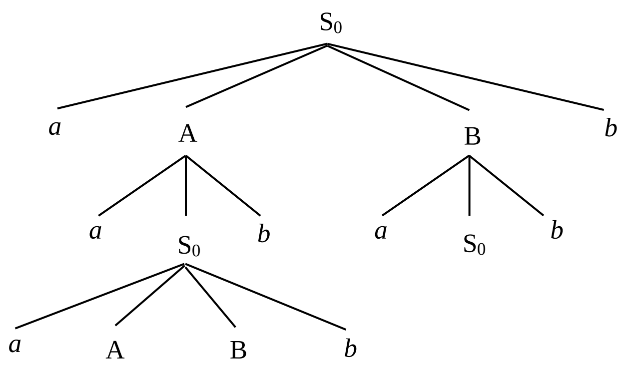
39) $S_0 \rightarrow a\ A\ B\ b$
$A \rightarrow a\ S_0\ b$
$A \rightarrow b\ S_0\ a$
$B \rightarrow b\ S_0\ a$
$B \rightarrow a\ S_0\ b$

Therefore, the following is one of the possible generation sequences for a word in the language:

40) $S_0 \rightarrow a\mathrm{AB}a \rightarrow aa\mathrm{AB}bb \rightarrow a^n\mathrm{AB}b^n$

In the more familiar tree-form:

41)



Each level of branching represents a derivational step, that is, a *push-pop* cycle. As the reader can see, the kind of language generated here is an A^nB^n language (i.e., same number of [a]'s and [b]'s in the final string, regardless of the stochasticity of the grammar we have presented), and it could not be accepted by a FSA insofar as the rules involve both terminals and non-terminals, which implies maintaining a unit in the stack for the purposes of further computations. Let us see a natural language example of the kind of dependency this model allows, taken from MGG literature:

42) The cheese the mouse wanted rotted

The structure of (42) is (43):

43) [*a* [*a*’ *b*’] *b*]

The grammar is slightly different from that presented in (39) (since (39) is a probabilistic CFG), but they are formally equivalent, and thus both equally acceptable by a PDA. Interestingly, the sentence in (42), adapted from from Berwick et al. (2014), displays both center embedding (which can also be seen in rules 2-5 in (40), which include the axiom S_0 as part of the right side of the rule) and a long-distance dependency across the embedded unit thus requiring some inner probing into the memory tape (that is, having access to certain portions of the tape from either the outside or other parts of the tape, and operating over those specific parts of the tape), see what has been written, and act accordingly[52]. Such inner probing is arguably beyond the possibilities of a strict PDA, since by the time *b* (agreeing morphologically with *a*) is inserted in the stack, *a* is no longer accessible, recall the automaton works with a simple last-in-first-out kind of memory: it is thus difficult to see how a dependency (e.g., involving morphological agreement) can be established between two mutually inaccessible elements (as they are *never together* in the active space) if not by resorting to further levels of representation or interpretation beyond the definition of the automaton.

Lewis and Papadimitriou (1998: 136) proof the theorem that ‘*the class of languages accepted by pushdown automata is exactly the class of context-free languages*’ (CFL) (also Hopcroft and Ullman, 1969: 70). We will not delve into the mathematical proof here, as we are just interested in pointing out that the class of context-free languages *can* be put to exact correspondence with the set of sentences generated by the Base component in a Generative Grammar, which a PDA can therefore accept; as well as with the bottom boundary of Tree Adjoining Grammars (TAGs henceforth, see Joshi, 1985: 211). Let us elaborate a bit on this. The Base Component of a generative grammar is a triple BC = (Σ, *S*, δ), where Σ is the alphabet of terminals, call it the *Lexicon*, *S* is the set of possible non-terminal states, and δ (the *transition relation*) is a finite set of *Phrase Structure Rules* (PSR) of the kind Σ → F. Consider the following set of PSR (from Chomsky, 1957: 26):

44) (i) Sentence → NP + VP
(ii) NP → T + N
(iii) VP → Verb + NP
(iv) T → the
(v) N → man, ball, etc.
(vi) Verb → hit, took, etc.

The PSR above can be implemented by means of a PDA, and the sequence [the man hit the ball], derived via successive applications of δ as follows (we follow Chomsky’s 1957 derivational model here) can be accepted by a PDA:

45) Sentence (Axiom)
NP + VP (by (i))
T + N + VP (by (ii))

[52] It is possible to encode any possible long distance dependency in the form of a specific PSR, but that implies an exponential growth of the grammar. We will not even consider this possibility, as it would be intractable, and of course impossible to implement in a cognitive system with finite resources.

T + N + Verb + NP (by (iii))
The + N + Verb + NP (by (iv))
The + man + Verb + NP (by (v))
The + man + hit + NP (by (vi))
The + man + hit + T + N (by (ii))
The + man + hit + the + N (by (iv))
The + man + hit + the + ball (by (v))

Such a derivation can be carried out by a PDA, since only a *single* terminal is 'pushed' in the stack at each derivational point (the first to which a rule can be applied, going from left to right), following a strict top-down order (from rule (i) to rule (vi), sequentially). This grammar can generate infinitely many (structural descriptions of) sentences (just like a finite-state grammar can), but, crucially, it cannot generate *all* the structural descriptions of a relevant language. More concretely, it is bound by its limited memory, which does not allow it to establish dependencies that are not extremely local (where 'extremely local' is defined as 'next symbol in the stack')[53]. Consider then a case like (46):

46) John wants to buy himself a car

where there is a referential dependency between the terminals [John] and [himself] (usually indicated by means of *coindexation*), but, crucially, when [John] is pushed in the stack, [himself] is no longer accessible. Cases of *displacement* make this limitation in the search space even more obvious:

47) Which car did John say he would like to buy himself?

assuming there is some syntactic-semantic dependency between the object position of [buy] (call it an empty terminal Ø –frequently, the symbol ε is also used in automata theory, particularly when applied to computation, but we will make a difference here: ε is a blank symbol, whereas Ø merely has no phonological exponent-) and the non-terminal [which car] (see Krivochen, 2015b for discussion and alternative approaches to such dependencies). There are operations, like Generalized Transformations (Chomsky, 1957: 113; Kitahara, 1994), which allow us to replace a terminal with a non-terminal in a Chomsky-normal grammar, therefore, the replacement Ø → *S* should not be an obstacle for building the relevant structure. However, that replacement is not possible due to memory limitations, given the stack-based architecture of PDAs. Moreover, if a transformational analysis is assumed (which is *by no means* a necessary condition to account for this kind of 'displacement' effects, as the analyses in Gazdar, 1981; Kaplan & Bresnan, 1982; Pollard and Sag, 1997; Culicover and Jackendoff, 2005; the studies collected in Borsley and Börjars, 2011; Krivochen, 2015b; Kosta and Krivochen, 2014a, *inter alios*, show), the computational power of a PDA is more manifestly not enough to manipulate

[53] Given a derivation as just presented, it could be argued that the derivation in (46) can also be carried out by a FSA, since we appear to have a sequence of states. That, however, is actually a problem of the early Chomskyan representations (a problem carried over time), which were insensitive to constituency (see Wells, 1947): as Uriagereka (2002a) correctly points out, while strictly monotonically assembled structures are indeed exhaustively modelable by means of a Markov model (see also Hughes, 2004 for a perspective from graph theory), here we have two Markovian sub-derivations: [the man] and [hit the ball]. See Uriagereka (2002a, 2012) for details and discussion of a Multiple Spell Out model based on the concept of materialization as dynamic Markovization of structure in order to make tree graphs LCA-compatible.

symbols in the desired manner, such that the relevant *displaced constituent-base position* dependencies can be established. A further, Transformational component that maps phrase markers (nonterminals) is required if a Phrase Structure Grammar (PSG) of the kind in (45) is assumed, and with this Transformational component, we must also assume a more powerful automaton than a PDA. It is crucial to notice that the mere addition of a transformational component is neither necessary nor sufficient to satisfy the initial goal of generative grammar, that is, provide a structural descriptions of all and only the grammatical strings of L: on the one hand, non-transformational theories which include context-sensitive interpretative procedures (and even more powerful routines, like *interactive* procedures of the kind we will analyze below) can strongly generate languages outside the recognition limits of PDAs; on the other hand, Tree Adjoining Grammars (TAGs), by means of an *adjunction* operation[54] and the addition of *links* which are preserved after adjunction, are also capable of providing structural descriptions for strings displaying dependencies beyond the recognition capabilities of CF-PSG like G/HPSG (that is: there are strings which are TAG-recognizable but they are *not* CF-recognizable). The question is, then, *to what extent* 'natural languages' (if understood as recursively enumerable sets of strings, as Chomsky does) are *context-sensitive*, under a traditional CTM view.

2.1.3 Linear Bound Automata

Linear Bound (or bounded) Automata are essentially non-deterministic Turing Machines (to which we will return in **Section 2.1.4**), with contextual conditions over operations in the memory tape (Kuroda, 1964: 209). Formally, a linear bounded automaton (LBA) is denoted by $M = (S, \Sigma, \Gamma, \delta, s_0, F)$, where S is the set of states, $\Sigma \subseteq \Gamma$ is the set of *input symbols*, where Γ is the full set of tape symbols. s_0 is the *initial state*, $F \subseteq S$ is the set of final states, and δ is a transition function from $S \times \Gamma$ (Hopcroft and Ullman, 1969: Chapter 8; Kuroda, 1964 and references therein). The peculiarity of LBAs is that Σ contains two special symbols, usually denoted by ¢ and $, which are the *left* and *right* endmarkers, respectively (monetary metaphors aside). This means that no transition (i.e., no application of δ) can go further to the left than ¢ or further to the right than $, or print symbols over the endmarkers. This condition restricts the power of LBAs with respect to Turing Machines, which will be our next immediate concern. PSR for these automata generate what is called a *context-sensitive language* (CSL) (as shown by Landweber, 1963), since rewriting rules are sensitive to left / right boundaries, and there are distributional constraints over symbols, as in (48) (a generic CS rule):

48) $s_x \rightarrow s_y \in \mathrm{S}\ /\ W\_Z$

The distributional condition can be seen as an *iff* clause (i.e., proceed from state *x* to state *y* if and only if the distributional constraints obtain). It is quite straightforward why CSL properly contain CFL: CFLs rewrite *without contextual constraints*, which is translatable into CSL as an 'elsewhere-case': a CFL can simply be described as an 'elsewhere case' of a CSL, in which a

[54] Such operations are similar (but not identical) to a *generalized transformation* (Chomsky, 1955, 1995; Kitahara, 1997) including a condition like Uriagereka's (2012) *Address Issue* that links separate graphs by means of a node present in all of them which is like a 'hinge' where sub-derivations connect; making the operation mildly context-sensitive (generating a proper subset of CS languages). When *adjunction* applies only at the frontier of a tree, it is merely *substitution*, and in that case its generative power is equivalent to a grammar with *generalized transformations*. *Adjunction* will be invoked again when discussing Medeiros' (2015a, b; 2016) QLSR architecture, and we will return to the Address Issue in terms of fields and their interference in **Section 4.5** below.

rewriting rule for nonterminal N is not contextually conditioned or, equivalently, is constrained by ∀(*s*). Graphically, an LBA would look as follows:

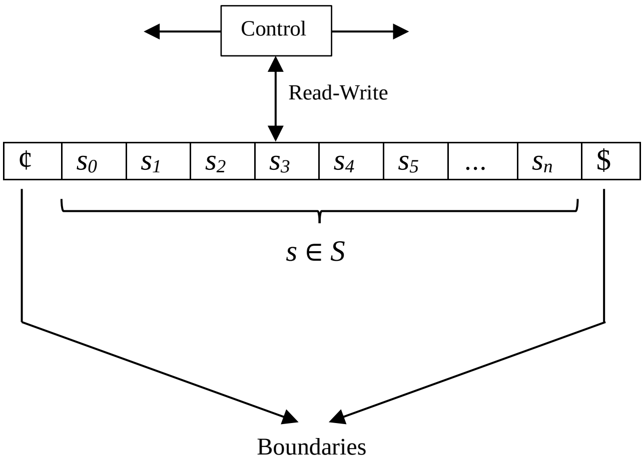


Figure 4: Linear-Bound Automaton

LBAs differ from PDAs in terms of search space as well: while, for any input *i*, the search space of a PDA is limited to *i* (the last element to be pushed onto the stack), for LBAs the search space is defined as a linear function over *i*, call it *f*(*i*), such that they can probe anywhere within the memory stack within the limit imposed by the endmarkers (see Uriagereka, 2008: 227-228), thereby allowing not only *head-tail* recursion and *center embedding* (available in FSA and PDA respectively), but internal manipulations and cross dependencies, that is, *free embedding*. Let us give the reader an example of a stochastic (and admittedly complicated) CSG (from Krivochen and Matlach, 2015):

49) $S_0 \rightarrow a\ b$ A b B
A → A *b b* B
A → *b*
B → *b a*
B → *a b* B S_0 A *b* A
a A *b* → A A A [note that this is equivalent to the notation A → A A A / *a* __ *b*]
b A *b* → B *b* S_0 *b*
A *b* → B *a b* B
a B *b* → S_0
B *b* → A *b a* B S_0
b S_0 *a* → B

It is relevant to point out that there are natural language strings which have been argued to be *at least* context-free, more specifically, *mildly-context sensitive* (see Joshi, 1985: 213 for an argumentation within the framework of TAGs). Joshi's original argument is that the grammars that can account for (i.e., generate adequate structural descriptions for) natural languages are slightly more powerful than CFG, but not as powerful as CSG; consequently, the languages they account for are beyond the limits of CFL, but only barely within CSL (most likely due to the limited, but existing, possibility of long-distance dependencies, if we consider memory issues). Mild-context sensitivity comprises a number of non-trivial properties (see also Stabler, 2003; Joshi, 1985: 221, ff.; Joshi & Schabes, 1991):

50) a. Polynomial parsing complexity (for TAGs, it amounts to O(n^6) in the worst case, see Joshi and Yokomori (1983); Kroch & Joshi (1985: 25)[55].
b. Constant growth (for any string w, $|w|$ grows by a constant factor, which is a linear combination of the length of the terminal string of some initial tree and the lengths of the terminal strings of the auxiliary trees)
c. Limited crossing dependencies (by virtue of a limited active search space, restricted to sub-trees or linked nodes in an annotated TAG)
d. CFLs are strictly included in Tree Adjoining Languages (TALs), which themselves are strictly included in *indexed languages*. All closure properties of CFLs, furthermore, hold for TALs.
e. TAGs can generate the context-sensitive language $\{a^n, b^n, e, c^n, d^n \mid n \geq 1\}$ via *adjunction* in an annotated TAG with links (which are preserved throughout the derivation), but *not* the context-sensitive language $\{a^n, b^n, c^n, d^n, e^n \mid n \geq 1\}$

(50 a) is an essential condition for the implementation of the formal system: we expect the formal model to be able to parse sentences in polynomial time (upper bounded, as the worst case scenario). It is not clear, to the best of our understanding, whether natural languages would indeed fall into the category P (in informal terms, 'quickly solvable') or NP (informally, 'quickly checkable'), particularly given the fact that it is still an open question in computer science whether P = NP (= NP-Hard) (that is, whether, if the solution to a problem can be quickly *verified* by a TM –in nondeterministic polynomial time- the TM can also *solve* that problem quickly–in deterministic polynomial time-); notice, however, that if Joshi's theory is indeed correct, natural languages would satisfy P, but not necessarily NP. There are problems with the consideration of polynomial time in the picture, as it has been objected that *Cobham's thesis* (which relates polynomial time to 'easy' computational tractability) does not take into account the typical size of the input or external variables that can affect it, the thesis thus being restricted to closed, serial, digital computation. However, it might be the case that under a different view of computation in which natural languages are not function-based (and thus, the 'big omicron' notation for calculating growth rate for a function does not apply globally, although it might apply locally, within *cycles*), the problem does not even arise, a possibility we will explore below, after dealing with Turing machines and the arguments in favor of a 'Turing program' for linguistic theory (as argued in Watumull, 2012; Watumull et al., 2014).

2.1.4 Turing Machines

TMs are formally defined as a set $M = (S, \Sigma, \Gamma, \delta, s_0, F)$, just like LBAs, but without the (optional) {¢, $} distributional constraints or any limitation on search space within the memory tape (Hopcroft and Ullman, 1969: 80; Martin, 2010: 224, ff.). TMs have an unlimited memory tape, and can probe anywhere within that tape, which makes them more powerful than LBAs in terms of the kind of dependencies they can establish, having *no probing memory limitation*. The basic structure of a TM consists of an infinitely long memory tape divided in a countably infinite number of squares, each of them capable of bearing a single symbol; within which a read-write head (a 'finite control', like PDAs and LBAs) can probe anywhere, and a set of rules

[55] A simple (perhaps even simplistic) formulation of 'big omicron' notation for function growth is the following: $f(n) = O(g(n))$ for sufficiently large n *iff* there is a constant $c > 0$ such that $f(n) \leq c \times g(n)$. See Knuth (1976) for details.

or instructions of the kind ‘If X, do Y (and proceed with Z)’. A graphical representation of a (right-infinite) TM is as follows (taken from Lewis and Papadimitriou, 1998: 181):

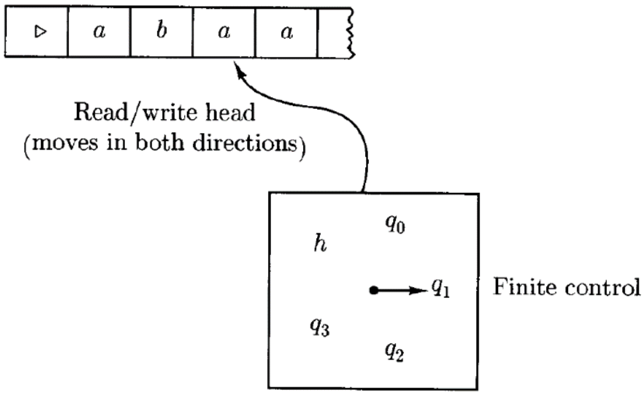


Figure 5: Turing Machine (here, *q* is what we have notated as *s*, possible states of the machine. *h* stands for ‘halt’, an instruction to stop the computation)

At each derivational step, then, a TM:

- Changes state
- Prints a non-blank / blank symbol on the ‘scanned symbol’ square
- Moves its head left / right
- Proceeds with another instruction

A computational system so specified is usually referred to as a *deterministic Turing Machine*, which means that at each derivational step there is only one path to take and one tape to work on. However, this is not always the case. Not only can we have multiple tapes in a TM (a so-called ‘Multi-track TM’ which, crucially, does not change *what* it can compute, but perhaps *how fast*), but, more importantly, we can have two (or more) valid possibilities at each step, say:

- Write 1
- Write 0
- Write Ø (or, simply, ‘erase’)

A TM that chooses between the available transitions at each point according to some probability distribution (which can be random) is commonly called a *probabilistic TM*, which is actually an ‘umbrella term’ for a number of theoretical computing machines, including *stochastic TMs, quantum TMs* (see, e.g., Deutsch, 1985), among others (single and multitape), all depending on the probability amplitude *p* (for $0 < p < 1$) assigned to each option. It can also be the case that the machine does not have enough data to proceed with an instruction unambiguously, as Turing himself recognized (1936: 232), and thus needs the input of an *external operator*. Later theoreticians have preferred to either assign a higher probability value to one of the options by tweaking the parameters in several ways, or let the machine make an ‘educated guess’, when all outcomes are equally possible, without any access to external information. We will argue that this is one of the aspects of TMs which make them unsuitable as models of the human mind.

TMs are claimed to have ‘unlimited memory’ (e.g., Uriagereka, 2008: 228), but it is important to notice that this unlimited character only applies to the tape, not to the active memory: for a TM, ‘*the "scanned symbol" is **the only one** of which the machine is, so to speak, "directly aware"*’ (Turing, 1936: 231. Our highlighting). In this respect, there is no difference between the ‘active’ memory or Working Memory (or ‘Episodic Buffer’, Baddeley, 2007) of a TM and that of a FSA: differences arise with respect to the *probing space*, unrestricted in the former, and restricted to a single state in the latter. A TM is defined, at any moment, by its ‘*m*-configuration’ (the set of possible states) and the symbol currently being scanned.

The peculiar significance of TMs for the theory of computation is that it is theoretically possible to ‘*invent a single machine which can be used to compute every computable sequence*’ (a so-called *Universal Turing Machine*; Turing, 1936: 241). This means, in short, that for every number function there is a TM that can compute it (and every TM can be simulated by the Universal TM for arbitrary inputs); that is, the set of strings that can be accepted by a TM is *not* bound by probing memory capacities, unlike other automata. Thus, a TM can compute everything other automata can, *plus* sequences with dependencies between elements that are indefinitely far apart within the memory tape. Of course, given the fact that TMs were made to compute *number theoretic functions*, ‘every computable sequence’ means in that context ‘every number function’, not ‘every effectively computable sequence’ (it must be noted at this point that if ‘computation’ is equated to ‘mechanical deterministic method’ -as done by, for example, Murawski, 1999: 20- there is no choice but to identify the set of effectively computable sequences with the set of recursive functions over naturals, because there is nothing else out there; we will show that such identification is too simplistic for our purposes). This is a crucial leap, which -as we will see below- has frequently been overlooked in the computational and linguistic bibliography, with far-reaching consequences for linguistics and cognitive science (particularly under strong interpretations of the Computational Theory of the Mind).

TMs are examples of the so-called ‘von Neumann model’ or ‘von Neumann architecture’ (von Neumann, 1945), according to which, at any given time, only a *single agent* in the computational architecture could be active (it must be noted that von Neumann greatly admired Turing and there was personal contact and familiarity with each other’s work). This condition translates to an inherent *sequentiality* of computation, based on the concept of function (to which we will return below), which is not surprising if we consider that ‘*the subject of this paper* [Turing, 1936] *is ostensibly the computable numbers. It is almost equally easy to define and investigate computable functions of an integral variable or a real or computable variable*’ (Turing, 1936: 230)’. Milner (2006: 4) links this type of computational modeling to the properties of *short term memory*, but that is a concept he leaves undefined. For concreteness, we can think of it as a workspace in which symbolic manipulation takes place. Understood strictly, this limits our *conscious* thought to a single thing at any given time (i.e., a single derivational step), even though that ‘thing’ (object, process) might be internally complex: Feigenson (2011: 14) proposes certain ‘representational flexibility’ for WM, such that different kinds of elements can function as ‘items’ for the purposes of memory limitations (if considered in terms of ‘number of items’, in a trend traceable back to Miller’s, 1956 ‘magical number seven’), reducing the number of allowable items in the working memory to 3-4 items concurrently maintained (based on experimental data with adult humans, measuring whether they could perceive changes performed over groups of more than 4 items). Those items are allowed to be:

- Individual objects
- Sets (built by ‘binding together’ representations of individual objects ‘*while still preserving access to individual representations of the sets’ contents*’, the resulting set

being taken as a unit for the purposes of WM thus factually increasing the number of elements kept active in WM, although via a grouping operation, syntactic in nature[56])

- Ensembles (built by approximation, when numbers of objects are too big to count them individually, also taken as a whole for WM purposes)

Needless to say, the dynamic character Feigenson attributes to human WM is absent in TMs, which have an infinite memory tape but whose active memory is as limited as that of a Markov process (i.e., the present slot within the tape, as explicitly said by Turing himself). TMs, by virtue of operating a single slot in the memory tape at a time, only operate with *objects*, and are incapable of chunking and grouping operations (thus, they cannot form *sets* or *ensembles*; more generally, they cannot *categorize*). *This qualitative difference between WM and tape-memory has crucial implications for devising an implementationally plausible model for language in the human mind.* We will provide arguments in favor of interpreting the emergence of *cycles* in linguistic computations as an emergent property of a system that oscillates between different ways of chunking and grouping information: a flexible memory allows for flexible cycles, only some of which end up being *strong islands*.

2.2 The place and relevance of L-systems

In this section we will deal with a special kind of rewriting rule system: L-grammars. We will see that this very well-known kind of formalism poses a major problem for a conception in which 'computability' is identified with 'Turing-computability'. As such, it is particularly useful to illustrate a transition between function-based computation and alternative frameworks, leading to the *mixed computation* model we propose for natural languages. Lindenmayer grammars (L-grammars henceforth) also feature transition functions of the form $\Sigma \rightarrow F$, just like the formalisms analyzed in previous sections, but they have some special properties, both of which will be reviewed below:

a) Terminals / nonterminals are defined *contextually* within a rule
b) All possible rules involving elements in a representation apply simultaneously

Property (a) means that terminal and nonterminal nodes are not defined in the alphabet: in traditional computational terms, every member of the alphabet is at the same time a member of the set of initial states and of the set of accepting states. That is nearly inconceivable from a classical computational point of view, but the biological origin of L-grammars *qua* formalisms justifies this departure from classical assumptions. Property (b), on the other hand, implies a deviation from sequential rule systems (an example of which was shown in the development of Phrase Structure Rules in **2.1.2** above), since everything that *can* be rewritten at each generation *is* effectively rewritten. Moreover, the transition function in L-grammars imply that the cumulative nature of the derivation does *not* rest on subsequent generations existing at the same time (as opposed to Phrase Markers in which, say, a verb phrase contains *both* a VP node and its daughter nodes V and NP). Because rules apply to multiplicating biological organisms -like

[56] This kind of operation falls within what we have studied and described in past works as *concatenation* (e.g., Krivochen, 2015a), which is by no means specific to language. Neither is it sensitive to the inner characteristics of the objects it manipulates nor does it impose order over the resultant construct. It is thus considerably different from MGG's *Merge* (Chomsky, 1995, 2008, 2013, among others), which builds ordered sets in its orthodox definitions, an order that is ultimately mapped onto phonological precedence (see Uriagereka, 2012: Chapter 2 for extensive discussion of this latter process).

bacteria- or plant growth, once a cell divides, say, the original cell *does not exist anymore*. This follows directly from (a) and (b), and is essential to bear these properties in mind, as well as their implications, when working with L-systems from a formal point of view. Crucially for models of labelling (i.e., identification of nonterminals), L-systems in their strictest interpretation do not allow backtracking of any kind, and thus any attempt of labelling nodes via Minimal Search algorithms (e.g., Chomsky, 2015) cannot apply. More generally, as we will insist on below, the concept of *label* is not formulable under L-grammatical assumptions, given (a) above. Note that the fact that more than a single element is acted upon at any time (as we rewrite *any and all symbols that we can rewrite*) make L-systems incompatible with computational systems built around Von Neumann's architecture: unless sequentiality is forced upon the L-grammar (that is, unless we make L-systems comply with some form of Traffic Convention), simultaneity in rule application makes L-systems orthogonal to any system based on Von Neumann's architecture, which crucially includes Turing Machines.

Apart from those unique properties, L-systems share many of the aspects that characterize other formal grammars: they have an alphabet, a set of initial states, a set of accepting states, and a transition function from one state to another. The peculiarities of L-grammars pertain to how they operate with those elements, which they have in common with other formal systems. Let us take as an example an old friend, the L-grammar whose 'semantic result' (Uriagereka, 2011, 2012) per generation (that is, the result of adding the number of 1*s* or 0*s* per generation) is the Fibonacci sequence ((28), which we repeat here as (51)):

51) Axiom: 0
$0 \rightarrow 1$
$1 \rightarrow 0\ 1$

So far, we have seen sequential grammars, in which operations target *only* the current symbol in the memory tape. Moreover, the rules always involve a nonterminal as the leftmost element, whereas the rightmost side of the rule may or may not contain a nonterminal: this is known as a Chomsky-normal (or Greibach-normal) grammar (Martin, 2010: 206; Greibach, 1965). L-grammars are quite different in this respect:

> *Consider strings (words) built of two letters a and b, which may occur many times in a string. Each letter is associated with a rewriting rule. The rule a → ab means that the letter a is to be replaced by the string ab, and the rule b → a means that the letter b is to be replaced by a*. (Prusinziewicz and Lindenmayer, 2010: 3)

The grammar described in the fragment above is precisely the Fib grammar (let *a* = 1 and *b* = 0). The choice of grammar is, we think, not random: as Turing (1952), Saddy (2009); Saddy and Uriagereka (2004), Uriagereka (1998 et seq.) -among others- have shown, Fibonacci patterns are ubiquitous in biological systems at several scales, as is their ratio $\text{Fib}_n/\text{Fib}_{n-1}$, which for any two $n^{\text{th}}/n\text{-}1^{\text{th}}$ Fib terms tends to the Golden Mean as its limit (Medeiros, 2012: 206, ff.; Piattelli-Palmarini & Medeiros, 2018). Precisely, biological modelling was one of the main objectives of Lindenmayer when developing the formalism, which in the 1960s was revolutionary for two main reasons: (a) parallelism in the rewriting process and (b) the conception of grammar as a dynamic, time-dependent process (Rozenberg and Salomaa, 1980: ix). Both characteristics made L-grammars different from Chomsky-style rewriting rules:

> *In Chomsky grammars productions are applied sequentially, whereas in L-systems they are applied in parallel and simultaneously replace all letters in a given word. This*

difference reflects the biological motivation of L-systems. Productions are intended to capture cell divisions in multicellular organisms, where many divisions may occur at the same time. (Prusinkiewicz and Lindenmayer, 2010: 3)

While only one rule can apply *per generation* in a Chomsky-normal grammar, even if there is more than one nonterminal that can be rewritten[57], L-grammars rewrite *all* possible symbols *per* generation, yielding a completely different growth pattern.

The question is, where do L-systems fall *as formal grammars*[58]? Are they finite-state? Context-free? Context-sensitive? Here we will defend the idea that L-grammars as such cannot be implemented in any kind of automata contemplated in traditional automata theory, which is modelled on Turing-computability (as Rozenberg and Salomaa, 1980: x, point out, '*there is no way of presenting iterated morphisms or parallel rewriting in a natural way within the framework of sequential rewriting*'). In Krivochen and Matlach (2015), following Rozenberg and Salomaa (1980) and much related work, we proposed the following hypothesis:

No tape-based machine can compute simultaneous rule application

In effect, it is not possible to encode the grammar in (28) and its development in Chomsky-normal form; nor is it possible to make a cursor-based program model into an L-system, because the procedure does not proceed step-by-step in L-systems, but simultaneously. Consider the following scenario: let DTM be a deterministic single tape Turing Machine DTM = ($S, \Sigma, \Gamma, \delta, s_0, F$) where S is the set of states, $\Sigma \subseteq \Gamma$ is the set of input symbols, whereas Γ is the full set of tape symbols. s_0 is the initial state, $F \subseteq S$ is the set of final states, and δ is a transition function from $S \times \Gamma$. For a grammar like (28), we get $0 \in S \land 0 \in F$; $1 \in S \land 1 \in F$. But such a machine can only produce a unary string, {0}: if we indeed start with $s_0 = 0$, that is also a final state (for it appears in the right-hand side of a transition rule), and the procedure halts. If we choose to start from 1 instead, the string is still a unary string {1}. In either case, there is no need to even rewrite the relevant symbol (0 or 1) because they are both part of S *and* of F. This is most certainly not a desirable result at all, and it is profoundly counterintuitive. Of course, the problem here is not the L-formalism, but the attempt to express it in ('translate it to') Chomsky-normal form. We are not arguing that Turing computability is 'wrong' in any way, rather, we argue that there are procedures which we can characterize formally, which we call – understandably- 'computations', but which *cannot* be modelled within the Chomsky-normal space nor implemented in any formal automaton based on Von Neumann's architecture.

There is a point about 'context-sensitive' L-systems which we would like to make here, for clarification purposes. The general format of a CS-rule is X → Y / W_Z, or, alternatively, WXZ

[57] In this respect, it is useful to refer to Lees' (1976) analysis of the formal conditions over *immediate constituent* approaches to structural descriptions. He concludes that the essential condition for the formulation of the rules of a grammar '*is simply that no more than one abstract grammatical symbol of a string be expanded by a given rule at a time*' (Lees, 1976: 30). Transformational grammars in all their variants are still instantiations of Von Neumann-style architectures.

[58] This is, we are asking questions about the properties of the *grammar*, not about the properties of the *strings* the grammar can generate. In other words: we care about *strong generative capacity* and the format of the rules. We will see it is possible to find a context-sensitive grammar that generates at least some *strings* that are also derivable by L-grammars. This does not mean, of course, that the relevant L-grammar and the relevant CSG are equivalent *qua* formal procedures in any meaningful way (their *strong generative capacity* will still differ).

→ WYZ (Prusinkiewicz & Lindenmayer, 2006: 30 use the notation $a_l < a > a_r \rightarrow \chi$ to indicate that *a* rewrites as χ in context $a_l\_a_r$, thus we have $a_l a a_r \rightarrow a_l \chi a_r$). In either case, the read-write head must consider *more than a single symbol* when selecting the new symbol to be printed. Strictly speaking, and given enough memory, contexts can be complex phrases or words of any length up to the memory limits (Uriagereka, 2008: Chapter 6 presents a reinterpretation of the Chomsky Hierarchy in terms of memory capacities, terms in which is useful to think about this problem). In any kind of automaton based on Von Neumann's architecture, this presents no problem: the tape is read from left to right and the read-write head acts as a cursor. Thus, ambiguities never arise. However, this is *not* the case with L-systems, precisely because of their lack of sequentiality. Consider the following L-grammar, whose third rule is *context-sensitive*:

52) 0 → 1 0
1 → 0 1
0 1 → 1 0 1

The context-sensitity of the third rule should be apparent: 0 rewrites as 1 0 1 *iff* followed immediately by 1. Developing the grammar, we get to the following after applying the rules as appropriate:

0
10 (via Rule 1)
0110 (via Rule 2 and Rule 1)

And now… what do we do? Do we rewrite 0 1 or just 0 in the third line? The problem generalizes:

53) G = 0 1 → 1 0 1
1 0 → 0 1 0 1

If we are presented with a substring generated by *G* in (53), a deterministic parser cannot automatically decide what to rewrite: derivations in L-systems are strictly 'top-down', without any lateral orientation (and this is what allows all rules to apply simultaneously instead of sequentially). In L-systems there is no inherent hierarchy between the rules, that is, there is *nothing* in the system that determines that rule X should apply *before* or *after* rule Y, contrary to what happens in grammars in CNF: in the latter, we cannot apply lexical insertion of [ball] to the terminal N if we haven't first rewritten NP as Det + N, for instance. In a system of the kind we analyze here, under the conditions in G, every 2-*gram* generates a conflict between the two rules, which can –and thus, *must*- apply to the *same object at the same time*, as we saw above. The derivation proposed by Prusinkiewicz & Lindenmayer (2006: 31) follows a traffic condition *de facto*, which is rather surprising. Moreover, there is no way of biasing the derivational process once it has started (because we are dealing with a computable function implemented through an *a*-machine (in the sense of Turing, 1936), there is no external controller that can arbitrarily decide which rule should apply), consequently, what we get is effectively a *locally and dynamically frustrated system*:

…or this?

54) 0 1 0 1 0 1 0 1 0 1…

Rewrite this…

The computational frustration illustrated in this substring of course extends all throughout the string. *The point is that the kind of 'context-sensitivity' that L-systems present cannot be assimilated with Chomsky-normal context-sensitivity*.

A related argument against the assimilation of L-systems to grammars contemplated by the CH can go along the lines of the characteristics of the alphabet and the concept of a Chomsky-normal grammar: since in L-systems 'terminals' and 'nonterminals' are not defined in the alphabet but *contextually*, depending on the 'side' of the transition relation in which they appear; thus, we cannot establish whether '1' is a terminal or a nonterminal outside the context of a rule in a grammar like (28). L-grammars are *not* Chomsky-normal, and thus a special kind of context-sensitivity is required to interpret each symbol insofar as we have to look at the symbol in the context of a rule and know its position with respect to the transition function; we are at the limits of classical computability. Notice that this also implies L-graphs are *not* labeled graphs, because there are *no nonterminals* in the development of the formalism. However, as shown in Saddy (2018), this fact is far from being a disadvantage: if the graph (not the grammar, but the graph it generates) undergoes a series of operations (including Uriagereka's 2015 'pruning' and 'atomization', which we will review below alongside a couple of operations of our own), it is possible to transform the tree into a *normalized* form and operate over that representation in the usual way. Moreover, the format of the Fib grammar, which yields a right-adjoining tree, also guarantees that the leftmost branch at any given generation *g* is an exact copy of the rightmost branch (the 'heavy' branch) at *g*-1: this means that the Fib graph is a perfectly self-similar fractal[59], as we see in **Figure 6.**

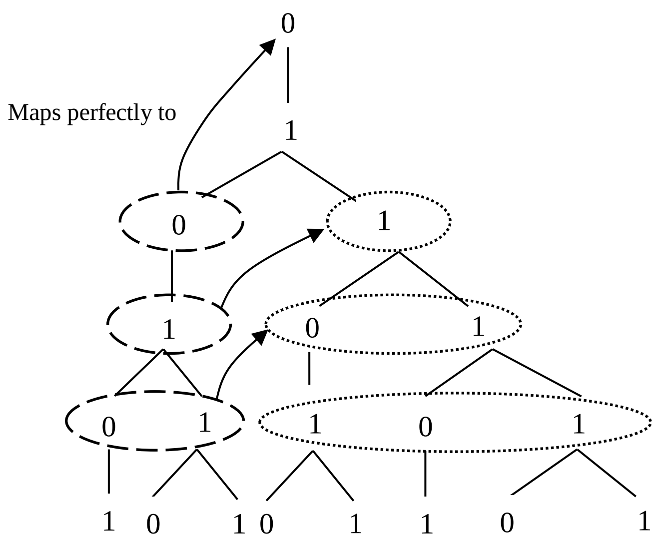


Figure 6: a perfectly self-referential L-system

[59] In Saddy and Krivochen (2016b) we have proposed the following definitions pertaining to the self-similarity of graphs generated by means of a grammar:

> A grammar G is *perfectly self-referential* iff any generation $g_x \in$ G can be expressed by means of *pure* concatenation relations between $g_i, g_j, \ldots, g_n \in$ G for *i, j, n* integers < *x*.
> A grammar G is *partially self-referential* iff any generation $g_x \in$ G can be expressed by means of a combination between concatenation relations and homomorphic mappings between $g_i, g_j, \ldots, g_n \in$ G for *i, j, n* integers < *x*.

Note that the relation between generations in the Fib tree can be expressed by means of strict concatenation relations, such that $g_n = g_{n-1}{}^\frown g_{n-2}$.

The problem we will attempt to tackle here is the relevance of L-systems in the development of emergent properties within cognition, including aspects of natural language syntax and phonology (see also Saddy & Krivochen, 2016a; Saddy, 2018 for a more developed perspective on the role of L-systems in neurocognitive dynamics from a physical-topological viewpoint). In this respect, an L-grammar like the one in (28) is extremely simple in terms of alphabet size and number of rules. Yet, because of the purely contextual definition of terminals and nonterminals, it never halts, and it generates a nontrivial pattern as an emergent property[60]. Curiously, in order to generate the same pattern directly, by means of one of the kinds of formal grammars specified above, we would need to go up to context sensitivity, and tweak the sequence to have a unary Fib pattern of the kind {0, 0, 00, 000, 00000, 00000000, …} (Holzer and Rossmanith, 1996; Mootha, 1993). It should be obvious that a string of the language generated by the grammars proposed by these authors is just a long string of zeroes, with no diacritics or spaces between chunks acting as cues to actually recognize / assign Fib. The only way to do so is to access the process *generation-by-generation*, and we are actually back to sequential grammars. It seems that, whereas it is possible (and in fact, simple) to come up with a recurrence relation function that gives us the actual Fib sequence (1, 1, 2, 3, 5…); Chomsky-normal rewriting systems of the [Σ, F] kind within the Chomsky Hierarchy (to which we will return briefly) are not capable of generating it, regardless of the power of the automaton we choose for the task. This is a nontrivial result, particularly when combined with the fact that, given a particular L-grammar (the simplest nontrivial form, we will argue), Fib arises as an emergent property of the 'semantic' (number of '1's) or 'syntactic' (number of branching nodes) interpretation of each generation (Uriagereka, 2011, 2012: 288, 2015: 666; Medeiros, 2008). Consider the evolutionary scenario this presents us with: we start with a very simple, underspecified binary asymmetric grammar (28), which produces an emergent sequence that is everywhere in the natural world, and at different scales (from sunflowers and plant growth, shell growth patterns, to virus coats and atomic structure –the latter, via Pascal triangles, as pointed out to us by Jess Tauber, p.c.). The grammar's possibilities are literally infinite: it defines an infinite phase space. What is more, it also *seems* to be the lower boundary of other, apparently more complex grammars under special conditions: consider the cases of (55) and (56):

55) Axiom: 0
$0 \rightarrow 1\ 0$
$1 \rightarrow 0\ 1$

56) XP → YP X'
X' → X ZP

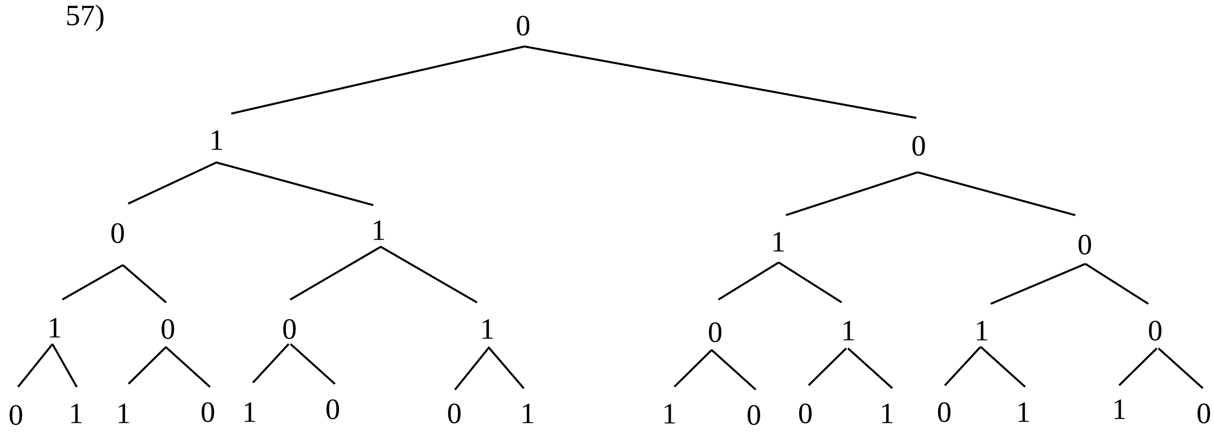
(55) is known as the XOR or Thue-Morse grammar (as it defines an exclusive-OR logical gate), whereas (56) is a well-known structure-building system in generative linguistics, X-bar theory (Stowell, 1980; Chomsky, 1970). They are both symmetric grammars, insofar as there is no 'weak' term –i.e., all nonterminals branch binarily- and all elements of the alphabet rewrite; the systems are also deterministic, as rewriting functions are unambiguous. In the X-bar template, however, each term contains a maximal and a non-maximal projection, which implies that the growth of the tree at each point is non-symmetric (particularly if, as Chomsky, 1994 argues,

[60] This simplicity in design in the context of a complex output makes L-systems good candidates for studying *computational frustrations* in a domain different from DTMs. Deterministic TMs are the examples Binder (2008: 322) proposes, as we said, because they can compute any number-theoretic function (so, they are very powerful) but they might never stop (so, they can be useless or impractical for some problems).

intermediate projections are invisible for syntactic computation, in which case only XPs and X_0 should count). There is an essential difference between Fib and XOR grammars: only the former has a weak term, which yields an asymmetric growth pattern. The growth pattern of the latter is symmetrical, as both terms are ‘strong’ (i.e., they both branch binarily). Mathematically, however, the growth patterns of both are describable by exponential functions over a growth matrix M. Interestingly, growth functions for deterministic L-systems are *at most* polynomial (Rozenberg and Salomaa, 1980: 31).

The question we would like to ask now is the following: is there any relation between the XOR and the Fib grammars? Well, it would *seem* so. The Fib L-grammar is itself irreducible: containing a minimal alphabet of elements in complementary distribution and a minimal amount of nonredundant rules, it is the simplest possible grammar including a distinguishability condition over the elements of the lexicon (i.e., a non-unary grammar) and yielding emergent outputs. Let us consider the derivation of the grammar in (55), visualized as a tree:

57)



Notice that, when identifying Fib in the development of (28), we pay attention to the development of Fib in terms of *labels per generation* (regardless of the branch length, since we can simply characterize each generation in terms of the number of rules that have had to apply to get there; we are not making topological hypotheses *yet*, but we will do so in **Part III**). This kind of interpretation, while not directly against the derivational direction (top-down), is at least *orthogonal* to it, focusing on the representations accessible at each derivational moment. In the paragraph above we proposed the possibility that the Fib grammar be a lower boundary for other L-systems (and possibly normal systems as well) under special conditions; let us now spell these out and see to what extent this claim holds:

58) a. <u>Collapse</u>: when 0 is immediately dominated by a branching 0, collapse the former with its *non-empty* sister (thus a pair (0, ε) does not collapse)

b. <u>Percolate</u>: rewrite a branching 0 immediately dominating (0, *x*) as *x iff x* is *non-empty* (once again, (0, ε) does not get rewritten)

The original idea behind those tweaks to the L-tree was to reduce the XOR grammar to the Fib grammar, in case that reduction was possible[61]. With *collapse*, the idea was to have any rewriting like (59)

59) 0 → 1 0

Replaced by (59')

59') 0 → 1

Since the 0 in the *rightmost* side of the rule is immediately dominated by another 0, and has a non-zero sister 1, collapse *should* yield (59') if applied to (59). Thing is, *it doesn't* (what is more, it *can't*), and we will see why momentarily. As we showed in Saddy & Krivochen (2016b), the space of L-grammars is 'covered' by disjoint sets: symmetric and asymmetric L-grammars, which are *neither* mutually translatable *nor* reducible to one another. Let an L-grammar *G* be a set of rules $R = \{r_1, r_2, \ldots, r_n\}$. Each $r \in$ R has an associated index *i, j, k, …n*. To each index there corresponds an integer, which denotes the number of *non-null* symbols in the right-hand side of *r*. Then,

> *An L-grammar G is said to be symmetric if* $G = \{r_i, r_j, \ldots, r_n\}$ *for* $i = j = n$ ***and*** *every member of* Σ *rewrites as a non-null symbol.*
>
> *An L-grammar G is said to be asymmetric if:*
>
> *i. Weak condition:* $G = \{r_i, r_j, \ldots r_n\}$ *for* $0 < i < j < \ldots n$
>
> *ii. Strong condition: Same as i. with the added requirement that* $i \subsetneq j \subsetneq \ldots n$.
>
> (Saddy & Krivochen, 2016b)

These conditions pertain to the rules, and provide an exhaustive classification of L-systems. Fib is an asymmetric L-system (having a 'weak' term and a 'strong' term), whereas XOR is a symmetric L-system. This classification, which attends to properties of the rules as well as the growth of the tree, helps explaining why we cannot reduce a symmetric grammar to an asymmetric one, but that is only part of the story: the kind of arguments we will revise here pertain to inherent properties of the mappings that have been proposed to carry out this 'reduction'.

Let us begin by considering the second tweak, *percolate*. While *collapse* operates by looking at the branches, *percolate* looks at the branching node, and tampers with the *leftmost*

[61] It is to be noted that a similar tweak was proposed by Uriagereka (1998: 193), but, crucially, he assumes it works, because he considers the L-tree analogous to the generative X-bar template, and in doing so, imposes 'normality' over the L-system. Without normalizing the system, there is no way of making an L-grammar halt. More recently, Uriagereka (2015: 668) proposes operations of *pruning* –'A non-branching symbol can be ignored in certain designated [structural] contexts' [adjacent to or immediately dominating an atomized '1'] - and *atomization* –'Any string of sister symbols [other than those involved in *pruning*] can be atomized into a single constituent symbol'-. (quite similar to our *collapse* and *percolate*) to transform a Fib tree into a Lucas tree (derived by the grammar 0 → 1, 1 → 1 0 *k*; a sequence 1, 3, 4, 7, 11…). These transformations also operate over labelled trees rather than over L-systems as such, for they need some *representational stability*, which a strongly derivational system with no labels cannot guarantee.

element of the rule, rewriting it in the context of a branching situation in which there is a non-zero element. Thus, we *should* get the picture in (60') given (60) as the input to the mapping:

60) 0 → 1 0

60') 1 → 1 0 (via percolation of the label '1')

If linear order –or topological orientation- is not important (and it does not seem to be, for the purposes of the emergent properties of the sequence[62]), then (60') is the strong term in the Fib grammar (i.e., the one that branches binarily), the one that actually counts for the emergent properties of the tree. Alas, this does not work either. If the movie adaptation of this thesis is ever made[63], we will be able to show the reader exactly how this does not work *in real time* -and, with the aid of special effects, *simultaneously*-, which is the essence of L-grammars. But text will have to do. One of the problems, perhaps the simplest to express without actually drawing the graph, is choosing the directionality of the application of *collapse* and *percolate*. Do we start from a rule (in which we can see what branches and what does not) and tamper with that rule? Or do we apply the rule as it is and then tamper with the resulting branching structure? In either case, where do we start applying these 'transformations' in (58) (i.e., at which generation)? If we look at a rule, we could start the derivation with the modified grammar…but from a real-time perspective that would require that we should take into consideration a pair of objects (the objects on the right side of the rule, the 'F' part in a [Σ, F] grammar) which, at the point of applying the rule to the axiom, *do not yet exist*. Since L-systems are crucially *time-dependent*, the choice regarding what to operate on has far-reaching consequences: we could even talk about hypersensitivity to initial conditions, insofar as a single transformation operating on a single branching node could greatly alter the emergent properties of the L-system. This is related to a more fundamental property of (mapping) operations: they always apply to *representations*. This should come as no surprise, since processes cannot apply to other processes (or, in other words, a function cannot take a function as its only argument, a position already developed by Frege, 1891), but must apply to an object or set thereof. Summarizing: neither *collapse* nor *percolate* can apply *derivationally*. And this is profoundly 'counter L-intuitive', because, unlike normal grammars, L-systems are *strictly* derivational: as there are no terminal / nonterminal distinction, there is no 'representational stability', the system cannot get hold of an object, because everything is to be rewritten. As a result, strictly speaking, previous generations *do no longer exist*, and thus they (taken globally, or any of its members) cannot be tampered with.

At this point, it is essential that we should make explicit the concept of *label*, since it is of key importance for understanding why the representational tweaks we have proposed (and other possible ones) will not work on an L-grammar, and why we can think of grammatical formalisms in normal form as *restricted* with respect to L-formalisms (see also Saddy, 2018). For a phrase marker, a *label* is a specification that determines how the objects that it dominates (terminals and nonterminals alike)[64] are to be interpreted for purposes of further computations,

[62] Douglas Saddy (p.c.) has referred to this as 'rotational invariance' of the Fib structure, a terminology we most enthusiastically adhere to.

[63] This is a direct reference to Uriagereka's (1998: 187) '*maybe when the film of this dialogue is shot*…'. Unfortunately, no such film has been shot yet.

[64] As we will see, for terminals, labels are usually called 'categories', but the definition is equally applicable.

including structure building and mapping operations. Thus, the branching node labeled NP is thus an instruction for any relevant system to take whatever this node dominates as being headed by a Noun and having its corresponding distribution and semantic properties (e.g., sortal referentiality). For a bottom-up system like the one customarily assumed in MGG, devising a labeling algorithm is a nontrivial task: assume we have the set {0, 1}. How is it to be labeled? {{0}, {0, 1}} (i.e., <0, 1>)? {{1}, {0, 1}} (i.e., <1, 0>)? Some $k \neq 0$, $k \neq 1$, yielding {{*k*}, {0, 1}} (i.e., <*k*, 0, 1>)? We analyzed this problem, applied to natural language, in Krivochen (2015a), concluding that a narrowly syntactic and essentially aprioristic approach to labeling (like the one proposed in Chomsky, 2013, 2015 and related works) would not work, for both theoretical and empirical reasons (a similar conclusion, in a different context, was already reached by Lyons, 1968: 231, ff.; see also the 'generatively-informed' categorial grammar proposed in Bach, 1979; and Hockett, 1954 for more general discussion about grammatical models and what they need to invoke). We therefore considered a syntax-semantics interface approach to phrase structure inevitable in order to avoid stipulations with respect to possible combinations when generating structure (e.g., the {head, XP} requirement in Chomsky 2008: 145; 2013: 43). Thus, we proposed the following criterion (Krivochen, 2015a: 545)[65]:

> *Label(S), S a syntactic object of arbitrary complexity, depending on the information conveyed by all elements belonging to S:*
>
> *a. If all elements belonging to S convey the same type of information, maintain the label for as long as elements that enter the derivation convey that same type of information;*
>
> *b. Change the label otherwise*

But we acknowledged the fact that such requirement alone was not enough if we were considering phrase markers to be structures not unlike Calder mobiles (Uriagereka, 1998: 187; see also Lasnik and Uriagereka, 2005 for a related view on phrase markers, but limited by the structure building axioms of MGG); we needed some notion of orientation or its equivalent. Thus, we proposed that *labelling* should be conceptualized as 'semantic identification' operating over pairs of symbols within a *snapshot* of the 3-D mobile (i.e., a specification of the state of the mobile at some time *T*), not because that was the inherent nature of the phrase marker as an ordered set (*contra* Kayne, 1984, 1994), but because it made things easier for interpretation purposes:

> *For any non-terminal syntactic object K = {α, β…n}* [where $\alpha \neq \beta \neq n$, in terms of semantic contribution]*, a label L is determined at the semantic interface for a pair (α, β) ⊂ K when Analyze* [any routine checking whether a SO is interpretable] *applies, after each derivational step.*

---

[65] Note that in our presentation here and in Krivochen (2015a) we say nothing about the label having to be identical to the categorial specification of a member of the set to be labeled. That is, we reject *projection* and *endocentrism* as grounding principles of syntactic structure (see also Hockett, 1954 and Lyons, 1968; also Schmerling, 1983 for discussion). If the label of a nonterminal is determined at the interpretation point, and is not inherent to an object (e.g., in the form of a diacritic), then we can say that, strictly speaking, all syntactic objects are *exocentric* in a nontrivial sense.

However, in the case considered here, things are different. To begin with, it is not clear at all that the generation-interpretation dynamics that allegedly applies to natural language also apply to L-grammars: how do bacterial reproduction or protein assembling get 'interpreted' (see Kosta and Krivochen, 2014b for some discussion)? It is not clear that our labeling criterion applies to L-grammars at all. Nor does Chomsky's: recall that L-grammars are not Chomsky-normal since there are neither terminals / nonterminals as such, nor sequentiality in rule application (i.e., no 'Traffic Convention'). If the Minimalist labeling algorithm is based on a representational cell {terminal, nonterminal}, which is 'virtually everything' (Chomsky, discussion with Cedric Boeckx, 2009: 52; see Postal, 2004: 323, ff. for discussion about *virtually*)[66], it cannot apply to a branching node dominating two elements that are, at the same time, terminals and nonterminals within the rule system (as in (28)).

Similar considerations about the ubiquitousness of labels in natural language formalizations apply to non-transformational systems, all of which are modeled upon *Chomsky-Greibach normal* grammars[67]. This includes LFG and HPSG, which use labels very similar to those used in MGG (and have at least the expressive power of CFG; see Kaplan and Bresnan, 1982; Berwick, 1984); as well as Ajdukiewicz's Categorial Grammar and its extensions. In the latter, nonterminals are labeled with slash-categories, such that '*for all α, β, where α and β are categories, α/β and α\β are categories. α/β or α\β is the category of function-denoting expressions, or functors*' (Schmerling, in press: Appendix A). Partee (1975) points out that Ajdukiewicz-style CG has the strong generative power of a CFG, but Montague grammar goes beyond that. Montague version of CG grammar uses categorial labels in a similar way, proposing basic categories (terminals) of *event expressions* and *truth values* and then recursively defining more complex categories, *some* of which correlate with MGG lexical nodes (but see Partee, 1975). For instance, in the Montagovian version of CG, for basic expressions *e* and *t* (corresponding to 'entities' and 'truth values' respectively):

> *IV, or the category of intransitive verb phrases, is to be t/e*
>
> *T, or the category of terms, is to be t/IV*
>
> *TV, or the category of transitive verb phrases, is to be IV/T*
>
> *IAV, or the category of IV-modifying adverbs, is to be IV/IV*
>
> *CN, or the category of common noun phrases, is to be t//e* (Montague, 2002 [1973]: 18)

---

[66] The full fragment is the following:

> *The crucial fact about Merge – the "almost true generalization" about Merge for language is that* ***it is a head plus an XP. That is virtually everything.*** (…) *For one thing it follows from theta-theory. It is a property of semantic roles that they are kind of localized in particular kinds of heads, so that means when you are assigning semantic roles, you are typically putting together a head and something. It is also implicit in the cartographical approach. So when you add functional structures, there is only one way to do it, and that is to take a head and something else, so almost everything is head-XP*. (Highlighting ours).

[67] For a computational perspective, see Ristad and Berwick (1989: 383-384), who formalize feature-based CF and CS (i.e., normal) grammars which are extended to get agreement phenomena, understood as feature mapping. That formalization assumes a uniform (monotonic) system of structure building, though.

Thus, while some properties of the objects that qualify for category assignment and label assignment may differ, *all syntactic formalisms we have come across explicitly or implicitly distinguish terminals from nonterminals* (e.g., *e* and *t* do *not* 'rewrite' or expand into derived categories: they constitute the finest level of granularity in the Montagovian CG algebra). It must be noted that there is *no* direct correlation between the *basic category*/*derived category* and *terminal*/*nonterminal* distinctions, and this is because both basic categories and derived categories are in fact labels, basic categories being defined over the alphabet, and derived categories being defined recursively. So, *t* is a basic category, defined over the alphabet $\Sigma = \{e, t\}$, and *e*/*t* is recursively defined over the same alphabet. In a word, all syntactic formalisms deal with normal grammars, otherwise, there is no way to account for constituents –or domains, or cycles, or clauses…may the reader name it– and hierarchical structure within a finite string (and in fact, CGs have been proven to be equivalent to CFG, see Pentus, 1993). Other models, which explicitly reject the Immediate Constituent concept of 'label' (e.g., Dependency Grammar and related approaches, see Osborne et al. 2011, 2012; also Tesniére, 1959), do so *without* rejecting the existence of nonterminal nodes or some other kind of 'nomenclature' for words vs. non-words (i.e., structures made up from words), thus, they have the same kind of representational stability that label-based theories like MGG have (and they are even more dependant on the notion of 'head': there is no exocentricity in DG structures). In terms of generative power, Obrębski & Graliński (2004) provide arguments that DGs (at least in the version of Mel'čuk, 1988 and related works) are actually CFGs (see also Debusmann & Kuhlmann, 2010 for a classification of DGs of different power, which caps at CF). DGs eliminate intermediate levels of structure (i.e., 'intermediate projections' in X-bar jargon) and phonologically null elements, as well as the notion of 'constituent' itself, but crucially *not* the distinction between terminal and nonterminal nodes: because *dependency* is a *mother-daughter* relation, a terminal is simply a daughter which is not a mother, and a nonterminal is the elsewhere case (including the root). Or, in other words, a DG contains two sets: a set of lexical items and a set of categories: lexical items are assigned to categories and categories are organized in directed acyclic graphs. It is crucial to stress, again, that 'nonterminal' is not synonymous with 'constituent', which might be an equation the reader familiar only with MGG might be tempted to do. Dependency Grammars are undoubtedly different from constituent-based grammars, but they are modeled upon normal grammars in formal terms all the same.

Going back to the operations proposed in (58), it is important to stress that both mappings *collapse* and *percolate* operate over *labels* in a Chomsky-normal grammar (which is a problem that is *independent* from the classification of L-systems into symmetric and asymmetric proposed in Saddy & Krivochen, 2016b; we could take a symmetric L-system like XOR or an asymmetric L-system like Fib, and the operations would fail in the same way). They are defined *representationally* and thus require access not only to the branching node (i.e., the leftmost element of the rule) but also to the 'leaf' or 'leaves' dominated by it (which in turn may be branching nodes themselves!). The application of such strongly representational operations goes directly *against* the essentially derivational character of L-systems and their simultaneity: were we facing a Chomsky-normal grammar, we could apply representational constraints because such grammars are *mildly* derivational, their sequentiality being based on the premise that only a single nonterminal rewrites at each derivational step. We could thus affect a single nonterminal of our choice at a single derivational step, and everybody would be happy. Sadly, we cannot do this, unless we are willing to stipulate not only which nonterminal is to be affected, but also that the rest of the structure must be left untouched (or else the process would be non-conservative). Moreover, as we have seen, L-systems do not distinguish between terminals and nonterminals, with these notions being defined contextually within a rule. Thus, the strongly derivational character of L-grammars prevent us from operating over labels, that is, local chunks of structure, and tampering with the procedure. These grammars allow orthogonal processes, like *interpretation* (which targets the tree from one side, so to speak), but it is utterly impossible to go *against* the derivational flow. Simple though the grammar is, it defines a phase space with very specific properties, and it generates a nontrivial emergent pattern.

The consideration of Fib grammars over a unary alphabet, as in Holzer & Rossmanith (1996) and Mootha (1993), poses an interesting problem for AGL: how is it possible that a string of zeroes *contains* the Fibonacci sequence?[68] We know it does, in a very specific way, because we have access to the generative procedure and can resort to a recurrence relation to calculate the n^{th} term of the sequence (i.e., the number of zeroes in the n^{th} generation), but this information is not available when a parser of whatever nature is presented with the bare string, since the string contains no information about where to chunk it in order to get the Fib pattern (or any other emergent property, for that matter). Unary grammars are not frequently used in AGL, true, but even if we considered a {1, 0} alphabet, the *structure recognition vs. assignment* problem remains. To what extent can we say Fib *is to be recognized* in a string like S = {101011010110110101101} (7^{th} generation of the grammar in (28))? From an *assignment* point of view, things are not any clearer: since given a string S generated by an unknown L-grammar there is no algorithm to decide which L-grammar generated S, it is not possible to decide whether the S has been generated via (28) or via, say, a rule 0 → 101011010110110101101 (non-recursively). We can take Kolmogorov complexity (*minimal description length*) as the deciding factor (thus, since (28) is simpler, it is to be preferred over a structural description that just repeats the string), but we have to devise grammars in order to compare them complexity-wise. From a formal point of view, things are anything but straightforward.

So far, we have posed some questions pertaining to L-systems and their relation to the Chomsky hierarchy and other formal languages deriving from Chomsky-normal grammars. We will return to all of these questions in the remainder of the present work, since L-systems play a crucial role in the oscillatory engine we will describe for linguistic computations.

The next section will deal with the relation between two major paradigms for computation, including of course linguistic computation: function-based digital computation and interactive computation. The model for current theories of syntax is undoubtedly the former, and it has been since the 1950s, with the emergence of transformational generative grammar. The classical view of cognitive computation as function-based computation remained unchallenged even when some foundational notions of MGG were revisited (for instance, the development of non-transformational grammars in the 70s and 80s). We have seen some shortcomings of a structurally uniform conception of phrase markers for natural language sentences, now we will provide further arguments in favor of non-digital, interactive, continuous views of computation which also have a deep connection to the neurocognitive substratum of computation.

2.3 Functions and interaction

[68] As a matter of fact, it is still an open problem in computer science how to come up with a suitable (and non-trivial) L-grammar given a particular string, for the Entscheidungsproblem applies to L-systems as well. We can speculate, but a formal proof is still to be devised.

The belief that linguistic computation falls within Turing limits and must be characterized in the same manner as functions over naturals is already a commonplace in (bio)linguistic theorizing. From the arguments against finite-state models in Chomsky (1957, 1959) to the latest developments in favour of full Turing-computability in Watumull et al. (2014), encompassing the PDA+ proposal in Joshi (1985), the finite-state models for processing and structure in Kornai (1985), Galley and McKeown (2007), and Yli-Jyrä, Kornai and Sakarovitch (2011), and the dynamic perspective in Uriagereka (2008, 2012) -to name but a few-; linguistic computation has been hypothesized to be exhaustively describable by means of a Turing machine, formal models that are mathematically equivalent to a TM or which a TM is a proper superset of. While not all the aforementioned authors agree with respect to *which* computational procedure is the most accurate, they all agree that the procedures they advocate for are ultimately expressible in terms of a deterministic Turing Machine (DTM), a position summarized in Chomsky's theorem (1959: 143):

61) *THEOREM 1. For both grammars and languages, type 0 $\supseteq$ type 1 $\supseteq$ type 2 $\supseteq$ type 3.*

Where:

62) Type 3: Finite-state grammars
Type 2: Context free grammars
Type 1: Context sensitive grammars
Type 0: Unrestricted grammars

In a word, as the standard Church-Turing Thesis (CTT henceforth) assumes,

63) *TMs can compute any effective (partially recursive) function over naturals (strings)* (Goldin and Wegner, 2007: 16)[69].

It is not our purpose to discuss the CTT, which we will take to be true for the purposes for which it was devised, but to provide arguments for the idea that it does *not* apply to natural language processing or to its structure (and, more generally, that mental processes are not limited by the boundaries of function-based computation). The CTT applies, as we will see in more detail below, to the computation of number functions (often referred to as *algorithmic procedures*), which transform a finite input into a finite output by means of a *closed* process involving serial and subjacent applications of rules or operations, in which a single 'agent' is active at any given time (so-called 'von Neumann's bottleneck'). From the very beginning of modern computer science there was a divorce between properties of programs and the physical substratum on which they were implemented, as pointed out by Milner (2006: 4):

[69] Turing's original (1936: 230) formulation is somewhat different, and as we will see, more restrictive than modern interpretations: '*I give some arguments with the intention of showing that the computable numbers include all numbers which could naturally be regarded as computable.* […] *they include, for instance, the real parts of all algebraic numbers, the real parts of the zeros of the Bessel functions, the numbers π, e, etc. The computable numbers do not, however, include all definable numbers*' It is essential to consider that Turing himself was aware of certain limitations on the theory of computable number functions that his successors seem to have overlooked almost completely. If TMs cannot compute all definable numbers (e.g., numbers that are still to be classified as either algebraic or trascendental; imaginary parts of algebraic numbers, etc.), then we have that much less reason to expect the theory of TMs to include all *effective computation* (i.e., everything that can indeed be computed).

The early computers all followed the model of John von Neumann, in which—as far as the programmer was concerned—only one thing could happen at once; at any given time only one agent could be active. So the possibility of concurrent activity or even co-existence of such agents could not be expressed in a program—even though underneath, as it were in the machine's subconscious, many wheels would whirr and circuits cycle simultaneously.

However, even in modern computers, concurrence is the norm rather than the exception: an operating system interacts with several sub-programs at a time, coordinating and controlling their actions and re-assigning resources as required (e.g., CPU vs. GPU). Our hypothesis is that *the generation and processing of natural language, because of its structural characteristics and the way in which it emerges from an essential dynamical frustration at the core of cognition, is not the sequential, closed computation of a number-theoretic function, therefore, the CTT does not apply*. Consequences for structure building and mapping are numerous and far-reaching (some of which have been explored in past works); here we will focus our attention on locality phenomena (in **Part IV**) and oscillating computation.

We will argue that natural language generation and processing makes use of *open*, *nonlinear* processes, and *multiple* (though finite) inputs of different origins and natures (visual, auditory, tactile, etc.), impossible to calculate or foresee *a priori* and integrate in the definition of a function over 'integral variables', interacting with *outputs* to generate representations of the structure and meaning of a linguistic stimulus in real time. The notions of extrapolation and inference are also crucial to natural language processing, since recourse to different sources of information at different derivational points allow decisions to be made with what would be otherwise insufficient data if one makes use of a single source (e.g., just linguistic information represented as an *explicature*, in the sense of Wilson and Sperber, 2004).

It is essential to distinguish the original CTT (as correctly expressed by Goldin and Wegner in (63)) from its extensions, which are very common in the literature. Thus, for instance, Fitz (2006) proposes and discusses the so-called Physical Church-Turing thesis (see also Piccinini, 2011 for analysis and discussion), according to which a function is computable by a physical system *if and only if* it is TM-computable (see also Deutsch, 1985 for a view that is related, although more influenced by quantum mechanics). This claim amounts to saying that all computations within physical systems are function-based, which in fact requires independent formal proof. Deutsch (1985: 3) explicitly formulates the physical CTT principle as follows:

I can now state the physical version of the Church- Turing principle: 'Every finitely realizable physical system can be perfectly simulated by a universal model computing machine operating by finite means'.

However, Deutsch himself acknowledges a version of the CTT that only deals with function-based computation while not providing a formal proof that every finitely realizable physical system is actually function-based (strangely enough, no citation information is provided for this formulation, which is attributed to Turing):

[…] *according to Turing,*

Every 'function which would naturally be regarded as computable' can be computed by the universal Turing machine. (Deutsch, 1985: 3)[70]

Deutsch advances on the implementational concerns pertaining to the CTT, by pointing out that:

> *I propose to reinterpret Turing's 'functions which would naturally be regarded as computable' as the functions which may in principle be computed by a real physical system. For it would surely be hard to regard a function 'naturally' as computable if it could not be computed in Nature, and conversely* (Op. Cit., 3)

The 'conversely' part is the one we are not so sure about: along with Wegner (1997), Milner (2006), among others, we propose that there are computations that occur in 'Nature' (e.g., the development of L-systems; interactive-based computation in general) which are not Turing-modelable (and incompatible with a 'single-action-at-a-time' architecture).

G&K (2010: 48) adopt a similar version of the physical CTT, as a putative consequence of Turing's work:

> *A most surprising result that emerged from the work of Alan Turing* […] *is that all of the functions that can be physically implemented as procedures (all computable functions) can be determined through the use of composition and a very small class of primitive functions.*

This formulation (singled out as an example) explicitly claims that physically implementable functions are equal to computable functions, and, moreover, that they are determined by a small set of primitive functions, which can be decomposed into simpler functions: $f(x) = x^2$ can be decomposed into $f(x) = x \times x$, which in turn can be decomposed into $(x + x)$… x times.
The aforementioned equivalence seems to be taken for granted, without it requiring an appropriate formal proof, not to mention a proof that cognitive operations are in fact functions over naturals. G&K simply assume, as we have mentioned above, that function-based computation is the most efficient way of processing and transmitting information in a physically realizable system, and they explore no further options in this respect. They claim that

> *A guiding conviction of ours – by no means generally shared in the neuroscience community – is that brains do* [something] *close to the best possible job with the problems they routinely solve, given the physical constraints on their operation. Doing the best possible job suggests doing it digitally, because that is the best solution to the ubiquitous problems of noise, efficiency of transmission, and precision control.* (G&K 2010: 24)

While digital, function-based computation might be the *best* solution in the simplest scenario and considering no contextual variables (let us assume that just for the sake of the argument, but this is actually *an empirical question* still unsolved), when we put the (embodied) system to work in a physical environment of ever changing stimuli and in which decisions have to be

[70] Unsusprisingly, we still see such misconceptions about what the CTT actually is, and what Turing wrote, as in an anonymous review we received: "*The point of a Turing machine was to have a device so open-ended that* ***anything one could possibly imagine being a "computation"*** *would be something that could be accomplished by a Turing Machine*" (our highlighting). The equation of effective computation to Turing-computation is obvious here.

made on the bases of partial and multimodal information, the advantages of digital computing are not so clear: if information sources are not digital but analogic, then digital brains should – inexplicably- convert analogic signals into digital form. Moreover, and on a more general note, it is also not clear that we want the *best* solution; rather we want *an optimal* solution *for a given problem, in a given context*, to be solved with a dynamically changing array of resources (which can be redirected, for instance). G&K's 'guiding conviction' that brains optimize by no means implies digital computing (thus, G&K 'suggest' that is the best way, even though later on their prose is quite a bit more assertive). In other words: the claim that brains optimize is distinct from the claim that optimization equals digital computing. G&K devote quite a bit of space to arguing in favour of the implementation of the functions they hypothesize are at the core of cognitive computation in a TM, in a not-so-covert plea for computational uniformity (following the subset relations established in the CH). The following passage is revealing as to what G&K consider goes on in the brain:

> *Nothing we currently understand about computation in the brain presents any challenge to a Turing machine. In fact, all current formally specified models of what goes on in brains are implemented on contemporary computers.* (Gallistel & King, 2010: 106)

It seems clear that, unless models (i.e., programs) are isomorphic to actual neurodynamics, the fact that a model has certain characteristics or is implemented on a specific kind of hardware tells us little about the nature of the process being modelled. That is a separate problem. As a matter of fact, there are aspects of neurodynamics relevant to computation / cognition that find no explanation in digital models of computation, including, for instance, resting-state gamma bursting. All brain activity must be synaptic for G&K's argument to hold, and, moreover, synapsis must be conceived of as an *on-off*, switch-like mechanism. Even if this was accepted, however, the bridge between single-neuron behaviour and higher-level processes (e.g., categorization) would remain unaddressed.

In a similar vein to the other sources we have reviewed, Hopcroft and Ullman (1969: 80) claim that TMs have been proposed as the computational devices for describing 'procedures', and explain that:

> (…) *from the definition of a Turing machine, it will be readily apparent that any computation that can be described by means of a Turing machine can be mechanically carried out.* […] *It has been hypothesized by Church that any process which could naturally be called a procedure can be realized by a Turing machine. Subsequently, computability by a Turing machine has become the accepted definition of a procedure.*

There is a subtle difference between H&U and G&K: while we can infer from H&U that there can be mechanical procedures (in the mathematical sense) outside the scope of a TM (but not the other way around), G&K (2010: 48) take a stronger stance, *equating* TM-computation to physical implementability. The apparent relativization that H&U make of the CTT makes it a better candidate for interpretation, although it is still, in our opinion, beyond the original scope of Turing's work. It is potentially problematic that there can be mechanical procedures outside the possibilities of TMs, even though such procedures are not mentioned. If such procedures exist, then TMs cannot be the definition of 'procedure', because only a subset of mechanical procedures is TM-compatible.

Copeland (2002a) discusses Deutsch's (1985) physical extension of the CTT, claiming that

The notion of an effective method played an important role in early debates about the foundations of mathematics, and it was sufficiently clear to allow Turing, Church, and others to recognize that different formal accounts gave alternative modellings of the notion. Their notion was certainly not that of a 'finitely realizable physical system'

but himself proposes a version of the CTT that is stronger than Deutsch's: all *effective computation* is carried out by a TM. Thus, all effective computation is function-based in this view, and can be modeled by means of a TM:

There are various equivalent formulations of the Church-Turing thesis. A common one is that every effective computation can be carried out by a Turing machine.

In Copeland's article, 'effective' is not (at least explicitly) equivalent to 'function-based', therefore, we can safely say that, despite the possibility that this is an independent thesis from CTT, it is certainly not equivalent to the original CTT (which makes no reference to *effective computation*). It is crucial to point out that Turing's object in his seminal (1936) paper were computable numbers, but he claims his theory can be extended '*to define and investigate computable functions of an integral* [read: integer] *variable*' (1936: 230). Crucially, a TM (and thus, any other model of computation that claims to be formally equivalent to a TM) is at every step determined by its *configuration*, which includes the symbol that is currently being read $\mathfrak{G}(r)$ and a condition q_n, from the set $\{q_1, \ldots q_n\}$ of the possible states of the machine (1936: 231), what Turing calls an 'automatic machine' (or α-machine). However, Turing himself concedes that

For some purposes we might use machines (choice machines or c-machines) whose motion is only partially determined by the configuration […]. *When such a machine reaches one of these ambiguous configurations, it cannot go on until some arbitrary choice has been made by an external operator. This would be the case if we were using machines to deal with axiomatic systems.* ***In this paper I deal only with automatic machines*** (…) (Turing, 1936: 232. Highlighting ours)

For our purposes, it is essential to keep in the foreground the fact that the formalization in Turing (1936) is aimed at automatic machines, since this helps us grasp the true scope of CTT and the impact it has had in formal linguistics, particularly linguistic theories of generative orientation. We will compare the overall characteristics of proposals that accept the stronger versions of the CTT, the physical version in particular, with proposals that embrace a wider concept of computability, which properly includes *algorithmic computation*, but goes beyond its limits: so-called *interactive computation*. We argue that the latter paradigms (developed in Wegner, 1997, 1998) are better equipped to deal with natural language processing, as well as with purely formal issues, like what we have argued is the mixed nature of phrase structure (Krivochen, 2015a, 2016c; Krivochen and Schmerling, 2016), and the continuous interplay among the syntactic mechanisms and semantic requirements required throughout a derivation. Crucially, this does *not* mean that function-based computation is rejected at all. What it means is that there is no reason to assume that human cognition is essentially digital or algorithmic; and more specifically, that there is no principled reason to accept the superiority of proof-theoretic approaches to natural language syntax over alternatives (model-theoretic and interactive) as a given. As we pointed out above, these are *empirical questions*, which we address from the perspective of linguistic computation, more specifically, the interplay between structure and meaning.

2.4 The algorithmic vs. the interactive nature of computations

2.4.1 Function-based computation

It is essential that we should define clearly the terms we will work with in the remaining sections. For starters, we must make explicit what we include within the term 'function'. The definition we adopt is the following (based on Falcade et al., 2004; Youschkevitch, 1976/1977: 39; May, 1962, among others)

> *A function is a deterministic relation between a set of inputs and a set of permissible outputs with the property that each input is related to exactly one output by some definite rule.*

That is, a function establishes a one-to-one relation between an input and an output. Here, we will only discuss *injective* functions, which are those that preserve *discreteness* in the following sense: an *injective* function never maps more than one element from its *domain* (that it, the set of permissible inputs) to a single element of its *codomain* (the set of permissible outputs). A function-based machine provided with a suitable (possibly finite) input set and a finite set of rules for its manipulation, can generate a (possibly finite) output set. Consider, for example:

64) $f(x) = x^2$

This function relates each value of x to its square x^2 by means of a definite rule, 'multiply x by itself'. The alphabet is (say) $\mathbb{Z}$, and provided an infinite tape as in a TM, the function can go on and on, until halting is stipulated by a rule. Crucially, neither the alphabet nor the rules can be changed in the middle of the process, nor can information from other systems influence the process. Moreover, each step is subjacent to the previous one, defining subjacency for a sequence as follows:

> *A process p is an ordered series of steps [$x^\frown x+1^\frown \ldots x+n$], where $^\frown$ is linear ordering; and, if x is subjacent to x+n via x+y, for any integer y, then x+y is a necessary condition for a linear deterministic derivation to reach x+n departing from x.*

This is easily exemplifiable: let us take (64). If the input is the set of positive integers $\mathbb{N}$, (64) can be defined sequentially by means of Peano's axioms (see Leung, 2010 for a clear explanation of recent reformulations of the original axioms, adapted to modern terminology). However, this means that there is a certain *order* in the introduction of the input to be manipulated by the function. A fragment of the derivation of (64) would be the following:

65) Step 1: $f(1) = 1^2$
Step 2: $f(2) = 2^2$
Step 3: $f(3) = 3^2$
…
Step n: $f(n) = n^2$

The n^{th} step is defined by the function alone, as the system has no access to earlier information or to what will come next. The value of x at each derivational point (each 'step') is defined *at that point alone*: we need not access any other step. Moreover, we can tell the following derivation is ill-formed, even though every step is well-formed itself: there are constraints not only on objects ('representations'), but also on the operations that apply to those objects, in terms of *order* and *subjacency*:

66) Step 1: $f(1) = 1^2$
Step 2: $f(3) = 3^2$
Step 3: $f(2) = 2^2$

A simple informal proof could go along the lines of Peano's axioms: if $\mathbb{N}$ is the result of applying a successor function S starting from $0 \in \mathbb{N}$, such that 0 is not the successor of any member of $\mathbb{N}$, each member of $\mathbb{N}$ has exactly one successor, and no member of $\mathbb{N}$ can be the successor of two other members of $\mathbb{N}$; we have to follow the application of the successor function in order to get the derivation right. Instead of 1, we can write $S(0)$, bearing in mind that S means *succession* here. 2 is then $S(S(0))$, and so on. If 3 is put as the second element in the derivation, as in (66), then it should be defined as $S(S(0))$, but that would mean that both 2 and 3 are the successor of the successor of 0, which violates the axioms. This is intimately related to the concept of *injective function*, since in such functions, $f(a) = f(b)$ *iff* $a = b$, which is indeed one of the axioms (Murawski, 1999: 100); an *injective function* never maps a single element of its domain to more than one element in its codomain. Strict sequentiality (and rule ordering), then, is derivable from axioms in number-theoretic functions, which were the original scope of Turing's (1936) work.

Linearity and sequentiality are characteristics that all TMs share; the reader should recall that every state of a TM is determined only by its *configuration*. Rules of the form '*if…then…and…*' (e.g., in everyday language, '*If you find 1, then write 0 and proceed right*') have access to the content of a single state at a time, because the read-write head has that limitation (at least in its TM incarnation, then, if all function-based computation can be modeled by a TM, it should follow that this feature is common to function-based computation. This limitation is unavoidable in a Von Neumann architecture).

In and of itself, there would be nothing objectionable to this view of computation, if it were not for the fact that it is a common practice to equate *computation* with *computation of functions*, as Davis (1958) eloquently puts it in his introduction to a now classic textbook:

> *This book is an introduction to the theory of computability and non-computability, usually referred to as the theory of recursive functions.*

We are thus faced with *two* problems, not just one. The CTT is a claim that must be separated, as Goldin and Wegner (2007: 21) point out, from the assumption that *computation* equals *computation of functions*:

> *The legitimacy of this claim* [that a problem is solvable *iff* there is a TM for computing it] *is based on two premises. The first one is the Church–Turing Thesis, which equates function-based computation with TMs. The second one, usually left unstated, is the mathematical worldview—the assumption that all computable problems are function-based.*

This equation has far-reaching consequences, for even those who have tried to increase the power of TMs in order to compute more complex problems (e.g., Deutsch, 1985, who holds a physical view of TMs and expands the concept to a Universal Quantum TM; Santos, 1969 and his analysis of *probabilistic* TMs; among many others) have come up with algorithmic extensions which are either generalizations of TMs (including procedures such as increasing the number of tapes, changing the alphabet, and so on; for instance, Deutsch conceives of the set of states and the memory tape as Hilbert spaces, and the transition function is an automorphism, a generalization of a transition monoid); or straightforwardly not TMs in the sense of Turing (1936) (e.g., if a probabilistic TM requires an external controller so as to avoid the 'luckiest possible guesser' situation when probabilities for outcomes are roughly even at a stage q_n, it does not fall within the definition given above of an α-machine). In sum, in function-based computation, all we need to know the output (say, the value $f(x)$) is to have the value of the input (x) and the rule that relates input and output values (the *transition function*, call it f). The process is closed (for no external sources of information are allowed to influence the process once it had begun), linear (for successive steps are subjacent), and allows for no interaction with an external agent.

It must be noted, though, that deterministic functions do not exhaust the possibilities for a TM. In fact, non-deterministic (probabilistic, quantum, stochastic) TMs occupy a good part of the TM spectrum nowadays. However, if the CTT is correct, then they are all equivalent to a deterministic TM under appropriate circumstances. In G&K's argumentation (2010: 43-44) it is essential that the theoretical machine is *deterministic*, and algorithm-based. While we agree with them in that we would like the theory of cognitive operations to be *physically realizable* (i.e., achieving implementational adequacy, borrowing the term from Marr, 1982), there is no need (in fact, we maintain that it is a mistake) to equate *effective computation* with *function-based computation*, as G&K do:

> *We say that a procedure, when effected as a computation,* ***determines*** *a function, and that the procedure, as a physical system,* ***implements*** *the function.* (G&K, 2010: 44. Highlights in the original)

Our objections *are not aimed at the CTT as such*, but at the misinterpretations and extensions of Turing's thesis beyond Turing's (1936) claims about *a-machines*, and more specifically, at the attempts to model human cognition around the equation of *effective computation* and *function-based computation*.

We will now proceed to review some arguments that have arisen against this equation and its consequences, discuss each of them in turn, and then present the framework we will adopt for the rest of the present work (to be expanded on and complemented in **Part III**).

2.4.2 Arguments against function-based computation

Before the emergence of interactive computation in the mid- and late '90s (e.g., Wegner, 1997), Arbib (1987) had already criticized the 'mind as a serial, one-at-a-time operation computer' position, which had been dominant since the '60s after the so-called 'cognitive revolution' with ideas like the Derivational Theory of Complexity (DTC henceforth) making their way into the newly born Cognitive Science. The models of neural networks from the '60s onwards, influenced by traditional psychology and neurological theories that studied the brain in isolation from its environment (the 'brain-in-a-vat' approach), as well as theories about mental computation (in the sense of symbolic manipulation) and representation, involve a series of simplifications:

- Discrete states / discrete time
- Fixed threshold and weight for neural networks[71]

[71] This simplification rests on the assumption of a Hebbian learning equation: $\Sigma_i w_i(x_i) \geq \theta$, where w is the input weight, θ is the activation threshold, and x is the timing of firing, for i neurons.

- No external influences, information encapsulation
- No non-synaptic neural interactions (e.g., interactions due to an externally induced electromagnetic field, as in an MRI; or energy dissipation in the form of heat as a sub-product of action potentials; also ephaptic coupling)

These aspects might be a historical consequence of adopting the CTT, explicitly or implicitly, as a model and guidance for cognitive science. But external influences and non-discrete states are essential parts of more recent interactive models of mental computation. In this context, it is crucial to take into account that while interactive models make use of information coming from the external context, the amount of such external information at any given time, as something directly perceptible or directly inferable (see Sperber and Wilson, 1995; Wilson and Sperber, 2004), is at best partial and incomplete. The importance of *context-dependant inferential mechanisms* to fill information gaps is crucial, and thus an explicit theory of inferences is required as part of an interaction-based model of mental computations. Our position is that context sensitivity enters the game as a way of preventing an over-activation of contextual knowledge: access to properties of lexical items, of constructions (e.g., idioms), as well as recourse to extra-linguistic knowledge (including what is immediately perceivable and inferable as well as associations which are freer, but not completely unrestricted) are triggered and restricted by the interpretational requirements over a particular string *S* in a context *C* by an interpretant subject *S*. If the process were independent of contextual factors, then there would be no way of preventing the activation of an unrestricted number of contextual propositions. Inferential mechanisms could only be *partially* statistic-probabilistically modelled, if the right variables were taken into account: in any case, permanent two-way communication between the *m*-configuration (i.e., the configuration of the assumed machine at any given time, see our discussion of TMs above) and external information sources is crucial for the success of the derivational inferential process. The partial and incomplete character of external information is no impediment to decision-making, as Kilmer et al. (1969: 282) note in a foundational paper: subsystems make tentative decisions based on partial information which is integrated across systems, and resources can be dynamically re-allocated depending on contextual necessities (for instance, '*In fighting*, […] *the visual system sacrifices spatial acuity to gain a slow motion picture of the world, the pain system nearly shuts down, clear thinking and long-range planning cease (suicidal attacks are common), vocalization is ejaculatory, rage is expressed, and movements tend to be stereotyped action patterns*', Kilmer et al., 1969: 282).

When we make an interpretative hypothesis (in our terms, when we assign a grammar to a certain object or string), at no point do all subsystems unequivocally opt for one model over another. That is -and we will return to this below-, we keep our options limited, but open in the dynamical process of assigning structural representations to particular strings. There is a competence between possible structural descriptions *in real time* that is decided by *both* simplicity and faithfulness measures, as the resolution of the tension that arises between them; *simplicity* is likely entropic, whereas *faithfulness* presumably departs from what is the simplest case computationally.

The 'discrete time' problem is tackled by Spivey (2007: 30) in a straightforward way:

> (…) *it seems highly unlikely that naturally organized systems of multiple interacting units, such as a brain, a society, or a planet, would function in lockstep to the pace of some systemwide counter using a particular temporal quanta, such that each new second, or millisecond, or attosecond signalled an instantaneous and simultaneous updating of*

> *the discrete state of each and every unit in the system. This is not a straw man I'm building. This is the kind of lockstep synchronization that would genuinely be necessary for the brain to function like a digital computer.*

The case Spivey builds is one for continuous time as the basis of cognitive processes. While it seems obvious to point out that discrete labels of time are just that, labels, and that time is not *uniformly* compartimentized in physical systems, the consequences of assuming discrete-time as a methodological choice cannot be overlooked. It is not obvious that cognitive measures / perceptions of time are actually synchronized to any external, phenomenological measure of time; rather, as we will see, it seems that the cognitive perception and manipulation of time is dynamic, being able to adapt to more than a single measure, and even operate at more than a single temporal scale simultaneously, at both cognitive and neurological levels (see also Tognoli and Kelso, 2014).

Milner (2006) analyzes some shortcomings of strict digital, function-based theories of cognition from the opposition between *concurrency* and *sequentiality*. Concurrency might be defined, simply, as several programs running simultaneously, under the control of an operating system. As Milner (2004: 5) suggests, concurrency is the *norm*, not the exception (particularly in biological systems, as Prusinkiewicz and Lindenmayer, 2010: 3 point out), and, importantly, the methods devised for sequential computation *cannot* be applied to concurrent computation, thus making TMs unsuitable as formal models for a huge number of processes (including, we hasten to add, natural language processing). Sequential computation, by virtue of its limited power in terms of operations that must be carried out in polynomial time, is appropriate to deal with non-complex systems, in the technical sense (see, e.g., Boccara, 2002 and Ladyman et al. 2012, for an introduction): however, if the system in question is the result of multiple agents interacting in a non-linear way, thus deriving the relevant properties of the system by emergence (e.g., the Internet), strictly sequential computational models are not enough. Sequential models, in which encapsulation of information is often a crucial part, have great difficulties when faced with simple real-world activities involving subjects interacting with their surroundings. Spencer, Perone, and Johnson (2009: 100) correctly emphasize that models of neural networks often overlook the fact that more than a single function might be required at any given point. Their specific example concerns neural networks that serve either a perceptual or a working-memory function: the task of remembering the location of critical objects in the immediate surroundings of the subject requires '*the integration of both functions*' (Spencer, Perone, and Johnson, 2009: 100), such that perception and working memory functions cannot be in complementary distribution (to use a phonology metaphor) in a given neural network, nor can their process be sequential, let alone informationally encapsulated. Standard TMs display not only strict sequentiality, but also *determinism* in the application of rules. Determinism means, in this case, that at no point will we need to resort to an external agent to make a choice (in Turing's 1936 original formulation, *c*-machines were explicitly left aside), because at all points, the state of the machine is a function of the *m*-configuration. Therefore, the (α-) machine only does what rules tell it to do, in the order they tell it do to things, without interacting with other machines, information sources, or an external operator.

A claim that is common to most proposals for function-based computation, to a different extent (and with different degrees of explicitness), is that the properties of the physical substratum (composition, limitations, possibilities of interaction and interconnection…) are not relevant for the description of the formal capacities of the abstract machine. Therefore, it is not unreasonable to think of a machine with an unlimited memory tape, even if its physical implementation is impossible (in neurocognitive terms, this would mean an organism physically instantiating the abstract machine should have unlimited Long Term Memory); nor is it unreasonable to assume that the active read-write head has access to a single slot within the tape at a time (which translates in neurocognitive terms to a very limited WM—an assumption that does of course not follow empirical evidence but must be true by the definition of what a TM can do at each step). To phrase the problem in philosophical terms, proponents of function-based computation implicitly or explicitly embrace ontological dualism, the separation of the mind (or, in this case, the abstract computational machinery) from the body (the physical substratum). In this respect, function-based computation represents the mathematical aspect of traditional views within cognitive science and philosophy of mind, commonly grouped under the vague (and often incorrect) label '*computational theories of the mind*' (CTM). As we saw above, equating *effective* computation to *function* computation is stretching the CTT far beyond its original sense and scope, which means that we can have computation either *beyond* the Turing limit (e.g., the notion of *hypercomputation* put forth by Siegelmann, 1995 in the form of a chaotic dynamical system –the 'analog shift map'–, and much subsequent work, see e.g. Syropoulos, 2008; Copeland, 2002b for an overview), or *just outside* the original Turing number-function-based template (assuming that all other forms of function-based computation, from finite-state automata to Linear-Bound automata, are subsumed to TMs; see Chomsky, 1959). On the other hand, with the advent of neural networks and computer modelling, the position that since a single neuron can be essentially modelled as a binary switch (fire / not-fire) which performs computations (in some not very well specified sense), the whole brain is just a collection of such switches, thus essentially a digital computer itself. The introduction of more points in the scale (so that there is place for more subtle adjustment, 'fire more/less strongly' for instance) does not make an essential qualitative difference, since the object is still modelled as digital and discrete. Some proponents of this position go as far as suggesting that, since it is possible for a neuron to adjust its firing pattern to environmental conditions (including timing and strength; see Johansson et al., 2014 for a purely neural study), the neuron is somehow *computing* instead of, say, adjusting dynamically to its environment with no formal operations involved. Needless to say, this position is faced with a gap in terms of how to get from adjustment of firing patterns to even simple arithmetic operations, which under any definition constitute *computation*. Not to mention more complex tasks, including those which require decision making under non-binary conditions (when we are not 100% sure about something, or there are holes in the data that is available to us), including –but not limited to- natural language use. Let me illustrate: imagine we take apart two N64 cartidges (say, Super Mario 64 and Ocarina of Time). We then study each little piece of cartridge separately, to the greatest possible level of detail. We know how each piece works and what it does. We know about ROM memory limitations, processing speed of the console, GPU power, any technical detail there is to know. Now, if we do that, will we be able to tell, from a given piece of cartidge, to which of the games it belongs? Are we going to understand the game? Will we have any clue about how to get good at that game? Will we know the glitches, bugs, game-breaking moves? More fundamentally, will be able to tell the games apart? The scope and nature of these questions (some of which have to do with the concept of *emergence*, but others tackle more essential issues), and their relation with the methodology of understanding the hardware at a *very local* level is, I think, pretty much what the 'neuron as a digital computer' proposal should take seriously sooner than later in order to be useful for a science of the human mind and brain, which need not be coextensive with Computer Science. If anything, it is an *empirical* matter, not the result of an aprioristic decision. Of course, some traditional CTM enthusiasts answer all those questions 'yes', like Brown (2014) –to give a recent example-, and consider that tools like the Hodgkin-Huxley equations (which we will come back to later on) are adequate formalizations (even though little if anything is ever said about their *explanatory* value, or lack thereof). The object and the model are, for these authors, homomophic, and there is no essential 'gap problem'. However, a properly worked out (i.e., a program, an algorithm…) solution of how we could speedrun Mario 64 or beat Ocarina without dying once provided with (only) a detailed description and complete understanding of the cartridge components (or, how we can build inferences, associations, and so on given linguistic stimuli and a number of neural networks of the mechanistic kind) is never provided.

It is important to note that some of the defenders of the traditional CTM acknowledge some of the flaws we have mentioned to different degrees (e.g., Charles Randy Gallistel, in the *Observer*, 29(5), 2016[72]), but more often than not they are not willing to give alternatives a chance because for the purposes of computer modelling, digital models work well (even though, as Bill Benzon (p.c.) very well observed, '*the "neurons" in neural networks resemble real neurons about as much as this ☺ resembles the Mona Lisa*.': this is not meant to be damning for the traditional CTM, but it should get us thinking about its limitations when it comes to making assertions about the human mind and brain). While proponents of this view often criticize interactive computation for its lack of implementability as a computer program, we think it is precisely because it *cannot* be implemented by means of a program based on binary code in a digital computer that interactive computation is an interesting path to explore, particularly if supplemented with physical considerations across levels of granularity. We are, of course, *not* saying that a deep understanding of the structure and function of each individual part (e.g., each individual neuron) is not required, *quite on the contrary*. What we say is that a purely mechanistic theory *is not enough* to account for the inner workings and use of cognitive processes like –but not limited to- natural language processing, at both the levels of production and comprehension. Any mechanistic theory, we argue, *needs* to be complemented with an emergentist perspective. The gap we have mentioned before pertains to the granularity appropriateness of a theory, and where that theory puts 'computation' (including, of course, what 'computation' is taken to mean and which processes is it taken to comprise). We will return to problems of granularity (and what should and should not be expected to hold across granularity levels) all throughout the thesis (and see **Section 1.4** above).

It is methodologically and empirically necessary to make a distinction at this point in the argumentation: the hypothesis that the mind performs operations over symbolic objects—a position often referred to as *symbolism* or the *representational theory of the mind*, see Pitt, 2013 for discussion— among which we are particularly concerned with *structure building* and *structure interpreting* operations, is something not only different from but also independent of the view that the mind performing those operations is a *digital* computer (or any other kind of α-

[72] Even after more than 50 years of CTM, Gallistell writes that '*What we haven't yet learned are the answers to the computational questions that we have learned to ask. We do not yet know how the brain implements the basic elements of computation (the basic operations of arithmetic and logic). We do not yet know the mind's computational primitives. We do not yet know in what abstract form (e.g., analog or digital) the mind stores the basic numerical quantities* [this expression 'abstract form - numerical quantities' already presupposes an answer, I hasten to add; cf. Spivey, 2007. It is not clear what 'storage' is supposed to mean in this context, either] *that give substance to the foundational abstractions, the information acquired from experience that specifies learned distances, directions, circadian phases, durations, and probabilities. Much less do we know the physical medium in nervous tissue that is modified in order to preserve these empirical quantities for use in later computations*' (Gallistel, 2016). Maybe it is time to change the paradigm, or at least complement the one that has proven so far inadequate to answer those questions.

machine) operating with *functions* in *polynomial time* in a *serial* manner (*determinism* aside, since TMs can be probabilistic to various extents). Of these two theses, we adhere only to the former, while stressing the point that the former is not *necessarily* linked to the latter. In fact, the negation of the latter thesis does not result in inconsistent systems (just like with the negation of Euclid's fifth postulate, which, in combination with the assertion of the other four, gave rise to hyperbolic and elliptical geometries, both perfectly consistent formal systems). Rather, it results in logically robust and empirically plausible theories about the mind-brain relation.

2.5 Alternatives to function-based computation

In this section we will review some of the alternatives to the second thesis we have mentioned, the equation of *effective* computation and *function-based* computation performed by a *digital* mind. We will characterize two alternatives to that thesis, which are not equivalent: on the one hand, we will introduce some concepts of *embodied cognition* and *enactive cognition*; on the other, we will deal in more depth with *interactive-based computation*, which will be crucial for the development of the present work.

As our point of departure, it is useful to consider the proposal of Stewart (2010: 1), who argues that any paradigm in cognitive science must fulfill two requirements:

- It must provide a genuine resolution of the mind-body problem (or, otherwise put, prove it is not a problem)
- It must provide for a genuine core articulation between a multiplicity of disciplines, including psychology, neurology, and linguistics

Undoubtedly, as the works of Fodor (1975, 2000), Chomsky (2002, 2005)[73], G&K (2010), among others[74] show, different variants of the traditional CTM are consistent systems that provide answers to both requirements, within their respective capacities and aims. We argue here that those answers, logically consistent though they might be (overlooking for the moment the objections that Behme and Postal, among others, have raised regarding the coherence of the generative biolinguistic ontology), are at odds with empirical linguistic phenomena, and also psycho- and neurolinguistic experiments. We will also make the case for a need to explore alternatives that enhance empirical adequacy while having theoretical advantages as well, roughly following the epistemological model of Lakatos (1978).

2.5.1 Embodied Cognition and Enactive Cognition

[73] Interestingly, Chomsky (together with other proponents of 'biolinguistics' within generative linguistics) claims that there is a connection between Generative Grammar and Biology (such that, for instance, UG, the initial state of the Faculty of Language, is innately given, by means of genes). We have argued (Krivochen 2016a; 2016b) that there is no solid evidence for a non-trivial interpretation of innateness within generative biolinguistics. See Behme (2015a) and Postal (2004; 2009) for extensive discussion from philosophical and linguistic perspectives, respectively, and the contributions in Kosta and Zemková (2015) for a more nuanced discussion of this issue.

[74] We have focused -and will continue to do so- on those references in which language has a preeminent position, since we are primarily interested in the dynamics of linguistic computations and their neurocognitive substratum. This is in no way meant to imply that other references are not relevant for the purposes of the CTM, but simply that they are less relevant to the present work.

Broadly speaking, '*Cognition is embodied when it is deeply dependent upon features of the physical body of an agent, that is, when aspects of the agent's body beyond the brain play a significant causal or physically constitutive role in cognitive processing*' (Wilson and Foglia, 2011). It should be obvious (but we wish to stress it nonetheless) that this position is *not* compatible with Cartesian dualism, which (allegedly) lies at the foundations of Generative Linguistics (Chomsky, 1966; see Behme, 2014 for an updated critical discussion). Rather, it is closer to complex systems science (Chemero, 2009; also Dotov, 2014; Cosmelli and Thompson, 2010), and compatible with the hypothesis that computation is an emergent property of neural networks *qua* dynamical nonlinear systems. Cosmelli and Thompson put this explicitly:

> (…) *numerous neurobiological considerations* […] *indicate that the brain needs to be seen as a complex and self-organizing dynamical system that is tightly coupled to the body at multiple levels.* (2010: 362)

Self-organizing systems are the object of studies in dynamic systems, and a crucial part of non-linear physical models, insofar as outputs are not linear functions of inputs (self-organizing behavior being an emergent property). While the original concept of self-organization was formulated by a mathematician (Ashby, 1947, 1962)[75], it has rapidly gain acceptance in biology, physics, and neurodynamics. The principle of self-organization (rephrased in modern terms from Ashby's original formulation) states that the behavior of a dynamical system which behaves as if it was physically (here, *materially*) implemented with more than one variable tends towards a stable point or set of points in a phase space for a wide variety of initial conditions, what is referred to as an *attractor* (e.g., Ashby, 1962: 270). Population dynamics, climatic conditions, and heartbeat dynamics, all have a similar behavior in this respect, which we will explore in the present work since we argue that language also shares this dynamics, initially presented in the form of a dynamical frustration between semantics and morphophonology, to be later on generalized and studied more in depth.

The assumptions and proposals that CTM and embodied cognition make are, *in some respects*, not as different as they appear. For example, curiously enough, Chemero (2009: Chapter 3) argues *against* the concept of representation in cognitive science (at least, as 'representation' is traditionally understood), a point in common with the otherwise diametrically opposed proposal of Gallistel and King (2010) –but under completely different assumptions-. That is, there is no inconsistency (either logical or empirical) in maintaining the *notions* of computation and representation within agent-environment dynamics (as we will see in more detail below), as long as the definitions of computation and representation are adequately modified to take into account the properties of the actants and the relevant context, and the subsequent *emergent* properties of their interaction (an idea that goes back to Aristotle's *Metaphysics*; see Boccara, 2002 and Hofstadter, 1979 for an introduction; also Fromm, 2004 for some discussion about the relevance of the concept of *emergence* in science and other areas of human practice). In any case, methodologically, Chemero's (2009) initial chapter is an excellent introduction to the methodology of (cognitive) science and a critical assessment of the empirical validity of logically sound arguments (e.g., *non-sequitur* in Hegel's arguments against an eighth planet in the solar system, as well as Chomsky's critique of Skinner's behaviorism and his own

[75] It should be noted that the concept of 'self-organization' currently in use is often different from the sense in which Ashby meant it; Ashby himself feared this and wrote (1962: 269) "*since the use of the phrase 'self-organizing' tends to perpetuate a fundamentally confused and inconsistent way of looking at the subject, the phrase is probably better allowed to die out*". It should be evident that it hasn't.

proposal of an innate Faculty of Language). While some premises in an argument using *modus ponens* might be sound (and even correct) on their own, this does not mean, as it should be obvious, that the conclusion derived from the collective consideration of these premises and their vinculation via a context-free logical argument form is equally (theoretically) valid or (empirically) true. This is, we think, the case with Chemero's own arguments against a Fodorian(-like) CTM: *computational cognition* is most emphatically not equivalent to *body-independent*, *function-based* computation, which is what Chemero (and other proponents of embodied cognition) assumes: this is only so if '(effective) computation' is equated to 'digital, closed computation'; otherwise, there would be no inconsistency in proposing a computational approach to '*Agents and environments* [being] *modeled as nonlinearly coupled dynamical systems*' (Chemero, 2009: 31): problems arise with closed, sequential, linear models. Tacitly, alternatives to the CTM have often indeed accepted the strongest versions of the CTT, which equate *effective* computation to (number-theoretic) *function-based* computation: the reasoning these alternatives pursue is that, since function-based computation does not capture the properties and behavior of interactive organisms, and, more specifically, of crucial mental processes carried out by those organisms (among which we will focus on *Homo Sapiens* as the relevant organism, and *language* as the relevant process), then 'computation' as a concept is to be directly eliminated from cognitive science[76].

Within embodied cognition, the proposal known as *enactive cognition* has gained popularity since the seminal publication of Varela (1979), in which the relevant division is not between mind and body, as in the Cartesian tradition, but between *matter* and *living organisms*. Living organisms are defined as context-interactive processes of indefinite self-engendering, and cognition is defined, in turn, as a property of living organisms. Bourgine and Stewart (2004: 327) define a system as 'cognitive' if and only if it *generates its actions* (notice the importance of agency), as opposed to the 'brain in a vat' metaphor used in traditional cognitive science. Cognition is thus not something that occurs within an agent, but is rather the product of the interaction between agent and environment. Enactivists do not accept the proposal that a brain separated from the body and connected to a computer which controls its functions via (say) electric stimuli (a famous thought experiment proposed –but not endorsed- by Hilary Putnam; a contemporary counterpart to Descartes' Evil Demon), is an adequate paradigm for cognitive science, since in this case, 'actions' would come from the outside (i.e., from the computer). That kind of system, strictly speaking, would not be considered 'cognitive' under strict enactivist assumptions: the 'mind' extends beyond the brain, encompassing the whole body and its environment (Bernecker, 2016: 69); this is known as the hypothesis of the *extended mind*, which includes *enactive cognition* and *distributed cognition* (the idea that cognition and knowledge are not confined to an individual but are distributed across objects, individuals, artefacts, and tools in the environment).

The claim that life and cognition (of which we are only concerned with the latter) are not 'things' but 'processes' is far from nonsense, but there have been enactivist attempts to include Buddhist philosophy and phenomenology within the cognitive paradigm that are, at best, only tangentially scientific, and strongly metaphysical. In our opinion, those works (e.g., Varela, Thompson and Rosch, 1991) undermine the status of enactive cognition as a serious

[76] This argument has certainly been oversimplified for the sake of the exposition, but it is by no means a straw man: we have left aside the technical details, but the core points of radical embodied cognition proposals are as we have summarized them here.

contender for traditional cognitive science. Note, however, that it is not logically inconsistent to claim that cognition is indeed a process, of a *computational* kind. The devil, as usual, is in the details: what do we understand by *computation*?[77] In this respect, enactive cognition does not provide sufficient formalization (it seems to be focused mostly on philosophical argumentation rather than on neurocognitive evidence or mathematical proof) to compare it face-to-face with traditional CTM. It is to be noticed, however, that enactive approaches to sensory-motor processes and categorization (e.g., Thompson, 2007) are usually presented without appealing to metaphysical arguments, and have drawn attention from different disciplines. All in all, and without attempting to exhaust the topic, we do not consider *enactive cognition* as a serious contender to the traditional CTM or even subparts of it, like the notion of function-based computation. For starters, it is not as developed as the latter approaches, but it also does not seem to be developing in a direction that will yield a coherent, formalized, and unified framework.

2.5.2 Interaction-based computation

Interaction-based computation, advanced by Wegner (1997, 1998); Goldin et al. (2004); Milner (2006), among others, is based on the idea that

> *computation is viewed as an ongoing process that transforms inputs to outputs – e.g., control systems, or operating systems.* (Goldin and Wegner, 2007: 5)

It is important to notice that interaction-based computation (in which several sources of information and processes are dynamically accessed and combined *during* the computation in order to yield an output) and function-based computation are not related by means of subset relations (i.e., function-based computations are not a proper subset of interaction-based computations), but they model processes of different nature. Thus, the CTT remains valid, even though only for function-based computation, for which it was originally proposed.

An example of a non-function-computable process is given by Wegner (1997): program a car to drive back and forth between home and work with a predetermined set of instructions. Ongoing processes are crucial to solving this kind of problem. If the computation were function-based, then a finite and encapsulated input containing, say, a detailed map, a weather report, etc., could not foresee dynamic obstacles (like the speed and direction of the wind, which has to be compensated for; pedestrians, who change their location permanently –e.g., as they are crossing the street, etc.); and thus the output, consisting (say) of the rotational position of the steering wheel plus shift-changing at predetermined rpm (assuming a manual transmission) and the pressure on the fuel, brake, and clutch pedals could not be determined by the input alone: the car might very well run over an occasional pedestrian that was not taken into account in the input and, since function-based computation is by definition closed, could not be included at any point in the computation. In the event that a pedestrian appeared in front of the car running over

[77] A similar perspective, although significantly less developed, is that of Wilson and Foglia (2011), who claim that '*This general approach* [enactive cognition] *encourages a view of enaction as essentially distinct from computation, as it is traditionally conceived*.' No details are provided with respect to what exactly 'traditionally conceived' implies: we have argued (and presented evidence to support the claim) that 'traditional' computation is *function based computation*. If this is the case, as we think it is, then enactive cognition does not preclude the possibility of a computational mind, it just does not accept that the nature of those computations can be function-based. However, Wilson and Foglia do not present an alternative in terms of computation, and their definition of 'process' is rather informal; more appropriate, perhaps, for certain branches of philosophy than for cognitive science.

said pedestrian is a possible, yet not preferable, output. However, the appearance of the pedestrian requires of an external controller to decide what to do, as the instructions specified in the input are not enough to give us a satisfactory result: a function maps a set of inputs to a set of outputs, without having the possibility to establish preferences depending on contextual conditions: that is the realm of c-machines (which are explicitly outside the scope of the theory of computable functions as formulated in Turing, 1936), or, better still, *interaction*. The crux of the problem is that, if all computation were function-based, the 'driving home from work' problem would fall outside the realm of *computability*, which is in our opinion a wrong assessment of the situation: the problem is computable, just not *function-computable*. As Goldin and Wegner (2007) correctly point out, a mechanism that continuously receives input from the road and processes it in real time, evaluating possible courses of action and establishing preferences in real time, adjusting the behavior of the steering wheel and pedals accordingly (i.e., generating a dynamic non-linear output), is not only *necessary*, but also *sufficient*, to solve the problem. Implementing interactive computation is not an easy task, particularly because 'implementation' must still struggle with the limitations of digital computers and the *pre-hoc* nature of programs determining possible responses. Müller et al. (2015), for instance, classify 'interactive' software in two broad categories: *functional-reactive programming*, which enables interactive programs to be written at a higher level of abstraction by using primitives that operate directly on time-varying or reactive values, and *event-driven programming*, which enables the programmer to write efficient and responsive interactive programs by offering a general-purpose, expressive language and providing complete control over computation and input sources. However, the difficulty of realizing the abstract ideas given the unpredictability of the external world and bidirectional data flow (and information bleeding) often limits the practical possibilities to variants of standard Turing-computable functions (for example, Müller et al. develop an 'interactive' language based on Church's λ-calculus, called λ^i; however, that language explicitly forbids storing past interactions, which is actually an asset in embodied cognitive systems).

Apart from *dynamic and continuous input reception*, an essential feature of interaction-based computation is the *entanglement of output and input*: later inputs depend on earlier outputs, and these later inputs are adjusted accordingly; moreover, the adjustment of later inputs may require access to previous real-time updating cycles (e.g., aspects of the mapping between input and output at an earlier derivational point).. Thus, for example, the position of the steering wheel at a particular point in time t_x depends on its position in t_{x-1} *plus* the new information from the outside world obtained in the derivational step relating those states, together with inferential processes regarding the properties of the road, how the car will behave under those conditions, and possibly a competition between decision courses based on extrapolation from incomplete data. This view is, obviously, quite a bit more complex (in the technical sense) than traditional proposals about computation in the mind, such as the one put forth by G&K:

> *The communication between the world and the brain* […] *is a mapping from states of the world to representations of those states.* […] *From a mathematical perspective,* ***these mappings are functions****. The notion of a function is at once so simple and so abstract that it can be hard to grasp. It plays a fundamental role in our understanding of representation* […] *and computation* […]. (G&K, 2010: 43. Our highlighting)

This view is incompatible with *radical* embodied cognition (Chemero, 2009: 31) insofar as the environment plays no active role in cognitive processes: it is just there to be mapped, and it is that mapping that gets operated over in digital symbolic manipulation. Functions, we have said before, entail unidirectional information flow (e.g., following the sequence of $\mathbb{N}$) which in the case of mind-world relations would translate into world-to-mind flow only: we can easily picture 'quanta' (in the sense of 'discrete, quantifiable objects') of information sequentially entering the relevant vertical faculties via transducers, adopting a Fodorian stance. On the other hand, interactive-based models allow access to multiple sources of information during the process, as well as parallel computations, interacting at non-aprioristic points. Now, let us consider a possible counter-argument: of we have a multi-tape machine, couldn't it process several information sources at once? It sure could, but there would be no *interaction*, for each tape is informationally encapsulated. One process *cannot* influence another. While a function is all about its output, interactive computation is focused on the *process* to get to that output, a process of which the subject (in the sense of an external controller, along the lines of Turing's c-machines) is an essential part. It is to be noticed that the concept of 'interaction' does *not* refer only to linguistic computations or communication, but to *any* dynamic computation in which derivational steps are not strictly subjacent in the sense specified above. Take, for instance, the computation of explicit and implicit content in linguistic stimuli (techically, *explicatures* and *implicatures* respectively) in Relevance Theory (Sperber and Wilson, 1995; Wilson and Sperber, 2004). Consider the following quotation, in which the procedure is made explicit:

> *Sub-tasks in the overall comprehension process:*
>
> *a. Constructing an appropriate hypothesis about explicit content (in relevance-theoretic terms, explicatures) via decoding, disambiguation, reference resolution, and other pragmatic enrichment processes.*
>
> *b. Constructing an appropriate hypothesis about the intended contextual assumptions (in relevance-theoretic terms, implicated premises).*
>
> *c. Constructing an appropriate hypothesis about the intended contextual implications (in relevance-theoretic terms, implicated conclusions).*
>
> *These sub-tasks **should not be thought of as sequentially ordered**. The hearer does not FIRST decode the logical form of the sentence uttered, THEN construct an explicature and select an appropriate context, and THEN derive a range of implicated conclusions. **Comprehension is an on-line process**, and hypotheses about explicatures, implicated premises and implicated conclusions are developed in parallel against a background of expectations (or anticipatory hypotheses) which may be revised or elaborated as the utterance unfolds. In particular, the hearer may bring to the comprehension process not only a general presumption of relevance, but more specific expectations about how the utterance will be relevant to him (what cognitive effects it is likely to achieve), and these may contribute, via backwards inference, to the identification of explicatures and implicated premises.* (Wilson and Sperber, 2004: 615. Our highlighting)

A function-based approach is manifestly incapable of performing these computations in which on-line processing and non-sequentiality are central[78]. Given the scenario presented by Wilson

[78] Consider as well the case of multiple readings for a single sentence, where we have both literal and non-literal (hyperbolic, metaphorical) meanings:

i) A: What do you think of Norbert's last book?
B: It puts me to sleep

and Sperber, the following consideration by Goldin and Wegner (2006: 29) only reinforces the connection between information sources all throughout the process we have proposed:

> *Turing machines and algorithms must* ***completely specify all inputs before they start computing****, while interaction machines* ***can add actions occurring during the course of the computation****.* (our highlighting)

Needless to say, the kind of complete input specification that is required in TMs is not the case with inferences, as they are often made upon incomplete data (i.e., partially unspecified inputs), or with processes that depend at least in part on trial and error (an Analysis by Synthesis approach to comprehension, for instance; see Townsend and Bever, 2001), involving more than a single hypothesis, as well as a process of refining the initial interpretative hypothesis (so-called 'quick and dirty parsing').

Things get even more complicated when one considers that there are two kinds of explicatures (which constitute the first sub-task of the comprehension process listed above):

- Lower-level (related to propositional content: the extended causative verb phrase *v*P, in MGG terms)
- Higher-level (related to the *modus* and illocutionary force, including Modality –and associated projections- and the Complementizer Phrase CP, again in MGG terms)

Relevance Theory has the merit of being one of the few pragmatic theories that has a natural bond with syntactic theory, since explicatures are built based on information present in the syntactic structure as *conceptual* and *procedural categories*; a distinction that Escandell and Leonetti (2000: 367) have argued corresponds to MGG's distinction between *lexical* and *functional* categories[79], a distinction or long tradition in linguistic studies (hints can be found even in Aristotle's *Poetics,* XX; more recently, Brøndal, 1928; Jespersen, 1933). Their thesis is that '*the semantics of functional categories* […] *is always of the procedural type*' (2000: 367; our translation), a proposal that links syntactic structure with post-(narrowly-)syntactic inferential processes. The semantic contribution of D(eterminer), T(ense), C(omplementizer), and the causative *v* is thus procedural, and Escandell and Leonetti make it explicit for each

Wilson and Sperber (2004: 619-620) argue that literal meanings are *not* considered and rejected first before considering non-literal meanings (in case the literal meaning goes against Relevance expectations, which might not be the case here), but that in interpreting (i) there is no conflict between the literal and non-literal interpretations (there might be cases where there is such a conflict, and in those cases, the theory predicts the literal meaning is not even considered). Hyperbole and metaphor are not *subsequent*, but *alternative* routes to maximal Relevance.

[79] It must be stressed, however, that the correspondence between *lexical* and *conceptual* is weaker, as Sperber and Wilson (1995) and Leonetti & Escandell (2004) acknowledge. Some items that are customarily categorized as 'lexical' in terms of their categorial features [N] and [V] in the Chomskyan tradition actually provide *procedural instructions* to the interpretative systems (prepositions being clear examples of this). However, in Krivochen (2012a, b) we have argued that in fact these elements (most notably, some adverbs) are not lexical even under MGG assumptions: following a lexical decomposition approach, adverbs are the result of a conflation process involving a root and a relational, locative node within a VP (following Mateu Fontanals, 2002; Hale and Keyser, 2002; among others, which in turn borrow much from *generative semantics*). If this is the case, we argued, then adverbs are in fact procedural, because Ps are the prototypical procedural elements, conveying instructions to relate conceptual elements in terms of *central* / *terminal coincidence*.

category (see Krivochen, 2015c for discussion of *v* and its semantic contribution). Interactive computation seems to be the way to go for the study of explicit and implicit content in natural language. Furthermore, in combination with specific architectural assumptions, it can provide answers for long-standing structural puzzles, including the emergence of cycles and the local inadequacy of phrase structure grammars.

2.6 Interaction in Language and Cognition: some developmental and evolutionary considerations

Let us summarize what we have so far with respect to the two competing models of computation:

	Computation	Interaction
ACTIVE ENTITY P:	program	active object, agent
ITS MEANING:	function	process
STATICS (COMBINATION):	sequential composition $P_1; P_2$	parallel composition $P_1 \parallel P_2$
DYNAMICS (ACTION):	operate on datum	send/receive message

Table 2: Interactive vs. Function-based computation (from Milner, 2006: 6)

In general terms, we have two groups of theories of language and cognition: those built around *balance-based* systems, and those built around *tension-based* systems. Balance is defined here in very much the same way 'equilibrium' is defined in Binder (2008): an altogether halt of the system dynamics. Balance-based systems need well-formedness to be defined a priori (thus being strongly *constructivist*, in the sense of Lasnik and Uriagereka, 2005), and work with *closed* derivations in which there are no external perturbations. The dynamics of such systems is linear, the output being a function of the input derived via subjacent steps, as discussed in **Section 2.3**. Broadly speaking, this label includes all versions of the Minimalist Program, HPSG, LFG, and, in general, *proof-theoretic systems*, including P&P-like frameworks (the Parallel Architecture and Simpler Syntax architectures as well). The evolutionary bases of these models, when spelled out, are incremental: departing from a hypothetical evolutionary scenario in which *x* traits were available, these theories assume that additional mechanisms incremented the computational capacity (thus, *evolution* implicitly understood as 'change for the better'). This results in a general belief within MGG that evolution has led to a *perfect* faculty design (e.g., Chomsky, 2000: 96), built around the notion of *conceptual necessity*: the properties of the computational system for human language $C_{(HL)}$ perfectly satisfy the requirements of the interface systems C-I and S-M; moreover, any apparent 'imperfection' (uninterpretable features and displacement) is in fact part of the optimal solution. Kinsella (2009) correctly points out that evolution does not optimize a biological system, let alone, yield perfect systems (2009: 55). Similarly, Simon (1984: 138) claims that '*no one in his right mind will satisfice if he can equally well optimize; no one will settle for good or better if he can have best.* ***But that is not the way the problem usually poses itself in actual design situations***' (our highlighting). A good example of an incremental theory is the 'biolinguistic version' of *protolanguage* theory[80] (e.g., Bickerton, 2007), which is inscribed more generally in a framework in which evolution has

[80] We are forced to clarify that we are referring to the *biolinguistic version*, because 'protolanguage theory' with that name has quite a tradition in historical linguistics, although with a very different interpretation (consider, for instance, the reconstruction of the hypothetical Proto-Indo European).

stopped, both with respect to the genus *Homo* in general, and to language, more in particular (Bickerton, 2007: 511[81]). Pidgins, in this theory, are intermediate steps towards a 'fully fledged' language, a process which involves *adding* elements, pidgins lacking '*anything one would want to call syntax*' (2007: 515). Protolanguages (a term that Bickerton claims to be '*so* ***widely accepted*** *that the term seems to have passed into the general vocabulary of language evolutionists*', 2007: 515, our highlighting), in this view, involve a '*structureless*' system, and there were *no intermediate steps* between protolanguages and '*the earliest fully syntactical language*' (Bickerton, 2007: 511). Of course, a formal definition of 'syntax' and a clarification of why 'one' would want to call something 'syntax' (or why it should matter what one would 'want' to do) is never provided. The conception of 'computation' assumed in these approaches is also, from our perspective, not particularly clear:

> *Chomsky (1980)* [*Rules and Representations*. Oxford: Blackwell] *has made a clear distinction between the conceptual and the computational aspects of language. The conceptual element, especially if we take it to include conceptual structure as well as the lexical instantiation of the latter, must be* ***phylogenetically far older than any computational mechanism****.* (Bickerton, 2007: 511. Our highlighting)

If, as in the present work, we take 'computation' in the sense of 'symbolic manipulation and imposition of structure', without further specifications (and, crucially, without assuming that *effective* computation equals *function*-based computation), then conceptual structure *is* computational in a non-trivial sense (after all, it is *structure*). Moreover, a syntactic approach to conceptual structure like the one pursued here, as well as in Krivochen (2015b) and suggested in Uriagereka (2012), can tell us quite a lot about the physical and biological properties of the substratum of which those structures are emergent. We will return to this below, but for the time being, we would like to stress the contradiction we see in talking about *structure* without *computation*.

There are other approaches which, while rejecting the idea of protolanguage in Bickerton's sense, require a *mutatio ex machina* of some sort to justify the evolutionary emergence of a 'simple operation' *Merge*, which overnight allowed a certain species –strictly speaking, a certain individual in a certain species, via a 'mutation' in some unspecified gene- to command discrete infinitude (see, e.g., Berwick and Chomsky, 2011). This picture is somewhat simplified, but not too much. The idea of a sudden mutation, or 'rewiring' (Chomsky, 2005: 11, ff.; 2008: 137; 2009: 29, 31) of the brain is recurrent in so-called 'biolinguistic' works,

[81] Let us present the quotation in context:

> *First and perhaps most important is that "the evolution of language" is far too vast and complex (and vague) a concept for anyone to say anything sensible about it. In order to get a handle on it, it is vital first to isolate it, and then to clearly delineate what are the main issues. On the isolation issue, here is a classic How-Not-To, courtesy of Science, no less:*
>
> > *Language evolution has not stopped, of course; in fact, it may be progressing more rapidly than ever before. (Culotta and Hanson, 2004:1315)*
>
> *Of course it has stopped, because the biological evolution of humans* […] *has, to all intents and purposes, stopped also.* (Bickerton, 2007: 511)

The relation between 'language evolution' and 'human evolution' for Bickerton is not clear at all. Also, there is no unambiguous definition of 'language', thus we do not know what exactly is supposed to have stopped evolving.

particularly of the Chomskyan persuasion. These mutations / rewirings always *add* something to a system that somehow did not have the possibility of symbolic manipulation, and have no 'intermediate steps', they are a matter of 'all or nothing'. The arguments always proceed in the same way (we will not do a bibliographic survey here, but the reader is more than welcome to do so): a mutation yielded Merge, Merge made it possible not only to build phrases but also to *displace* them, provided that Move is actually Internal Merge. Movement is an apparent imperfection of language, but it is actually driven by the need to eliminate uninterpretable features, which are also an imperfection. Therefore, the two imperfections 'cancel' and we have a *perfect* system (Chomsky, 2000, 2005, 2007, 2009; Berwick and Chomsky, 2011; Boeckx, 2008; Lasnik, 1999; among many others). It is in this tradition that the FLN / FLB view arose, which we consider otiose and obscure. It will become obvious in the discussion that follows that the FLN/FLB distinction is both theoretically inconsistent and empirically inadequate: on our view, there is nothing 'specific' to language (whatever this is taken to be) other than the dynamic intersection of independent systems.

In contrast to *balance-based systems*, *tension-based* systems exploit the non-linear dynamics that arise from actants interacting in an open system, where forces pull to different directions in a *phase space* comprising all *formally possible* solutions, restricted by the requirements of substantive systems (e.g., form / meaning). The universal substratum of 'language' is thus a *phase space* containing everything that is *cognitively* possible. Within this space, possibly ultrametric in its topology (we will come back to this in detail below; for the time being, it is important to point out that *ultrametricity* and *dynamical frustration* are closely related in their physical interpretation), topological considerations are included as factors determining *attractors* and *repellors*, sets of points towards which the dynamical system tends and from which it escapes, respectively. In broad terms, once again, Harmonic Grammar, *some* versions of Optimality Theory (most notably, *bidirectional OT*), the CLASH model, Radical Minimalism, and Survive-Minimalism; are representatives of tension-based systems to different extents (the most extreme cases being CLASH and RM, which are also those with a stronger mathematical / physical basis)[82]. For these theories, the notion of *constraint* is essential: however formulated, constraints interact with each other and configure a system of their own, which in our proposal is embedded in the topology of the phase space itself (with its limits defining hard conditions for the dynamics). What does the evolutionary scenario look like for such a theory? Quite the mirror image of the one briefly sketched above, at least in the specific *tension-based* system explored in this thesis. Instead of building structure monotonically from

---

[82] It must be stressed that, unlike strict function-based systems, the notion of *mapping* is not a crucial component of interactive, tension-based models. Consider McCarthy's (2010: 2) definition of 'a' generative grammar (not quite Chomsky's, by the way, but closer to what the developments of transformational models have been historically) and its relation to OT:

> *A generative grammar is a mapping between two levels of representation. Is this mapping direct or indirect? A common answer in both phonology and syntax is that the mapping is indirect: there are intermediate steps in a derivation. In Optimality Theory (OT), however, the standard answer to date has been that the mapping is direct (Prince and Smolensky 1993/2004). Parallel OT, as I will refer to this theory, relates underlying and surface representations* ***without intermediate steps***. (Our highlighting)

In our opinion, not all versions of OT are bound to either the notion of 'level of representation' or the notion of 'mapping'. Crucially, a Harmonic Grammar *can* solve the tension between phonological and semantic (-oriented) constraints in a non-transformational way, without appealing to a concept of mapping. This is desirable, insofar as mappings are, by definition, functions.

below –as with Merge-, we start with an undefined phase space, corresponding to the syntactic engine in the simplest possible sense: an associative space representing the sensoria (Saddy, 2018), a kind of map of the world that is very likely shared with other species. Why an 'undefined' space? Because so far we have introduced no representation or relation between entities, all we have is the *potentiality* of relating elements (an *indefinite* number of them, in possibly more than one kind of computational way as argued in Krivochen, 2015a, 2016d; Krivochen and Schmerling, 2016; Bravo et al., 2015, 2017), but no way of getting usable outputs due to the lack of mechanisms that allow us to *categorize*. If a phase space is the set of all possible solutions to a system of equations, we need to have variables as well as relational operations (think about algebra, for instance: abstract though it may seem, it has arguments and functions, both defining the set of possible solutions for each equation). The syntactic phase space, too underspecified to be of any use on its own, is orthogonal to at least two other phase spaces (assuming an oversimplified but intuitive architectural model for natural language, namely, syntax + phonology + semantics). We will revisit this model in **Part III**, but as a transition, this should do. This scenario should make sense for those readers familiar with linear algebra: when we are presented with a system of *n* equations, the solutions for each determine planes which are orthogonal to each other in an *n*-dimensional space, and the solution to the system as a whole is defined in the intersection of all relevant planes (see Hefferon, 2014 for an introduction). The relevant planes we are dealing with are, given the simplified picture we assume (which, in the context of this work, excludes cultural and wider psychological considerations, as they are not directly relevant for the computational scenario we are presenting), the *phonotactic space* and the *conceptual structure space*, orthogonal to each other and to the space defining possible syntactic relations. The phonotactic space defines the set of possible sounds and its interface with the syntactic space, the possible sound combinations. The conceptual structure space defines what is *thinkable* and its interface with the syntactic space, all possible relations between entities, either sortal or eventive. In formal-grammar terms, it is interesting to notice that the simplest non-unary grammar we have come across, the Fib-grammar, defines an infinite phase space. This is trivial, since there are no constraints: there are no outputs, because the process never halts, formally (if a plant stops growing it is because of biological reasons related, for instance, to the availability of nutrients or the length of distribution channels and the amount of energy required to make use of these). We get output strings when we have external constraints over what the system can formally do, both in terms of the *form* of the output and their *content*. The thesis we put forth here is strong: *without a dynamical frustration at the core of cognitive processes, that determines mutually conflicting constraints over the form and the content of outputs of interactive routines* (which we distinguish purely for methodological reasons, we hasten to add), *there would be no high-level outputs, like language*. The system would either settle to equilibrium or grow without bounds.

Phonotactic spaces (unlike conceptual structures and the topological spaces in which they are constructed) have been the object of thorough research and rigorous formalization (e.g., within OT and *harmonic grammar* approaches). Heinz (2007) explicitly argues that patterns of natural languages are determined by learning mechanisms, which helps determining a phase space for phonotactics insofar as the learning mechanisms (or, optimally, *mechanism*) are equally explicit:

> *The learning mechanism itself constrains the available hypothesis space in a nontrivial way. Because on one reading, this hypothesis is logically necessary, it is possible to*

> *misread this hypothesis as trivial. However, once we get beyond the logical necessity of a restricted hypothesis space, the strength of the thesis becomes clear* (Heinz, 2007: 1)

If the possible learning mechanisms are determined by the physical configuration of the neurological substratum (assuming that the computational system, whatever a particular theory assumes it comprises, is an emergent property of analog neural interaction via electro-chemical signals), then the phase space of all possible solutions to the phonotactic problem is also constrained by that physical configuration. Learning is not within the range of topics we will deal with here, but we will make reference to the geometry of the relevant phase spaces, insofar as nontrivial properties of the derivational and interpretative dynamics arise only to the extent that they are possible assuming a certain topology for symbolic representations. We will deal with this in detail below.

2.7 The neurological perspective: Coordination Dynamics (CD)

The interactive perspective as we have presented it above has its parallel within neurological studies not only in Dynamical Field Theory (to which we will return below in detail), but also in so-called Coordination Dynamics (Kelso, 1995; Tognoli and Kelso, 2014; Schöner and Kelso, 1988; among others), a theory originally grounded in the physics of self-organizing nonlinear dynamical systems. This view analyzes the brain as an open system, in which '*many components have the opportunity to interact*', and, crucially, '*some ordering in space and time emerges spontaneously due to self-organization*' (Tognoli and Kelso, 2014: 35). Interestingly for our purposes, the model includes dynamics in multiple dimensionalities, claiming that '*pattern formation and change may take the form of lower dimensional dynamics*' (*Op. Cit.*) due to self-organization, a point we will tackle from the perspective of topological operations over manifolds in **Parts III** and **IV**. This view is compatible with the oscillatory computational system we argue for, implementing Lasnik & Uriagereka's (2012) 'dynamic flatness' in local domains: flattening structure amounts to finding a low-dimensional condensation of that structure which expresses the core properties of the complex structure or structures involved in the mapping to a lower-dimensional space, in essence (Saddy, 2018).

Tognoli and Kelso (2014) focus on the notion of oscillator, very familiar to physicists; this concept is a cornerstone of CD which deserves to be highlighted:

> *Coordination Dynamics builds upon the fact that oscillations and cycles are ubiquitous in nature* (Tognoli and Kelso, 2014: 36)

The relationship between our proposal here and the insights from CD will be obvious once we have derived the notion of *cycle* (and thus *syntactic island*) from the essential physical concept of *dynamical frustration* below. For the time being, we want to stress that CD cannot be formalized by means of function-based computation, because interaction and continuity are essential properties of the brain as assumed in this model. The interaction between different dynamics (represented by oscillators at different frequencies) is not determined *a priori*, but rather arises as the system develops in time. Kelso et al. (1990) showed that oscillators with different intrinsic frequencies can temporarily synchronize and affect each other's development (see also Tognoli and Kelso, 2014: 37). The appearance of temporary attractors that affect the development of the system and then disappear is a point of contact with Dynamical Field Theory (DFT), in which the disappearance of an attractor implies a reduction in the system dimensionality (a central point in our discussion of structure linearization and the emergence of strong cyclicity in **Part IV**, see also Saddy, 2018).

While some models of oscillators appeal to closed systems for simplicity (e.g., Varela et al., 2001, in connection with enactive cognition proposals and their neural implementation), Tognoli and Kelso argue that such systems are not encountered in nature, where *open nonlinear systems* are the norm rather than the exception, and

> *dynamic processes like "perceiving", "attending", "remembering", and "deciding" are not restricted to particular brain locations, but rather emerge as dynamic patterns of interaction among widely distributed neural ensembles, and in general between human beings and their worlds* (Tognoli & Kelso, 2014: 44)

The interaction between brain and environment, we see, is a common feature of CD and embodied cognition. Moreover, the CD model takes different temporal and spatial scales into account, arguing that '*Neural ensembles must work in a coordinated fashion at different spatial scales*' (we do need to say that Varela et al. 2001 distinguish between reciprocal connections at the same level of a neural network and connections across levels –*feedback* and *feedforward* respectively–). These authors focus on the concept of 'metastability', explicitly arguing that

> *Neural ensembles are continuously engaged in multiple interactions, and changes occurring at any place in the network may ripple through and affect each and every ensemble, with timescales that span from near instantaneous to long, and with coupling strengths from strong to minimal* (Tognoli and Kelso, 2014: 40)

The variety of temporal and spatial scales is obvious here. Mathematically, the states of the system can be modelled by means of ordinary and partial differential equations, depending on the number of variables. We take this to be an advantage of their model, particularly in reference to its compatibility with our own proposal and those close to ours (Saddy, 2018): as the reader will see below, ordinary and partial differential equations will play a crucial role in our mathematical model of mental computations.

It must be highlighted that the CD perspective focuses on neurological and functional evidence rather than on computational arguments, therefore, it cannot be seen as a replacement of any kind of the Interactive Computation paradigm; rather, it should be seen as a *complementary* approach, with its focus set on implementational aspects. Tognoli and Kelso's predictions (as well as Varela et al.'s large-scale integration of coordinated oscillations) involve the behavior of the brain in its internal dynamics (relations over the relative phase of oscillators modelling different aspects of the brain response to specific stimuli, like light pulses), but they do not make specific predictions with respect to the computation of specific pieces of data (for instance, specific kinds of linguistic constructions, or particular visual or auditory patterns). Such testing is also outside the scope of the present work, and our focus as far as empirical data is concerned will be focused on locality phenomena in language (involving structurally opaque syntactic objects known as 'islands' since the seminal work of John Robert Ross, 1967; these will be the object of analysis in **Part IV**): we will introduce the conditions under which *cycles* (or 'local domains') appear in physical structures and try to find such cycles in language. Crucially, the model of cyclicity we have in mind is heavily grounded in the concept of *oscillation*, which characterizes, we argue, not only neurodynamics but also computational dynamics, which is not surprising if we consider the evidence that that computation emerges from neurological oscillations at the mesoscopic level (Freeman, 2000: 11; Saddy, 2010). Oscillatory behavior might thus be a good starting point when attempting to characterize the relation –crucially, not the *reduction*- between formal operations (derivations and representations) and their neurological underpinning, which is characterized by nonlinear dynamical oscillations of electric and magnetic fields (see, for instance, the account of dissipative structures at a cellular level, and gene regulation, in terms of coupled oscillators – 'biochemical oscillators'- in Tzambazakis, 2015: 7-8).

As a complement to CD, the model of computation we argue for is based on the interaction between different components, and is thus better prepared than static, function-based models to model the kind of cognitive/computational processes we will consider, particularly under the light of a modified version of Uriagereka's (2012, 2014) CLASH model. We will consider emergent properties in language, as a complex system (see also Saddy and Uriagereka, 2004; Beim Graben et al., 2004, 2008), and inquire into the *reason* why these properties arise. In doing so, we will link language to other physical systems. But, before that, some more background is necessary in order to establish the niche our theory is to fill: the next section will analyze the role of *functions* in contemporary syntactic theory and some historical antecedents; and the consequences of assuming functions at the core of linguistic computations for both the theory and its empirical consequences. We will see that some shortcomings of contemporary syntactic theory follow from its adherence to structural uniformity as a consequence of the CTM's adherence to a function-based conception of linguistic computation.

2.8 Aspects of Functions in the Theories of Syntax

In this section we will review how function-based computation has been taken pretty much for granted in syntactic theory (including, but not limited to, MGG), and some of the concomitant theoretical and empirical problems related to this assumption (more often than not an implicit one). The assumption goes back to the birth of transformational syntax as we know it, roughly identifiable with Chomsky's 1955 *The Logical Structure of Linguistic Theory* and the 1957 *Syntactic Structures* (although earlier work by Zellig Harris is just as important in this respect). This birth could not have taken place without the influence of two (computationally oriented) mathematical works, which–and this is crucial–*were not concerned with natural language in the least*: Turing (1936) on the one hand, and Post (1944) on the other[83]. Both works, primarily focused on computation, as we have seen, relied on the notion of strict function-based algorithms, and were concerned with problems that only tangentially touch on natural language. Let us consider some explicit examples of syntactic theory that depend on function-based computation, and then proceed to point out some basic problems with their foundational assumptions. This will not only be crucial for theoretical purposes, but also impacts greatly on the model of the underlying mind-brain for a particular computational system. That is, the answer to 'what kind of neural and cognitive mechanisms do we need in order to have such-and-such computational properties?' depends on the kinds of computations we assume take place, as well as the properties of the representations, operations, and workspaces assumed.

Let us proceed historically. Consider the following quotation from Chomsky (1959):

> *Since any language L in which we are likely to be interested is an infinite set, we can investigate the structure of L only through the study of the finite devices (grammars)*

[83] It is almost otiose to say that this is a historical oversimplification, but not as much as the reader might think. The formal apparatus of *Syntactic Structures* and also *The Logical Structure of Linguistic Theory* is pretty safely describable as a watered down adapted version of Turing's formalisms, along with Post's mathematical insight, see, for instance, the Post Tag System (a variant of FSM). The lesser-known *Systems of Syntactic Analysis* (Chomsky, 1953) presents a more sophisticated formalism, although equally overly restricted for purposes of natural language analysis.

which are capable of enumerating its sentences. A grammar of L can be regarded as a ***function*** *whose range is exactly L.* (Chomsky, 1959: 137. Our highlighting)

We can already see here some characteristic traits of the early versions of the Standard Theory (and its extensions), most notably, the characterization of a language as an infinite set, the range of a grammar which is conceived of as a function. This function recursively enumerates the structural descriptions for strings in a language L (see Post, 1944: 285-286). The further characterization of this range as *exact* is due to *constructivist desiderata*: a grammar must generate structural descriptions for *all and only* grammatical sentences, that is, it must be satisfied (as a function) *only* by grammatical strings. The conception of *transformations* developed in Chomsky (1955: Chapter IX) puts emphasis not only on the fact that a grammar for a language L should generate exactly the range of the relevant function and nothing else, but also that *transformations* over a string are functions with a single output, such that a transformation T applied to a string S_i, $T(S_i)$, yields a unique output K_i, such that $T(S_i, K_i)$ is identified by means of a unique integer (a conception that would hold during the *Aspects* model). The connection to the definition of mathematical *function* that we provided above is obvious. Further developments of Chomskyan theory included more elements, and the establishment of relations between them, as we can see in the following quotation (which we already introduced above):

We must require of such a linguistic theory [a Generative Grammar] *that it provide for:*

(i) an enumeration of the class $S_{1'}$ $S_{2'}$, … of possible sentences

(ii) an enumeration of the class SD_1, SD_2, … of possible structural descriptions

(iii) an enumeration of the class G_1, G_2, … of possible generative grammars

(iv) specification of a ***function*** *f such that $SD_{f(i,\,j)}$ is the structural description assigned to sentence S_i, by grammar G_j, for arbitrary $_{i,j}$*

(v) specification of a ***function*** *m such that m(z) is an integer associated with the grammar G, as its value (with, let us say, lower value indicated by higher number)* Chomsky (1965: 31. Our highlighting)

This quotation, which belongs to the Standard Theory stronghold *Aspects of the Theory of Syntax*, makes use of the concept of *function* in two senses: first, it maps the set of sentences to the set of structural descriptions, and this mapping is the grammar itself (point *(iv)*); then, it specifies a monadic function that takes G_n as its value. Some features are shared by the two definitions: the specification of the set of 'possible sentences' and the set of 'possible structural descriptions' is one of the major issues here. While the set of structural descriptions is *not* in a bijective relation with the set of possible sentences if we consider PSR of the kind [Σ, F] (since a single PSR can in fact weakly generate strings of unlimited length), this is not explicitly stated. Another point that is not made explicit is that, under the strict constructivist formalism of ST and the Extended Standard Theory (EST), the set of 'possible sentences' is identical to the set of *grammatical sentences*, an equation that is not without empirical consequences when it comes to devising formal tools to characterize *competence* (Chomsky, 1965: 3, ff.). This should not come as a surprise, since PSR only generate well-formed formulae, thus sentences that are syntactically perfect. Semantics… well, that was a different story (and still is, unfortunately; see Partee, 2014 for some comments on the place of semantics within MGG all throughout its history). It is important to clarify in which respects sentences are *infinite* (this is a simplified version of the discussion, as we are not taking into consideration the cardinality of each sentence and other computationally relevant factors. See Langendoen and Postal, 1984; Langendoen, 2010 for technical discussion):

> (a) The set of grammatical sentences is an infinite set, as there cannot be a finite enumeration of all grammatical *sentences* (*not* structural descriptions) in any given natural language. This view follows from the assumption (foundational in generative grammar, but false) that '*a finite, rigurous characterization of a collection must take the form of an enumeration*' (Langendoen & Postal, 1984: 19)
>
> (b) Each member of the set is, potentially, infinite in length insofar as there are no *formal* boundaries (i.e., no ¢ / $ symbols, recalling our discussion of LBA automata) with respect to the number of elements involved in subsequent operations (this includes *head-tail* recursion as well as *center embedding* and *true recursion*, all mathematically defined).

Crucially, only (a) is a formally proven *fact* of language, (b) is a formal possibility which is not realizable by a human mind. Chomsky's early writings required each sentence to be of finite length (e.g., 1956: 114, 1965: 222), but this does not necessarily entail that structural descriptions are also finite in length, particularly in a theory that allows for unbounded deletion (see, e.g., Ross, 1970: 841; Langendoen & Postal, 1984: 16). As such, we regard it with skepticism: in any psychologically plausible view of language, sentences are *not* purely mathematical structures as they have *meaning* and *form* (in the structuralist sense, call them *semantics* and *morphophonology* if you want), and it is precisely those substantive characteristics that *limit* the power of syntax as a generative algorithm and give us language. We will come back to this below but, in a word, and unlike MGG, the *great evolutionary leap forward* was not the *addition* of elements to an otherwise impoverished substratum, addition that increased the computational and interpretative power of innate mechanisms (see Murphy, 2014 for a recent 'ethological' take on the matter; and Chomsky, 2005, 2007, 2008 for the orthodox 'biolinguistically oriented' MGG perspective that assumes the crucial 'rewiring' of the brain gave rise to the operation Merge), but the *limitation* of an initially unrestricted generative engine defining an equally unrestricted phase space (Saddy, 2018). This system, we contend, is to be identified with an L-grammar defining a weakly associative phase space in its simplest incarnations (e.g., plant cognition) by its intersection (*qua* topological space) with the orthogonal planes of sound and meaning, which limit the phase space of the L-grammar, trigger the emergence of crucial elements like *labels* (which in turn allow for *abstraction* and *categorization* of the sensoria), and give rise to a tension that is very much present at the heart of human cognition (Uriagereka, 2012: Chapters 6 and 7).

The centrality of the concept of *function* was maintained even in those theories that arose during the '70s and '80s as potential alternatives to transformational MGG: Lexical Functional Grammar and Generalized / Head-driven Phrase Structure Grammar. Let us consider them briefly in turn:

Lexical Functional Grammar is a generative, non-transformational theory that was initiated by Joan Bresnan and Ronald Kaplan in the '70s, originally inspired by work on Augmented Transition Networks and the possibility of eliminating transformations that arose after Emonds' (1970) seminal work on *structure preservation* conditions on mapping operations. Like most non-transformational theories, it is strongly lexicalist, and works with levels of structure, mainly *c-structure* (constituent structure, the level of syntactic constituents and their relations) and *f-structure* (functional structure, the level of grammatical functions like Subject, Object, etc.). Over the years, more levels of structure have been proposed (argument-structure, morphological-structure, etc.), but they are not related by means of transformations or mapping operations; rather, they constrain each other. Aspects of syntactic structure like *diathesis* are represented in lexical entries rather than being the product of applying a structure mapping rule to a certain phrase marker. *C*-structures have the form of context-free PSR (see Berwick, 2015 for a comparison between CF LFGs and CF Minimalist Grammars), whereas *f*-structures are sets of (attribute, value) pairs, where each attribute can be mapped to more than a single value (e.g., the [person] attribute can be mapped to the values [1^{st}], [2^{nd}], and [3^{rd}]). *F*-structures are the center of our attention in this particular case (since we have already seen the possibilities and limitations of context-free languages):

> *An f-structure is a* ***mathematical function*** *that represents the grammatical functions of a sentence* […] *all f-structures are functions of one argument* (…) (Kaplan and Bresnan, 1982: 182-183. Our highlighting)

It seems clear that, if grammatical functions (Subj, Obj…) are represented in *f-structures*, and *function* is a central concept in the definition of *f-structures*, then functions, in the mathematical sense, are also of central importance for the theory. There are two linking functions in LFG: one (call it α) links *a-structure* to *f-structure* and is universal, the other (call it ϕ) links *c-structure* to *f-structure* and depends on certain properties of languages (e.g., endocentricity vs. lexocentricity); these functions are isomorphisms. Kaplan and Bresnan (1982) provide a fully explicit model of LFG and an analysis of the formal architecture, situating the generative power of their theory between Type 2 and Type 1 languages. In essence, then, a LFG is a mildly-context-sensitive grammar which comprises several mutually constraining levels of representation, eliminating mapping functions of the kind of Move-α from the picture. In this light, apart from general problems with functions and natural language, let us see some specific difficulties that arise within an LFG framework, which explicitly assumes a context-free (or mildly context-sensitive) version of X-bar theory as '*a subtheory of LFG c-structure*' (Nordlinger and Bresnan, 2011: 118), but admitting exocentric, flat structures for the case of non-configurational languages (Austin and Bresnan, 1996). Functional and constituent structure (*f*- and *c*- structures) are constrained by principles over the outputs of the functions that map feature specifications to phrase structure trees, including conditions over the form of the trees (e.g., all nodes are in principle optional; the principle of 'lexical integrity') and the feature matrices (e.g., every attribute has a unique value; every function designated by an *n*-ary predicate must be present in the *f*-structure of that predicate). LFG, particularly in its early incarnations, made very explicit claims about the computational power of the grammar. Consider the following quotation from Kaplan and Bresnan (1982) concerning the decidability problem for a string *s* and a particular LFG (roughly, a set of parallel structures and a set of correspondence functions between such structures; Nordlinger and Bresnan, 2011: 112):

> *for any lexical-functional grammar G and for any string s, it is decidable whether s belongs to the language of G* (Kaplan & Bresnan, 1982: 267)

In short, this is an attempt at solving the *Entscheidungsproblem*, the 'decidability problem'. In very basic terms, the problem, articulated by David Hilbert, asks whether an algorithm that takes a proposition derived from a set of axioms (sometimes limited to first-order logics, but not necessarily) can specify whether the proposition is valid for every system complying with the axioms. The question whether a string *s* is accepted by an automaton A is decidable for FSAs, and the question whether it belongs to *L*(*G*) is decidable for PDAs (more generally, every CFL is decidable). However, for more powerful devices, the problem is not solvable.

HPSG (Pollard and Sag, 1994) is another non-transformational model, which empowers the Lexicon instead of multiplying the levels of representation. Lexical entries are highly specified, and the model can be pretty accurately defined as strongly *endoskeletic*, insofar as phrase structure is fully dependent on lexical specifications, in the form of (Attr, Val) sets. (See also Ristad and Berwick, 1989 for a computational perspective on feature-based Agreement-driven CFG, in which the decidability problem also arises. Ristad and Berwick, 1989: 386 claim the problem is NP-hard –not solvable in nondeterministic polynomial time-, which is for all ends and purposes the same as 'computationally intractable'). Unlike LFG, however, HPSG has lexical entries that contain several more features, including so-called *slash* features for displaced constituents. Basically, though, a word's lexical entry consists of two kinds of features: PHON and SYNSEM (the latter, in turn, decomposed into syntactic and semantic information, c(ategorial)- and s(emantic)-selection, among other things). A typical lexical entry in HPSG looks like this:

67)

$$\begin{bmatrix} \textit{word} & \\ \text{PHON} & \left\langle \text{`walks'} \right\rangle \\ \text{SYNSEM} & \begin{bmatrix} \textit{synsem} & \\ \text{CAT} & \begin{bmatrix} \textit{category} & \\ \text{HEAD} & \textit{verb} \\ \text{VALENCE} & \begin{bmatrix} \text{SUBJ} & \left\langle \begin{bmatrix} \textit{synsem} & \\ \text{CAT}\,|\,\text{HEAD} & \textit{noun} \\ \text{CONT} & \boxed{1} \begin{bmatrix} \textit{ref-index} & \\ \text{PER} & \textit{3rd} \\ \text{NUM} & \textit{sing} \end{bmatrix} \end{bmatrix} \right\rangle \\ \text{COMP} & \langle\rangle \end{bmatrix} \end{bmatrix} \\ \text{CONT} & \begin{bmatrix} \textit{content} & \\ \text{WALKER} & \boxed{1} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

In relation to the projection of lexical information onto phrase structure, Müller (2016) claims that

> *The HPSG lexicon* […] *consists of roots that are related to stems or fully inflected words. The derivational or inflectional rules may influence part of speech (e.g. adjectival derivation) and/or valence (-able adjectives and passive)* […] *The stem is mapped to a word and the phonology of the input* […] *is mapped to the passive form by a* ***function*** *f.* (Müller, 2016: 16. Our highlighting)

The sentence-level displays the same format as the word-level, at a different scale (taken from Green, 2011: 24):

68)

$$
\begin{bmatrix}
\text{PHON} & \langle \textit{gives}, \langle \textit{a}, \textit{book}\rangle, \langle \textit{to}, \textit{Sandy}\rangle\rangle \\
\text{SYNSEM}\,|\,\text{LOCAL}\,|\,\text{CAT} & \begin{bmatrix} \text{HEAD} & v \\ \text{SUBJ} & \boxed{4}\left\langle \begin{bmatrix} \textit{synsem} & \\ \text{HEAD} & n \\ \text{SPR} & \langle\ \rangle \end{bmatrix}\right\rangle \\ \text{COMPS} & \boxed{5}\ \langle\ \rangle \\ \text{SPR} & \boxed{6}\ \langle\ \rangle \end{bmatrix} \\
\text{HEAD-DTR}\,|\,\text{SYNSEM}\,|\,\text{LOCAL}\,|\,\text{CAT} & \begin{bmatrix} \text{SUBJ} & \textit{App-synsems}(\boxed{1}, \boxed{4}) \\ \text{COMPS} & \textit{App-synsems}(\boxed{2}, \boxed{5}) \\ \text{SPR} & \textit{App-synsems}(\boxed{3}, \boxed{6}) \end{bmatrix} \\
\text{SUBJ-DTRS} & \boxed{1}\ \langle\ \rangle \\
\text{COMP-DTRS} & \boxed{2}\ \langle \text{NP, PP}\rangle \\
\text{SPR-DTRS} & \boxed{3}\ \langle\ \rangle
\end{bmatrix}
$$

In this respect, Green explains that

> *This analysis* [Pollard and Sag's, 1994; above] *employs an App(end)-synsems* ***function*** *that appends its second argument (a list of synsems) to a list of the synsem values of its first argument (which is a list of phrases).* (Green, 2011: 24. Our highlighting)

It has been claimed that the *slash*-structures managed in HPSG, in which a constituent A containing a gap of a constituent B is notated as A/B (using notation originally devised by Ajdukiewicz, 1936 and adopted in Categorial Grammars, but with a completely different meaning) employ CF rules (Berwick, 1984, 2015), but in practical terms, HPSG articles make no use of classical rewriting rules, focusing on lexical entries instead (and having trees as mere notational variants of detailed feature matrices like (68)). In this sense, HPSG is a strongly representational theory, insofar as we can know pretty much everything we need to know by looking at a fully specified lexical entry. A clue with respect to the size of the grammar is given by the non-reality of traces (which in a GB-like model *have* to be 'real' at some non-metaphorical level, due to some local processes involving chains containing at least one trace or deleted copy of a syntactic object): HPSG is heavily focused on lexical insertion, accounting for gap phenomena by reference to a 'non-canonical' subtype of SYNSEM feature (Kathol et al., 2011: 70) and the ARG-ST (Argument Structure) list, which, crucially, remains unchanged even if there is an 'extracted' constituent. Thus, there is actually no copy + deletion (+ indexing) process involved (which implies an exponential reduction in the size of the grammar with respect to transformational models), *but* information concerning gaps is percolated up the structure, such that a sentence like (69) receives a context-free phrase structural representation like (70), following Berwick (2015: 7) (traces are added to (69) for expository purposes only)

69) [Which violins]$_i$ are [these sonatas]$_j$ difficult to play t_j on t_i?

70) CP → *wh*-NP$_1$ (*which violins*) S/NP$_1$; S/NP$_1$ → NP$_2$ (*these sonatas*) S/NP$_1$NP$_2$; S/NP$_1$NP$_2$ → (*pro*) VP/NP$_1$NP$_2$; VP/NP$_1$NP$_2$ → V NP$_2$/NP$_2$ PP/NP$_1$; NP$_2$/NP$_2$ → ε; PP/NP$_1$ → NP$_1$/NP$_1$; NP$_1$/NP$_1$ → ε;

While the context-free [Σ, F] form is not the chosen way to express an H/GPSG representation (recall that the theory is strongly representational and heavily based on information encoded in lexical entries, in contrast to the strongly derivational character adopted by recent Chomskyan formalisms), and the empty terminal ε is an added symbol to indicate the halt of the computation (i.e., ε does not get rewritten), we can indeed see that the information about NP gaps is carried up the syntactic structure, creating a decision point at each nonterminal node. As Berwick (2015) points out, the existence of *slash*-categories in the HPSG usage (but *not* in the original CG sense, importantly) implies an exponential growth of the structural description's minimal description length. The context-free phrase structure representation in (67) *exemplifies* that, but as we have said, HPSG does not use derivational tools like PSR in their structural descriptions, they rather enrich their representations than their derivations.

Next on the list is the most recent incarnation of transformational generative linguistics, the Minimalist Program (1995-?). We have assumed a certain familiarity with this framework (which is a reasonable assumption given the fact that it is widely taught and perhaps even more widely published). Therefore, let us go directly to the relevant citations from the creator and main proponent of the MP, Noam Chomsky. In the foundational text of the Minimalist Program, Chomsky claims that:

> *We take L* [a particular language] *to be a generative procedure that constructs pairs (π, λ) that are interpreted at the articulatory perceptual (A-P) and conceptual-intentional (C-I) interfaces* (…). (Chomsky, 1995: 219)

These pairs are created via the application of two kinds of rules to *terms*, which can be *terminals* (items from the Lexicon) or *non-terminals*:

- Structure building: take distinct objects α and β, *Merge* them, and build {γ, {α, β}} with γ = α or γ = β (γ the *label* of {α, β}), always *extending* the phrase marker. Furthermore, *Merge* leaves α and β internally unchanged.

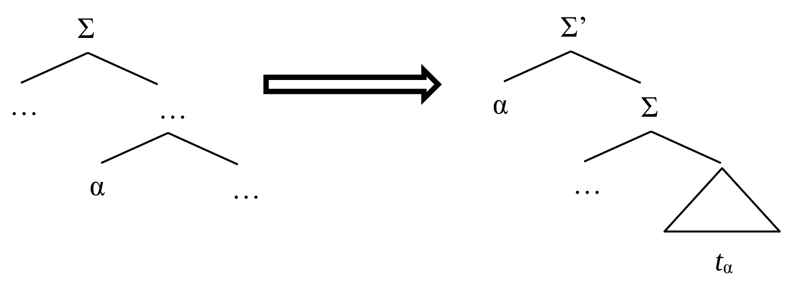
- Structure mapping: since the mid-'70s, transformational generative grammar has a single transformational rule, *Move-α*. In the MP, Move is a variant of Merge, such that structure building takes elements from the Lexicon and Move takes elements from within the active derivational space. Applying to Σ containing α (where α can be either a terminal or a nonterminal), Move creates Σ' by copying α and re-Merging this copy to the root Σ (in general, the lower occurrences of α are referred to as *traces*). Kitahara (1997: 24) formulates the operation in its cyclic version as follows:

  Input: Σ (containing α)

  Concatenate α and Σ, forming Σ'

  Output: *Σ'*

  In tree form:



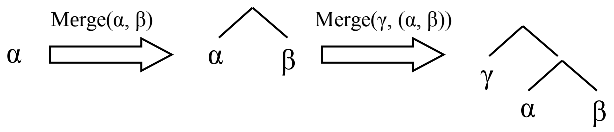
While the possibilities for structure building are, in principle *unconstrained* (‘*n*-ary’ trees), the MP is constrained by the stipulations of X-bar theory (which took form in Chomsky, 1970, being significantly expanded in Chomsky, 1986a), and allows only binary branching (for both Spec-X’ and X-Complement positions). Thus, *Merge* always applies to *two* objects, either at the root or at a non-root. During the ‘90s, Kayne (1994) attempted to reinforce the binarity requirement he had put forth in the ‘80s (Kayne, 1984), and came up with a linearization ‘algorithm’ for phrase markers known as the *Linear Correspondence Axiom*:

> *phrase structure* (…) *always completely determines linear order* […] (Kayne, 1994: 3)

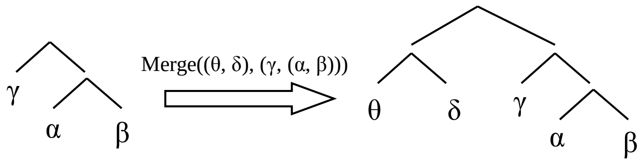
> *Linear Correspondence Axiom:* [LCA] *d(A) is a linear ordering of T.* [A a set of non-terminals, T a set of terminals, *d* a terminal-to-nonterminal relation] (Kayne, 1994: 6)

d is the syntactic relation known as *asymmetric c-command*. Quite transparently, then, linear order is a *function* of phrase structure[84], and phrase structure building is always a *two-place* operation, mapping α to {α, β} via the function *Merge* (Chomsky, 1995, 2013, 2015). α and β, in turn, can be atomic (taken from a Lexical Array), or not, which gives rise to two computationally different types of *Merge*: Monotonic and Non-Monotonic, respectively (see Krivochen, 2015a for extensive discussion and Uriagereka, 2002a for the original presentation). Let us illustrate both derivational options:

71)



72)



If the LCA is interpreted in all seriousness as a real linearization algorithm over objects whose nature is anything more than intra-theoretical or formal (but is rather, e.g., neurocognitive, attending to Chomsky’s desire to study the ‘Faculty of Language’, and the claims of the ‘biolinguistic’ enterprise), then (72) is a problematic phrase marker: there is no way to linearize {θ, δ} with respect to {γ, {α, β}} unambiguously, since there is a point of symmetry (i.e., mutual c-command) between the minimal nonterminal that dominates {θ, δ} on the one hand, and {γ, {α, β}} on the other. It is thus impossible to satisfy Kayne’s function *d*(*A*) globally. Here is where Uriagereka’s Multiple-Spell-Out model comes into play. However, that model’s increased theoretical coherence comes at a price: assuming Spell-Out can take place more than once in the course of a derivation *and* that the system in charge of ‘sending’ chunks of structure for their linearization is sensitive to points of symmetry (or at least to finite-state compatible sub-units, in Uriagereka’s more recent developments, see e.g. 2012, 2014). None of this has been explicitly formalized under an MGG approach, and the ontology of the LCA and the objects it allegedly applies to is still unclear.

[84] In this sense, we could express Spell-Out as a function *f*(SObj), for any syntactic object SObj, as a mapping of SObj = {α, {β, {γ…}}} to SObj’ = $\alpha^\frown\beta^\frown\gamma^\frown\ldots$, ($^\frown$ being a transitive, antisymmetric relation of linear precedence such that $\alpha^\frown\beta$ = α precedes β) and if $x \in$ SObj is a non-terminal, then *f* applies to *x* internally as well, and in the same manner.

The role of the Lexicon in the Minimalist Program is problematic as well. According to Chomsky (1995) and much subsequent work, a derivation starts with a subset of the Lexicon, called a Numeration, which contains the terminals that will be used in a given derivation. Apart from highly problematic *Select* issues—to use GB terminology; that is, how the system knows which units to select from the Lexicon in order to build a Numeration for a convergent derivation that conveys what we want it to convey; see Stroik and Putnam, 2013 and Krivochen, 2015b for two different possibilities—there is a serious problem that affects both computational tractability and acquisition, which we have called the 'combination problem':

- The combination problem:

The 'combination problem' is a purely mathematical problem that, in our opinion, greatly undermines the plausibility of syntactico-centric theories (in the sense of Culicover and Jackendoff, 2005), which commit to the CTM: it is crucial to remember that the problems we have pointed out so far, and those we will discuss in what follows, arise because of the equation between 'effective computation' and 'function-based computation' in modern linguistic theory; it is not an assault on 'syntax'. Rather, we argue for a reconceptualization of what 'computation' is in the more general context of linguistic inquiry. Assume we have a Numeration containing *n* elements. All possible combinations of *n* using all *n* are defined by the expression *n*!—but that is *not* what MGG says. The binarity requirement forces us to use all *n*, yes, but in successive sets of *two* elements (Kayne, 1984, 1994; Chomsky, 1995, 2007, 2008; among many others). This is mathematically expressed by an extension of the factorial formula:

73)

$$\frac{n!}{(n-k)!k!}$$

where *k* is the number of elements we are manipulating at any given derivational point. Changing the notation, to make it linguistically more transparent, the formula would look like this:

74)

$$\frac{NUM!}{(NUM-D_i)!D_i!}$$

NUM is a specific Numeration; D_i is the number of elements of NUM to be used at a derivational point D_i, such that a derivation can be defined as a set S = {D_i, D_j, D_k, …, D_n} of subsequent states. All that formula is saying is that the possible combinations given a Numeration of *n* elements are defined not only by the possible combinations of *n* in sets of *n*, but by the combinations of *n* in sets of *k* elements ($n > k$), which are represented by derivational stages. Assume now the simplest possible Numeration for a sentence like [John loves Mary], considering only the basic functional categories[85]:

[85] We will assume here that proper names are DPs, since the procedural instructions conveying by D (sortal referentiality) apply equally to proper and common names (not too different from a Montague approach to nominal constructions). See Sloat (1969) for an early treatment, and Krivochen (2012: 61-63) for discussion within the present framework. Two Ds are assumed since we have two proper names, and arguably, the procedural instructions for each could vary (e.g., if we know John but not Mary, we are only able to *presuppose* her existence, in the formal semantic sense). We will also assume, just for the sake of the exposition, that Person and Number morphological features reside in the T node (and are in fact

75) Num = {C, T, *v*, D, D, John, √love, Mary}

By the formula in (73), there are 28 possible combinations, only two of which converge: [John loves Mary] and [Mary loves John], and only one of which actually conveys the intended proposition *love*(John, Mary). If branching side is not really relevant, we are left with 56 possibilities, of which the same two converge at the interfaces. In our opinion, the system is not very efficient in this respect, particularly given the fact that there is no way to guarantee the syntactic engine will not generate any of the 26 non-convergent possibilities without resorting to stipulative filters or equally stipulative feature-valuation considerations (see also Jacobson, 1997 for a critical assessment of a transderivationality approach to solving this problem). While semantically-oriented theories can provide sound explanations, as can OT-oriented harmonic theories, MGG's focus on narrow syntactic operations leaves it vulnerable to this kind of objection.

Computationally, this is not a minor problem, since we either need a multiple-tape TM (with one tape *per* derivation), or we evaluate all candidates sequentially until we find the one we have been looking for. Of course, this sequential process takes time, and there are currently no proposals we know of to prevent time from building up exponentially, particularly with long-distance dependencies or center-embedding. In terms of the possibilities of having a generative grammar (a recursively enumerable procedure) for (weakly generated) languages over a finite alphabet, Lewis and Papadimitriou (1998: 47) observe that

> *The set of all possible languages over an alphabet Σ –that is, 2^{Σ^*}- is uncountably infinite* (…) *With only a countable number of representations and an uncountable number of things to represent, we are unable to represent all languages finitely*

Also, Fraenkel (1993) nicely summarizes the complexity problem we would be facing:

> *The most simplistic approach to solving a problem is to explore its entire "search tree", i.e. searching through all possibilities. Except for trivial problems, this search constitutes an exponential algorithm. Thus, a problem whose best algorithm is exponential has often no essentially better algorithm than to search through all or most possibilities.* (Fraenkel, 1993: 1200)

In our opinion, any function-based model that aims at implementational adequacy should provide a solution for the checking problem: this involves, among other things, a theory of extremely local transderivational constraints. So far, to the best of our knowledge, there is no such thing in MGG or other function-based models of syntactic structure. Some authors have proposed conditions over the featural composition of elements, such that an object X merges with an object Y if there is some operation over the features of X and Y taking place, more often than not, *valuation* (Pesetsky and Torrego, 2007; Wurmbrand, 2014), although some proposals also consider a relation of proper containment between the set of features defining the target of Merge and the currently manipulated object (Di Sciullo and Isac, 2008; Panagiotidis, 2014). Those approaches, far from solving the problem, merely shift the burden: instead of checking competing derivations, we have to check feature combinations, which are, incidentally, arbitrarily assigned to syntactic heads.

---

inherited from C), a common assumption in MGG (Chomsky, 2007, 2008; Gallego, 2010) that we find stipulative at best.

- The uniformity problem:

The ‘uniformity problem’, which we have already referred to, is simply the resistance of MGG to thinking in terms other than binary-branching structure. In more technical terms, any strictly constrained generative algorithm has the problem of forcing all strings to comply with a single structural template (e.g., X-bar), which sometimes is ‘too much structure’, and sometimes leaves aside (is not sensitive to) relevant structural features that impact on semantic interpretation. In a word, MGG’s syntactic theory is quite like a Procrustean bed. Consider (76):

76) The old old old man

(76) is simply an iterative pattern, but we cannot know that only from the string: it is the [X…X…X] situation we dealt with earlier. In this kind of situation, we cannot say that one instance of the A takes the others as arguments, as a binarily-branching traditional X-bar theoretical representation would entail: that would impose extra structure on the phrase marker, in the form of silent functional heads (Cinque’s 1998 Functional Phrase, for instance, and more generally the rest of the ‘cartographical approach’, see Cinque and Rizzi, 2009), or embedded APs, [old [old [old]]], which not only complicate the theoretical apparatus (as we need independent evidence for the silent functional material), but also fail to account for the semantics of the construction, which is not scopal but incremental (see Uriagereka, 2008: Chapter 6 for discussion). For examples like (76), a *uniform* phrase structure grammar of the kind [Σ, F] like the one argued for in Chomsky (1957) imposes too much structure over the adjective stacking (adjectives being terminals that are dependent on a null nonterminal), assuming a strict system of binary branching. Chomsky himself (1963: 298) recognizes the structural overgeneration of phrase structure grammars:

> *a constituent-structure grammar necessarily imposes* ***too rich*** *an analysis on sentences because of features inherent in the way P-markers are defined for such sentences*. (our highlighting)

Curiously, there was no attempt within MGG to improve the phrase structure engine in order to include these simple dependencies, rather, examples like (76) and similar patterns were increasingly overlooked from the ‘80s on: the rise of X-bar theory as a uniform (and aprioristic) template for syntactic structure (Chomsky, 1970, 1981a, 1986a; Stowell, 1981), and its ‘binary branching’ requirement (Kayne, 1984; but antecedents can be found already in Katz & Postal, 1964) directly excluded from the picture *n*-ary branching dependencies, some of which correspond to very simple Markov chains. Minimalism’s fundamental operation *Merge* maintained the essentially binary character, ‘deduced’ from seemingly independent principles like antisymmetry (Kayne, 1994, 2011; Di Sciullo, 2011) and feature valuation operations involving at least and at most a Probe and a Goal, such that Merge always involves at least and at most two elements. Structural uniformity is one of the cornerstones of Minimalism, argued in MGG from Chomsky and Miller (1963: 304) (who first proposed binarity as a condition over generalized transformations) on, and, despite its allegedly economical character, it has concomitant empirical and theoretical problems which we have analyzed in detail in Krivochen (2015a, 2016d, 2018a); Krivochen & Schmerling (2016), and other works. Computationally, structural uniformity is the assumption that the same set of structural descriptions (which is, under Minimalist assumptions, possibly a binary set, comprising monotonic and non-monotonic Merge) is valid for all and every string of the language. The hypothesis about the neurocognitive substratum that this implies is clear: uniform phrase structure is part of the initial state of the

Language Faculty (i.e., UG), and therefore the neurocognitive substratum of language must prefer uniformity for some reason, which is often claimed to be biological or physical in nature (e.g., Hauser, 2009, fallaciously extrapolating from Bejan and Marden, 2006; as well as some speculation by Cherniak, 2009). By questioning the theory of phrase structure, the underlying assumptions about the neurocognitive substratum it requires (the verb is justified by the MGG assumption that properties of language are properties of UG, which is biologically –possibly even *genetically*- determined) are also questioned, and we have made a case for the necessity of mixed phrase markers, together with interactive models of computation (Krivochen, 2015a, b, c, 2014c).

- The implementation problem:

This problem is simple to enunciate but more difficult to explain, at least under traditional assumptions. In short, derivations are at odds with real-time processing. That is, the 'false topology' assumed in MGG (either top-down or bottom-up) is in any case at odds with the processing of the linear signal, which is left-to-right, following the speech flow. We have spoken of a 'false' topology because there is actually no theory about the characteristics of the topological space in which derivations take place, except perhaps some general considerations we can make based on Roberts' (2015) concerning ultrametric distances in X' theory. However, those considerations pertain to the structures themselves, not the spaces in which they are derived. Our concern, then, is twofold: we have questioned the empirical adequacy and theoretical status of structurally uniform phrase markers, and we are also concerned with the properties assigned to the working spaces in which these phrase markers are derived. What is more, we have seen that there are inconsistencies between the definition of a grammar as a procedure to recursively enumerate sentences of a language, and the possibility to do that with finite representations (Langendoen & Postal, 1984; Lewis & Papadimitriou, 1998); that, if a *proof-theoretic grammar* is the best tool to begin with (see Pullum & Scholz, 2001 for discussion and comparison with *model-theoretic* approaches). The name of the problem comes, quite obviously, from Marr's (1982) levels of analysis:

- *The computational level*: what the system does (e.g., what kinds of problems it can solve), and why
- *The algorithmic/representational level*: how the system does what it does, more specifically, what representations it uses and what processes it employs to build and manipulate the representations
- *The physical/implementational level*: how the system is physically realised (e.g., by means of what kind of neural circuitry)

Having a look at the kind of arguments put forth not only within the MP, but also in LFG, HPSG and related theories, we think that, while they all address the *what* in the computational level, and propose theories for the *how* (with an incredible attention to detail and formal rigor in LFG and HPSG), the *why* is often just stipulated (e.g., UG in MGG), and there is little, if any, concern with respect to the *implementational* level, even for so-called 'biolinguistics'. In short, theories of 'language' tend to neglect crucial aspects of the implementational level and stipulate the characteristics of the computational mechanism and its functioning, well beyond what we could expect within a formal axiomatic framework. From a point of view that departs from mathematics, constrained by physical laws, in turn realized in biological structures, we will attempt to cover all three levels, which is an epistemological improvement in and of itself.

So much for this brief review of functions in theories of syntax. So far, we have (a) described the properties of formal automata, (b) reviewed the arguments in favour of and against function-based models of computation, and (c) –hopefully- begun to make a convincing case for interactive models of mental computations. The next step, then, is to go deeper into this line of inquiry, pursuing one alternative that, we think, brings together cognitive, linguistic, and physical considerations in a coherent way: the 'dynamical frustration' approach. In order to present the model, we will first problematize some recurrent oppositions in the cognitive / linguistic literature, and attempt to provide a way out by means of a 'frustrated' perspective.

2.9 A Frustrating Unification: Overture

In this section we will attempt to unify four sets of concepts that regularly appear in the cognitive science and linguistic literature as mutually conflicting or incompatible, although rarely related to one another (with the exception of Uriagereka's 2012, 2014; Saddy's, 2017 works, among very few others):

77) a. Locality vs. Globality
    b. Morpho-phonological vs. semantic requirements
    c. Simplicity metrics vs. accuracy metrics vs. likelihood
    d. Linearity vs. Hierarchy

It should be obvious that only (77 b) appeals to narrowly linguistic concepts; (77 a), (77 c), and (77 d) are ubiquitous tendencies all throughout cognition.

To begin with, the opposition between local and global processes within cognitive science can be traced back to the debate between modularism and Parallel Distributed Processing (PDP): informationally encapsulated modules (of the kind described by Fodor, 1983; Pylyshyn, 1999), with specialized functions, automatic functioning, and localized neural basis, predict a strictly local effect of neurological damage, such that non-damaged neural components should keep computing normally (Farah, 1994 for discussion and references). In contrast, a PDP model (see Rumelhart and McClelland, 1986 and much subsequent work; Bowers, 2017 relates PDP with contemporary deep networks and critically analyzes the psychological assumptions of PDP under a modern computational light) is based on interaction rather than encapsulation and strict localization, and the representation of information is distributed over a population of neural units, some of which, if one assumes a model along the lines of Elman, 1993 and related work, are in charge of keeping firing patterns active for a certain period of time in order to relate inputs and outputs in a network, thus providing the neural network with the possibility if 'learning'. Such a network is typically distributed in three layers: Input–Intermediate (Copy)-Output. Models of the mind based on modules (and even more so recent 'massively modular' models of the mind, which reject the existence of a central processor, see Carruthers, 2006) almost necessarily need to resort to local procedures, internal to each module, and rest exclusively on information available intra-modularly at every time, in order to provide the central processor with an output. Given the property of informational encapsulation, the amount of information available at any particular derivational point in a computation (of whichever nature) for any particular module is extremely limited (which in turn guarantees that processes are quick, since the search space for each module is reduced). However, there is no formalization of the memory capacities of different modules, nor is it made explicit how many elements a module can manipulate at a time. While there are more recent models that attempt to take advantage of the theoretical claims and empirical results of both proposals, claiming that lower-level processes are modular, whereas high-level processes work in a different way, possibly closer to PDP modelling (even Fodor himself has adopted a related view, see 1983, 2000), the debate is far from settled, and the opposition between these theories seems to remain strong. In any case, we are concerned here not with the historical developments of the concepts, but with their content, and there is no conclusive evidence, to the best of our understanding, in favor of either a *uniformly* modular (symbolist) or a *uniformly* connectionist mind; both kinds of processes seem to co-exist, although at different levels of cognitive tasks (Carreiras, 1997; also Calabretta and Parisi, 2005 for an *evolutionary connectionism* approach that combines evolution and learning processes). We hope that the future of cognitive science can go beyond aprioristic associations (e.g., computation = functions; modularism = nativism, etc.) in favor of a more dynamic and flexible (and also more rigorous) approach (see Townsend and Bever, 2001 for an example of such a paradigm).

(77 c) above, simplicity vs. accuracy metrics, is particularly relevant for learning theories and proposals about how the cognitive system predicts the environment and identifies structures and patterns present there (Shirley, 2014: 2, ff.). Of course, the problem is not exhausted by learning considerations, and practically any model based on comparison of potential candidates and filtering of sub-optimal structures 'must' choose between one of the metrics to base their constraints on (e.g., Müller's, 2011: 9-27 presentation of Transderivational and Local Representational / Derivational constraints, which are clearly based on simplicity metrics sometimes following economy considerations present at the core of the Minimalist Program. See Jacobson, 1997, for an argument against the existence of economy-based transderivational constraints). Even those theories based on the derivation of well-formed candidates from square one (i.e., *constructivist theories*, see Lasnik and Uriagereka, 2005) incorporate a measure of simplicity at some point (e.g., *fewer steps*; *Minimal Link Condition*, among others). Going back to more general cognitive approaches, accuracy-based models of interpretation (also known as 'likelihood approaches', see Pomerantz and Kubovy, 1986) propose that '*sensory elements will be organized into the most probable object or event* […] *consistent with the stimulus*' (Pomerantz and Kubovy, 1986: 36-9, summarizing the view of Helmholtz, 1962), thus implying a competetition model among *n* candidates (objects or events) where the choice depends on which is most accurate with respect to the stimulus given a learning process, stressed in Helmholtz (1962). The conditional probability of a hypothesized distal layout H gives us the likelihood of a stimulus, given the sensory input D, thus: $P(H \mid D)$. More elaborated likelihood measures consider Bayesian probability, such that the 'perceptual system' (whatever that is), chooses the hypothesized distal layout *H* that best satisfies the Bayesian probability equations, basically expressible as a direct proportionality relation between posterior probability for an event A and its prior probability times the likelihood ratio, based on event conditioning (see, e.g., Gelman et al., 2013: Chapter 1). This organization of perceptual information is not conscious, and proceeds through *inference* (i.e., a context-sensitive computational procedure involving *n* premises, where $n \geq 2$, and a conclusion or inferred content). In contrast, simplicity metrics, based on Gestalt perceptual principles, also consider the possibility of many candidate representations, but the filter to be applied is *simplicity*, or, better put, *simplified global structure* (we are not considering here approaches that simply claim that '*what the system likes is **short** descriptions*', Attneave, 1981: 417, emphasis in the original; because there is no explicit and formal definition of 'simplicity' we can work with there). The core notion of this view is the so-called *Prägnanz* ('succinctness'), which is interpreted in three main ways:

a) Stimuli will be organized into the simplest possible *configuration*, even if this means distorting the percept with respect to the stimulus (thus violating *likelihood*)

b) Stimuli will be represented by the simplest, most economical *description* compatible with the physical input. Note that this interpretation focuses on the structural description of the stimulus, not on the organization of the information about it. But the structural description does not need to be accurate in order to win over other potential descriptions, just simpler.

c) Stimuli will be organized using the simplest possible *organization mechanism* (a view that is explicitly dismissed by Pomerantz and Kubovy, 1986: 8).

Other approaches to simplicity conditions (e.g., Attneave, 1959) consider *shorter encoding* measures, based on information theory insights (Shannon, 1948), and Kolmogorov complexity (recall that this refers to the length of the shortest computer program that produces the relevant object as output). These approaches are mathematically formalized (thus, they are unambiguous), and provide an algorithm for approaching brevity in terms of the number of *bits* (binary digits) necessary to distinguish the stimulus from a range of mutually exclusive candidates, in what is essentially a competition-based model. This is problematic insofar as it is not obvious that either the stimulus or the cognitive processing are in fact digital in nature. These approaches include a measure for 'surprise', based on the notion of *entropy*, assuming that there is a direct proportionality relation between informativity and the low probability of occurrence of a particular candidate. Interestingly, when reviewing the opposition between simplicity and accuracy, Chater (1996: 569) expresses some concepts of this alleged 'simplicity measure' in terms of likelihood, since there is a crucial notion of 'noiselessness' in the information theory-based approaches which actually has little to do with actual brevity, but rather acts as an anti-entropic proviso. We will come back to this below, when presenting our derivational system in **Section 3.5.2**.

Finally, (77 d), the tension between hierarchy and linearity, seems also to be pervasive in human cognition well beyond language; and in computational approaches to *formal* languages. In general, the problem arises both in production and perception: when perceiving any string, as in the […X…X…X] examples above, there is the question of whether there is a hierarchical relation between X's (and, more specifically, which kind of hierarchical relation is involved). But the problem of multiple structural descriptions available to assign hierarchy to a physically linear stimulus (as a string or as a plane) is not limited to sounds, it also arises in other faculties, most notably in vision, where depth perception is the result of assigning a figure-ground hierarchy to segments of the perceived stimuli. For example, let us consider the following image and four possible structural descriptions (a-d), taken from Pomerantz and Kubovy (1986: 36-4):

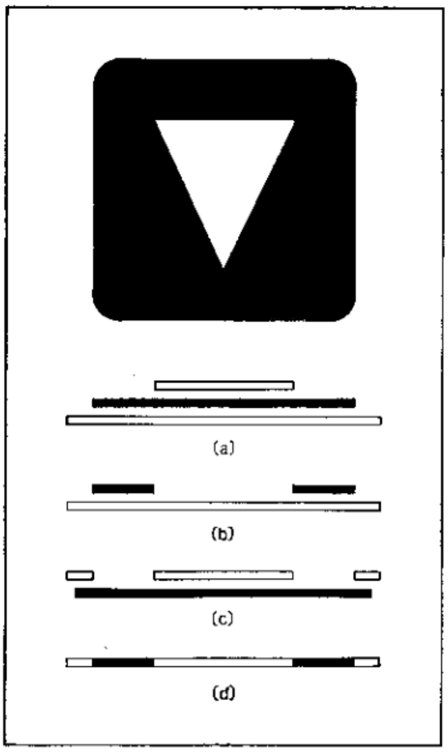


Figure 7: multistable figure-ground demonstration

The mere assignment of a figure-ground relation to the image is by itself the assignment of a hierarchical structure to a linear input (since there is no relative 'timing' for the triangle or the square to reach our eyes, it is an unstructured image until we assign a structural representation to it), but it is most notable that there are (at least) four possible 'parsings' of the figure, in decreasing number of subjects' responses (a) to (d). According to Pomerantz and Kubovy, most subjects assign the image a structure like (a) (personally, my first parsing was actually (b), but this intersubjective variation is contemplated in their article), in which there are three layers (a white triangle on top of a black background, in turn on a white paper), thus something like:

78) [X [Y [Z]]] (branching side being irrelevant)

Interpretation (b) is 'simpler' in terms of levels of embedding, but there is a discontinuous dependency on the *ground* (white), interrupted by the black shape (*figure*). The hierarchical structure would be along the lines of (79):

79) [X [Y] X] (where a dependency is established between the two instances of X)

However we parse it, the presentation of the image is strictly flat, complying with the [X…X…X] template (even though there is a Y element, the black shape, replacing one of the X's).

In production, things can be seen from a different (although complementary) perspective, depending on what theory of language the reader prefers (to narrow the focus to a single mental faculty). There is virtually no disagreement with respect to the hierarchical nature of internal language (there *is*, however, disagreement with respect to the *nature* of that hierarchy, and its place in language evolution and its relation to other cognitive faculties). Similarly, there is virtually no controversy with respect to the serial, 'flat' nature of externalized language as a physical signal (a fact already noticed by De Saussure, 1916: Principle II of the 'Nature of the Sign'), and the computational models that model such externalizations are different from those appealed to in order to account for other properties of language, like meaning or (more controversially) constituency / dependency. The specific procedures for 'flattening' a hierarchical structure for the purposes of language externalization are far from clear, and the question of whether there is actually such a procedure is not trivial: if constructions *qua* form-meaning pairs are stored as such in Long Term Memory (LTM), in agreement with Construction Grammar approaches (Goldberg, 2006; Croft, 2001), then the storage itself could be linear, hierarchy being a relevant property only when constructions are being acquired. In any case, logically plausible though such a theory would be, there is always some point at which order (in the sense that the term has in mathematical logic) comes into play: linear and hierarchical are two different kinds of order. Conceptual Structures (CS), of the kind advocated for by Taylor et al. (2007) and Krivochen (2015b), and which are not linguistic in nature, nor are they used exclusively by language, are hierarchical in nature; and so are their narrowly linguistic 'Conceptual Semantics' counterparts (e.g., Jackendoff, 2002). However, the take we will adopt here on CS is significantly different from that in the references, as we will concentrate on the physical side of things, attempting to strengthen the bridges between physics and neuroscience.

Before introducing our own approach to the oppositions and their relevance, which will be based on the notion that there is a dynamical tension between mutually opposing tendencies at different scales, we must briefly review a different kind of approach, which aims to unify *simplicity* and *likelihood* in theoretical and empirical terms. Chater (1996) discusses principles and predictions of likelihood and simplicity measures, and argues that '*despite appearances, they are identical*' (1996: 567). As we pointed out above, within information theory, it is not clear even that simplicity is the appropriate term in the first place, since the notion of *entropy* (and the noiselessness requirement) point towards a faithfulness-based system, and it becomes important for the perceptual system to capture regularities in the data that can betray an underlying pattern (such that the system evolves towards a more ordered state by means of recognizing local redundancies). Unsurprisingly, then, Chater's first argument in favor of the unification of likelihood and simplicity measures comes precisely from the consideration of information-theoretic consequences. In fact, variants of Shannon's approach to information theory (and related notions, like *minimum description length*) provide in more explicit ways a form of unifying simplicity and likelihood. In this context, likelihood involves assumptions about *subjective probability* (such that the perceptual system chooses the most likely hypothesis pertaining to a distal stimulus given Bayesian probability measures), whereas simplicity involves assumptions about the nature of the *code* (such that minimal description length becomes relevant when evaluating possible structural descriptions; again essentially a transderivational criterion). The relevance of transderivationality becomes evident if we take a look at Chater's (1996: 569) summary of what information-theoretic measures do:

> *the information-theoretic approach does not measure the information in a particular stimulus per se, but rather measures the amount of information in that stimulus relative to the probabilities of all the other stimuli that might have been generated*

Chater sees *likelihood* and *simplicity* as two sides of the same coin, which in fact follows straightforwardly from his premises. Concretely, *the most likely hypothesis, H_i, about perceptual organization is the H_i that supports the shortest description of the data, D, concerning the stimulus.* (Chater, 1996: 571)

He proceeds, then, to 'patch' some shortcomings of the information theory approach by means of a Universal Turing Machine (1996: 572), as an example of one of the 'simplest universal languages', which can be used in combination with Kolmogorov complexity to provide a single programming language for all perceptual stimuli, assuming, of course, that a the language generated by means of a TM properly contains the language(s) generated by means of simpler computational procedures (LBAs, PDAs…), which is far from obvious empirically speaking (as we will see below). The cornerstone of this part of the reasoning is precisely the universal character of the programming language, which, when considering natural languages, is actually a shortcoming: assuming a strong uniformity thesis (Culicover and Jackendoff, 2005: 6, ff;) leaves us with a generative system that is sometimes too powerful (and therefore assigns more structure than required), and sometimes too restricted (and therefore cannot accept open, dynamic computations, -like the ones required to get inferential contents like particularized conversational implicatures, to mention an extreme case- of the kind we will discuss here). The *argument* in favor of the unification via a single, universal programming language and a measure of complexity that constitutes an alternative to those presented above as argued for by Chater is, in our opinion, both theoretically and empirically inadequate: his assumptions include –crucially- the homogeneity of the universal programming language, which cannot capture the computationally varied nature of the linguistic input –as argued for in Krivochen, 2015a, 2016d; Krivochen and Schmerling, 2016; Bravo et al. 2015, among others– or, plausibly, any other kind of perceptual input. Moreover, the criterion of *minimal description length* based on Kolmogorov complexity is too rigid to eventually incorporate dynamic and analogic insights (like those derived from interactive computation and a field-theoretical approach, which we will pursue in **Part III** and apply to locality phenomena in language in **Part IV**) Chater concludes by considering that whatever evidence can be taken to support one measure inevitably serves also to support the other. We agree with that *conclusion*, but for different reasons: the measures are presented so vaguely that they are indeed satisfied by a wide range of empirical phenomena, without the possibility of having a non-trivial decidability criterion.

Shirley (2014), along with Chater and Vitányi (2003), also proposes that simplicity and accuracy metrics actually derive similar results, since '*a type of sequence encoded in a way that favours simplicity will also be more likely to be processed via the same representation if encountered in future*' (Shirley, 2014: 3). However, it is not explained in either work *how* exactly the system recognizes 'the same representation', or how a structural pattern is stored in order to be activated upon the encounter of that 'same representation'[86]. Problems arise with both metrics, in any case. On the *simplicity* side, there is no objective measure of complexity, neither in reference to formal languages necessary to provide the relevant description, nor with respect to the perceptive / organizational mechanisms themselves. The problem with the notion of *accuracy* as presented by Helmholtz is that it requires some external factors ('novel inputs') and the development of an internal measure (a 'category', possibly along Kantian terms). It

[86] Susan Schmerling (p.c.) has suggested that a linguistic example of this problem may refer to the *phonemic principle*, dealt with in Sapir's (1921) foundational work *Language*. We should clarify the concept of 'category' we have in mind: we refer to labels like 'sortal entity', 'event', 'actant', 'patient', 'figure', 'ground' (particularly the latter two), and those very, very basic cognitive categories we use to interact with our environment. Linguistic categories are of a different kind, less primitive and more specific in their structure. As a terminological fancy, perhaps we would call the former 'primary categories' and the latter 'secondary categories'.

must be borne in mind that the unification proposals assume (with different degrees of explicitness) that there are '*natural interpretations of simplicity in terms of shortest description length and of likelihood in terms of probability theory*' (Chater, 1996: 579) which is not free of problems, as we pointed out above.

For the time being, until we have a better knowledge of the relation between the mind and the phenomenological world and the way in which representations of the latter are constructed and manipulated (for example, see Jackendoff, 2002 for a proposal in which *parallel* systems build different representations that are mapped from one to the other; also Williams, 2003, who multiplies the linguistic sub-systems and organizes some of them hierarchically), we will assume a purely *internalist* concept of accuracy, based on global semantic requirements and, from an information-theoretic point of view, *conservation* (OT's *faithfulness*), a concept we will return to many times, but, for the time being, should be interpreted in its basic sense: 'don't delete or otherwise lose information (either lexical or configurational)' (see also Lasnik & Uriagereka's, 2005: 53 *first conservation law*). We will define *computational simplicity* of a string S as *a cost-benefit ratio, where the cost is directly proportional to the search space required in a formal automaton to compute S, and the benefit is measured as interpretative effects obtained from the computation of S in context* (namely, positive cognitive effects, along the lines of Relevance Theory. See Sperber and Wilson, 1986, 1995; Wilson and Sperber, 2004; Escandell and Leonetti, 2000). While informal, these characterizations will do for now.

2.10 Unifying oppositions: Allegro ma non troppo

Let us see how we can unify the oppositions above, without making the terms of the oppositions equivalent or neutralizing one of the terms (*contra* Chater's 1996 attempt), but precisely acknowledging that they are legitimate oppositions, and that considering them as mutually opposing forces has interesting theoretical and empirical consequences for the theory. Rather than leading to a glut of 'chaos' (in the informal sense of the term, roughly, 'complete disorder'), we will see that interesting physical / mathematical dynamics emerge from ever-struggling tendencies. It is precisely on these tendencies that we base the theoretical aspects of this thesis. The kind of reasoning we pursue here has the advantage of having already been applied successfully, in the branch of physics known as *complex systems* science. Adopting a complex-systems approach brings an implementationally viable theory of the neurocognitive substratum of language quite a bit closer than proposals centered on only one side of each opposition, without considering the consequences of the interaction between the participants (actants and forces) in a system.

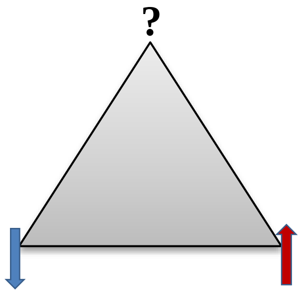
Uriagereka (2012, 2014), in the context of an analysis of Chomsky's (2005) 'Third Factor' in Language design (see Johansson, 2013 for some critical discussion of the concept and its derived consequences), claims that Third Factor considerations, which include:

> *(a) principles of data analysis that might be used in language acquisition and other domains; (b) principles of structural architecture and developmental constraints that enter into canalization, organic form, and action over a wide range, including principles of efficient computation, which would be expected to be of particular significance for computational systems such as language*. (Chomsky, 2005: 3)

are the consequence of mutually opposing tendencies at work, the result of a dynamics of conflict without resolution. Leaving aside the obscurity of the passage from Chomsky, which calls out for definitions that are never provided and which groups highly heterogeneous mechanisms with no justification (see, e.g., the list of 'third factor' mechanisms proposed in the biolinguistics literature in Johansson, 2013: 251), it seems clear that we are dealing with constraints over structures and processes that go well beyond 'language', both in its FLN and FLB aspects (Hauser et al. 2002). While Chomsky (and MGG in general) has taken these constraints and principles for granted, Uriagereka has attempted to *explain* this 'third factor' by resorting to a concept that has its origin in physics: a *dynamical frustration*. It is essential to point out that, whereas the 'third factor' is mostly a rhetorical device in current MGG biolinguistics (not unlike 'virtual conceptual necessity', see Postal, 2004: Chapter 13), Uriagereka's considerations about the dynamics of language arising from a *dynamical frustration* can be expressed without reference to a 'third factor'[87] in the Chomskyan sense. The concept of *dynamical frustration* will be crucial in our own work, but for reasons somewhat different from Uriagereka's. Consequently, it is essential to review his proposal first, so that we can point out its advantages and shortcomings, and propose our alternative.

Uriagereka (2014: 363) defines a dynamical frustration in a system-neutral way, as '*the irreconciliable tension between certain opposing tendencies that, in some conditions at least, give rise to a form of dynamical stability*'. In physics, these frustrations arise, for example, in a triangular crystal lattice under the condition that all three vertices be antialigned (that is, aligned in an opposite direction) with respect to one another: three neighboring spins cannot be pairwise antialigned, and thus a *frustration* arises (Moessner and Ramírez, 2006: 25-26; Binder, 2008: 322). Let us illustrate the situation:

80)

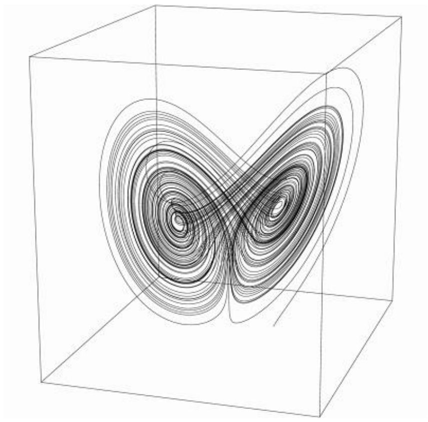


It should be clear that the antialignment condition cannot be satisfied by the system as a whole (that is, for all 3 electron pairings) for any given time, and this tension forces the system to display ever changing dynamics which can nonlinearly generate emergent properties, which are not proportional to the system's input or componentially predictable. More generally, spin glass always displays this kind of dynamical stability, in which two nearby localized magnetic momenta can be subjected to either ferromagnetic (aligned) or antiferromagnetic (antialigned) interaction with roughly equal probability (Stein and Newman, 2011: 115-116). Binder (2008) adopts a more general stance and proposes that one of the defining properties of complex systems is precisely *the presence of a dynamical frustration at its core*. He goes even further, positing that a complex system where no frustration arises will '*either settle to equilibrium or grow without bounds*' (2008: 322). Equilibrium, in this particular case, is not *balance* or *stability*, which could be, in principle, desirable: we are not referring to an optimal organization of matter or distribution of forces, but to a complete lack of activity (for instance, a pendulum,

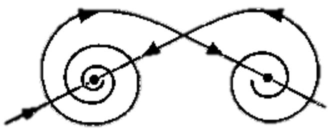
[87] The mention of the 'third factor', we think, actually weakens Uriagereka's case instead of supporting it, as Johansson (2013) correctly points out. What is more, Uriagereka's reasoning holds beyond the context of MGG, as he is dealing with fundamental properties of language (sound and meaning and their linking) which have been claimed to be in tension (as an '*antinomie*') since at least Tesnière (1959: 19-21).

which, after a while, stops moving altogether). (80) presents a dynamical frustration originated at a single scale (or ‘level of organization’). Crucially, a system can also display opposing tendencies at *different* scales: local and global tendencies may turn out to be mutually incompatible. A relevant example of scale-sensitive frustration is the well-known Lorenz attractor (81 a, b) (which describes the behavior of a chaotic system), where there are opposing clockwise and counter-clockwise tendencies in the 3-D phase space towards different sets of points ((81 b) is taken from Binder, 2008: 322):

81) a)



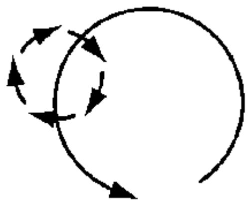
b)



In the 2-D figure (81 b) we can see opposing tendencies between ‘*stretching out and folding*’ (Op. Cit.), resembling centripetal and centrifugal forces pulling an attractor in opposite directions. As the attractor “grows” in each dimension, centrifugal forces can be informally said to prevail. This kind of frustration, exemplified by chaotic systems, is called a *geometrical frustration*.

There is another kind of frustration, which arises when global and local tendencies are naturally opposing (e.g., in fluid flow or cellular automata): a so-called *scale frustration*. In a model of local domains within a dynamical system, explaining how this kind of frustration arises and which kind of computation emerges from it becomes essential. Binder’s own example of *scale frustration* is a counter-clockwise global tendency with local clockwise cycles, as in (82):

82)



Crucially for our purposes, all dynamical frustrations are *time-sensitive*, that is, they evolve over time (apart from other factors that might enter the game, like electromagnetic interactions in the case of antialigned lattices); this property makes them suitable candidates for modelling complex systems (and, as Binder suggests, for understanding emergent behavior as a whole). In the next section, we will focus on a particular architecture for language that builds on the concept of *dynamical frustration*, from which some essential properties of linguistic structures are derived: the CLASH (acronym form of *Conditions Liberating A Simple Hiatus*) model.

2.10.1 Uriagereka’s (2012, 2014) CLASH model

Stemming from his seminal discussion of Fibonacci patterns in the X-bar template for phrase structure (Uriagereka, 1998: 192); Uriagereka (2012, 2014) develops the *dynamical frustration* perspective, and analyzes its consequences for Chomsky's (2005) notion of 'language design'. Uriagereka's inquiry builds on the idea of orthogonal although related forces tending to find a dynamical equilibrium, and proceeds to identify those forces / tendencies with fundamental components in language: in a general sense, the struggling forces would be left-to-right PF computation and bottom-up semantic computation at a level Uriagereka calls Deep Structure (as a *conceptual component*, not quite a GB-like level of representation), corresponding to a 'conceptual soup', a pre- (non?) linguistic structure which '*gets both lexicalized and computationally organized*' (Uriagereka, 2014: 367) after PF and DS collide at Spell-Out, giving rise to a *structural discontinuity* (a *hiatus*, in Uriagereka's terms) between two orthogonal sources of structuring that '*must track one another*' (Uriagereka, 2012: 267). This discontinuity breaks the system's structural balance, and the system that results, ever-struggling to regain balance, is defined by this *frustration*. The architecture Uriagereka has in mind is diagrammed as follows:

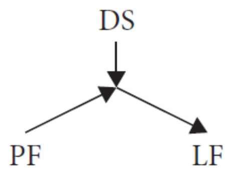


Figure 8: The CLASH model

The arrows point to the directionality of the conflict: orthogonal PF and DS tendencies that produce a dynamical frustration solved only after an interaction between these components, as LF. In this sense, Uriagereka claims that syntax '***emerges**, with all familiar (LF) syntactic details computationally determined, **only after an interaction takes place** between phonetic and conceptual networks*' (2014: 367. Highlighted in the original), that syntax being described as '*computational conditions of the 'mild-context sensitive' sort*' (2012: 276) following the TAG perspective of Joshi (1985) and much related work. The so-called 'narrow syntax' (that is, the generative engine of human language, identified with the operation Merge; see Hauser et al. 2002) is in this model an *emergent property* of the CLASH architecture. Moreover, the interaction between these orthogonal tendencies (given certain additional assumptions we will review below) also provides a 'principled' account of specific cyclic effects in linguistic computations, from syllable structure to phases (or other periodically determined domains within phrase markers).

For the model to work, in the terms in which Uriagereka presents it, there is a basic assumption that is not made explicit in any of the works cited, but is essential to understanding the dynamics of CLASH: *Structural Uniformity*. From syllables to phrases, Uriagereka assumes strict uniformity, in the sense that all results of computational procedures are exhaustively characterized by the same formal template, roughly, a *normalized* L-grammar (an L-system modified to be expressed in Chomsky-normal form, as a PSG). It is useful to take a look at the so-called 'F-game' (Uriagereka, 2012: 281; see also Uriagereka, 1998: Chapter 6) and its rules:

83) *F Game*:

(i) Starting with either a + or a −,
(ii) Go on to concatenate it to another + or a −, with one condition:

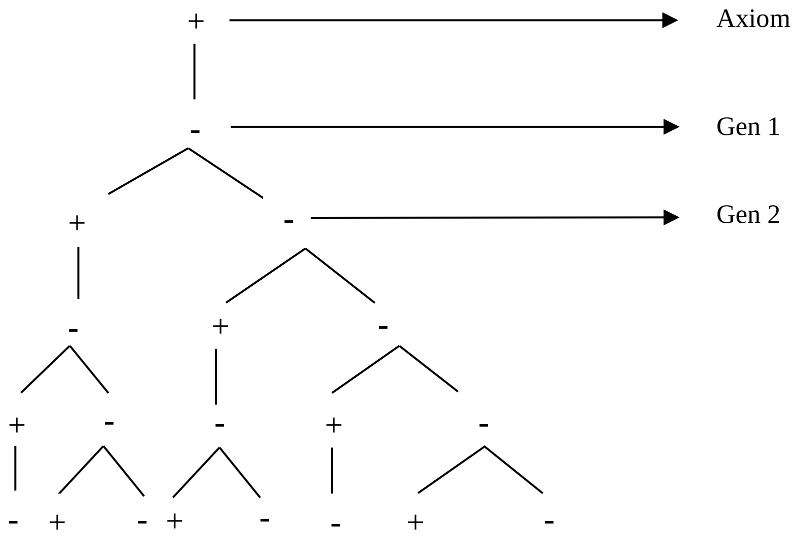
(iii) Avoid combining identical symbols, unless adjacent to a different symbol.

The alphabet is reduced to two elements, which is not problematic in and of itself, but the combination rules are: how is condition (iii) derived? Why should it be a well-formedness condition? The game is perhaps better understood if presented in the form of an L-grammar of the kind analyzed in the previous section (see also Uriagereka, 2012: 282):

84) Axiom: +
Rules:
(a) + → -
(b) - → +, -

If the tree is developed (and, again, taking into account not branch length but generation number, for the time being), the Fibonacci sequence arises by adding the number of [–] at each generation (or the number of [+] from generation 2 onwards), and we see that, at any branching level, there are no more than two 'adjacent' tokens of the same symbol:

85)



We see that rule (iii) of the F-game emerges from the L-grammar, but there is still a problem: what does the L-grammar emerge from (if anything)? In other words: the pattern is in some sense 'there', but, where does it come from? Within the CLASH model, it emerges when two orthogonal planes collide (such that '*the very Fibonacci forms* (…) *are themselves the result of dynamical frustration*', Uriagereka, 2012: 227), but it is important to bear in mind that this is not the *only* possible incarnation of a CLASH (or rather, a CLASH-compatible) model. It is possible to get the relevant patterns (including Fib) with a different underlying assumption about structure, more concretely (in our particular case), abandoning *structural uniformity* as a condition over phrase markers (see Krivochen, 2015a for discussion)? Could it be that Fib, or any other seemingly deep pattern, arises only *locally* within a mixed computational engine, and under specific environmental conditions? What would the *global* picture look like, them? Provided that not all cognitive interfaces are frustrated (as pointed out to us by Tom Roeper, p.c.), can there be other possibilities to model such interfaces (e.g., non-chaotic regimes)? In the way the issue was presented here, there is a crucial question left unaddressed: does Fib emerge because it is a real mathematical regularity, with explanatory potential at the computational and/or physiological levels, or is it an emergent property of the way this specific L-grammar has been devised? One potential problem we find, and which is too serious to be overlooked, is the possibility that the Fib argument is a 'self-fulfilling prophecy', in which we find exactly what we expect to find because the grammar has been designed to deliver certain patterns (e.g., we never find a ++ or a --- substring, see Shirley, 2014: 10-12; Patel et al. 2015 for more technical details). Moreover, phrase markers need not be trees in the technical sense (see, e.g., Jacobson, 1977; McCawley, 1982b, 1998; Ojeda, 1987 among many others), let alone L-trees. And even less so Fib-trees. So what do we do?

It is crucial for the reader to bear in mind we are *not* claiming that Fib patterns have no significance in language and / or cognition (since, as Saddy, 2009 has shown, there are measurable effects that *have* to be accounted for), we merely recognize the need to address 'fundamental' questions: it might be the case that these patterns arise because of more basic conditions (e.g., minimizing the size of the grammar while not making it trivial) and restrictions over possible computations, which we will explore in terms of mutually interfering *fields* (which might be equally a mirage, of course, but one with very different properties). The capacity to identify Fib from pseudo-Fib *need not* correlate with aspects of linguistic phrase structure (but it might), and *in and of itself*, it tells us nothing about the format of phrase markers (as the latter need to consider not only some representational stability, but also linear and nonlinear meaning composition; both of which properties are absent in L-systems). As a matter of fact, we will propose here (in **Part III**), following Saddy (2018) and expanding on Saddy & Krivochen (2016a) that linguistic computations oscillate back and forth between the L-space and the Chomsky-normal space (which corresponds to an oscillation between ultrametric and metric spaces respectively). Our perspective does attempts to critically analyze the foundations of previous work in order to identify weak spots and dig into deeper properties of the underlying physical substratum of the assumed system in order to either support or eliminate certain basic assumptions.

Before proceeding to propose our alternative to the CLASH approach to *dynamical frustrations* in language and cognition, we must say that the CLASH model is one of the very few coherently devised attempt within linguistics to incorporate insights from both mathematics and physics (including dynamical system theory) into the picture, not just analyzing 'narrow' syntactic phenomena (e.g., sub-extraction and opacity, see Uriagereka, 2012) but also formal and neurocognitive consequences of the model for the architecture of minds and brains (Uriagereka, 2014: §§ 8-9). This model in turn inherits the concern for mathematical patterns in language (Fibonacci sequences playing a central role, particularly if the considerations made above about Fibonacci being an emergent property of L-grammars under the simplest environmental conditions are on the right track) from previous works like Uriagereka (1998), Saddy (2009); Saddy and Uriagereka (2004), the latter of which includes the first discussion we know of about complexity in language arising from an interaction between '*the grammar of the language system itself and the procedural complexity resulting from marshalling processing resources in order to produce or interpret utterances that correspond to the grammar*' (Saddy and Uriagereka, 2004: 383), and also Medeiros (2008). These works (which have unfortunately been overlooked by MGG) analyze L-grammars in connection to the Fibonacci sequence and other mathematical regularities (like the Golden Mean, the limit of successive Fib terms). Saddy and Uriagereka call attention to the existence of common principles (Conservation, Locality, and Symmetry Breaking) between language and complex dynamic systems (a proposal we have expanded on in Krivochen, 2013c). This is crucially different from the claims of strong isomorphism between mathematical and linguistic structures that sometimes appear in the literature. Moreover, these works we have referred to usually take into consideration the implementational level of analysis, proposing consequences of the theoretical claims they make for minds and brains (e.g., the discussion of predictive strategies and Garden Path Sentences in Saddy and Uriagereka, 2004: 394-398; the observation of Fibonacci ratios in the EEG spectrum and the possibility of devising neurological tests for the presence of the relevant computational process in Uriagereka, 2014: 384-385), being thus truly *rarae aves* in the linguistic literature. We do want to make a point of these efforts, since, in our opinion, they constitute highlights in the studies of language which have largely been disregarded by the mainstream. The fact that we will present a somewhat different model, and criticize some of the claims made by these authors does *not* call into question their merit in contributing to the foundation of a real physics of language.

2.10.2 An interactive twist to CLASH

Now, how can the framework presented above—i.e., a CLASH-like cognitive architecture based on a *tension* rather than on *balance*—help us with the aforementioned oppositions? Let us repeat them here for ease of reading:

86) a. Locality vs. Globality
b. Morpho-phonological vs. semantic requirements
c. Simplicity metrics vs. accuracy metrics vs. likelihood
d. Linearity vs. Hierarchy

Local and global tendencies are precisely where *scale frustrations* arise. A crucial point in our argumentation will be that the *scale* we consider for the study of a certain phenomenon will determine the kinds of effects we can expect to see, and this will prove essential when we move to the experimental side of the present work. There is, then, no actual *substantive* content to (86 a), insofar as it merely states that there are differences in scale, and that local processes can be at odds with global processes; but no claim is made with respect to the nature of those processes or the elements involved in them. Here is where (86 b) becomes relevant. Following Uriagereka (2012), and Tesniere (1959) –among others-, we consider that morpho-phonological and semantic requirements are orthogonal to each other: this tension involves also (86 d), insofar as semantic structures are both global and of a higher order than morpho-phonological strings, which are local and of a lower order. In Tesniere's terms:

> *1. — The possibility of a term in the structural order having, beyond its unique higher connection, two or three lower connections* […] *collides, in its place in a sentence, with the impossibility of a word in the spoken string being immediately in a sequence with more than two adjacent words* […] *In other words,* ***every structural node is susceptible to the creation of bifurcations, trifurcations, etc…, that are incompatible with linear order****.*
>
> *3. —* ***There is thus a tension between the structural order, which has several dimensions*** […], ***and the linear order, which has one dimension****. This tension is the squaring the circle of language. Its resolution is the sine qua non condition of speech.*
>
> *4. — The tension between the structural order and the linear order can only be resolved by sacrificing at least one linear sequence at the point of placement in the sentence.*
>
> *5. —* […], *we can now clarify that to speak a language is to know what the structural connections are that may be available for sacrifice in transforming the structural order*

into a linear order, and inversely that to understand a language is to know what the structural connections are that are not expressed by sequences that may be available for reestablishment when the linear order is transformed to a structural order.

(Tesnière, 1959: 21. Trans. Susan Schmerling)

What is the role of 'syntax' in this tension? It cannot *emerge* from the frustrated interaction between sound and meaning, since we can find syntax in other cognitive domains (vision, arithmetic, audition…). Could it be another plane in the linguistic picture together with sound and meaning; what is more, a plane that gets dynamically *limited* by the sound and meaning planes, giving rise to a tension-based dynamics? In Krivochen (2013c); Kosta & Krivochen (2014b) we explored this idea, which we illustrated using a piece by the Dutch graphic artist M. C. Escher:

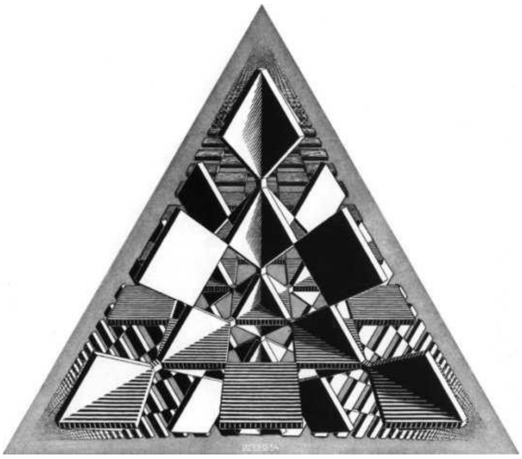


Figure 9: M. C. Escher (1954) "Three Intersecting Planes". Woodcut in three colors. 325 x 375 cm. Taken from J. L. Locher (ed.) (1988) *The World of M. C. Escher*. Amsterdam: Taschen. p. 210.

Given this architecture, and assuming just three *distinct* planes, Phon(ology) (more generally, a sound system, phonology / phonetics being restricted to the study of natural language), Sem(antics), and Syn(tax), there are four possible intersections, which we defined as follows (Kosta & Krivochen, 2014b: 36-37):

87) Sem ∩ Phon → interjections and vocatives (see Chomsky, 2008: 139 on the lack of combinatorial power of interjections[88], which would be explained by the absence of the Syn component in the definition, but their belonging to natural languages all the same), animal calls (e.g., vervet monkeys, see Demers, 1988)

Sem ∩ Syn → conceptual structures (Fodor, 1970; Jackendoff, 2002; Taylor et al., 2007; Uriagereka, 2011, 2012; Krivochen, 2015b)

Phon ∩ Syn → music capacity (Jackendoff and Lerdahl, 2006), structured bird sing (Uriagereka, 2012: Chapter 6)

[88] I.e., they cannot be combined with other lexical items to form constituents or phrases of any kind.

Phon ∩ Sem ∩ Syn → clausal grammar (dynamical dependencies between objects, 'mixed computation')

That is, back then (and in line with MGG) we assumed that there was an autonomous structure-building algorithm (an 'autonomous syntax' as in Chomsky, 1957 and much subsequent work), but added the caveat that cognitive structures thereby derived are neither inherently linguistic, nor need they be; in the proposal explored in the aforementioned works, natural language enters the scene *only* when:

a. There is *no* one-to-one relation between entities of the phenomenological world and expressions belonging to the language

b. There are sound-meaning relations (although not *correspondences*), such that phonological exponents materialize structure

c. Computational complexity can go up and down the Chomsky Hierarchy within cyclic domains

Point (a) is crucial, insofar as natural languages are never triggered by a stimulus-response mechanism: a human can see a snake and shout "there's a snake!", or anything else he wants, or just do nothing. In contrast, the vervet monkey has no choice but to respond to the visual stimulus "snake" with a particular call (Demers, 1988). (b) is a claim for representational economy: do not materialize elements that cannot receive an interpretation (e.g., *John bought the book the car, where [the book] and [the car] are two different constituents, vs. John bought the book and the car / John bought the book / John bought the car), do not have superfluous phonological exponents which do not correspond to any syntactic node, in turn required by the semantic component: this of course implies as well 'eliminate superfluous nodes', like ArgP (or any other purely theory-internal narrowly syntactic mechanism), as they are not required by the semantic component[89]. This kind of constraint can be found in other syntactic frameworks, notably, LFG's *economy of expression* (Bresnan, 2001: 96) and Postal's (1974: xiv) rejection of arbitrary syntactic diacritics and other unconstrained symbols in the meta-theory. (c), on the other hand, refers to the mixed computational nature of phrase markers, which we have thoroughly analyzed in Krivochen (2015a, 2016b, c, d); Saddy & Krivochen (2017); Bravo et al. (2015), and which we think is a deep property of language, rooted in more basic cognitive operations involving assigning an input the simplest grammar that satisfies a requirement of *strong adequacy*[90]:

[89] It must be noted that different dialects might differ with respect to what counts as 'meaningful': Susan Schmerling (p.c.) has noted that Boston English is notorious for having [And so don't I], where other dialects would say [And so do I]—the Boston example is *not* equivalent to [And I don't either]. Those examples are not counterexamples, for each element contributes to semantic interpretation. A distinction must be made between what is interpretable for some speakers of a certain variety (i.e., situationally-bound interpretability, which is external to the formal system) and what is not interpretable under any condition (e.g., *…and so don't will I), including word salad. We will return to the issue of 'ungrammaticality' and 'unacceptability' in **Part IV**.

[90] In this context, the question has been raised to us whether *strong adequacy* is equivalent to *strong generative capacity* (Chomsky, 1965: 60). The difference is crucial: the strong generative capacity of a grammar is the set of structural descriptions it generates; the definition given by Chomsky says nothing about that set being internally heterogeneous. The grammar that generates structural descriptions in Mainstream Generative Grammar is computationally uniform, and thus the set of structural descriptions is computationally uniform as well (falling within Type 1 languages). In contrast, Joshi's requirement for

> *A grammar G is weakly adequate for a string language L if L(G) = L. G is* ***strongly adequate*** *for L if L(G) = L and for each w in L, G assigns an* ***'appropriate' structural description*** *to w* (Joshi, 1985: 208)

The requirement of *strong adequacy* for natural language strings, we argue, cannot be satisfied of the generative engine is locked at a single level in the Chomsky Hierarchy: interactive computation, and the possibility of oscillating up and down the Hierarchy become essential. It is important to note that we do not claim that the study of dynamical systems exhaust everything there is to know about language: mental representations of culture (i.e., an internalist approach to social relations, common knowledge, and so on), which have been argued to influence aspects of grammar (see Everett, 2005 for discussion), to give but one example, could also be included in this framework, the only condition being that there be a fully explicit theory of the aspect in question, as well as the interactions with other systems and the emergent properties of such interactions.

Having this architecture in mind, we defined natural language *as*[91] the intersection of a module-neutral computational system (which adapts dynamically to the input, assigning to it the simplest possible grammar that captures semantic dependencies between terms in local chunks of structure) and the systems of sound and meaning; there is *no theory of language without a theory of the 'interfaces'*. This much is not too far from MGG. But we think that the idea can be simplified and improved upon. A relevant question at this point, which implies a critical revision of traditional MGG positions (e.g., Chomsky, 1957, 1965), is whether there is an independent Syn plane at all (that is, an independent, autonomous computational component) as assumed in MGG and in some of our previous works (Krivochen, 2011, 2014c; Kosta & Krivochen, 2014b); or whether 'syntax' is actually an operation affecting the topological layout of the other planes (under the simplest traditional –Aristotelian, Saussurean, Tesnièrian- assumptions Sem and Phon). In this latter view, phrase markers are manifolds which intersect when the space they exist in is modified and its metric, disrupted (see Saddy, 2018 for a development of this view). The scenario this latter possibility presents us with refines the architecture in **Figure 10**, and of course represents a significant departure from the traditional MGG claims about the 'independence' or 'autonomy' of syntax (Chomsky, 1957: 13, ff.) in favour of a more 'Kantian' perspective in which operations and substance are one and the same thing. The model we will argue for here is very much in the line of Hjelmslev (1961) who considered that *expression* and *content* were universal continua which were *partitioned* differently in each language; this yields a non-generative system which is internally consistent, and compatible with our topological approach. Schmerling's (in press) *neo-Sapirian Grammar*, very much influenced by both Sapir and Montague, is another example of such a system. The

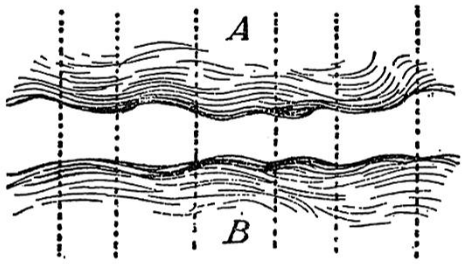
---

strong adequacy for structural descriptions can incorporate aspects of mixed computation if the grammar is made sensitive to semantic dependencies between syntactic objects (which, in Chomsky's architecture for a linguistic theory, cannot be formulated given the fact that the syntactic component is conceived of as an autonomous component). Chomsky's *strong generative capacity* makes sure there is a set of structural descriptions, Joshi's *strong adequacy* makes sure that set is (under present assumptions) minimally adequate (not assigning any more structure than strictly necessary), but not procrustean. See Krivochen (2015a; 2016); Krivochen & Saddy (2018) for discussion.

[91] *As* as opposed to *in* (the latter, arguably the position of Stroik and Putnam, 2013). What we say is that natural language *exhausts* the intersection (i.e., there is no other system defined as that intersection), and that this constitutes what is specific about NLs in their internalist, mental aspect, rather than 'recursion' (*contra* Hauser, Chomsky and Fitch, 2002; among *many* others).

idea of having ‘syntax’ as a partitioning function over sound and meaning continua can indeed be traced back to Saussure:

> *Psychologically* ***our thought—apart from its expression in words —is only a shapeless and indistinct mass****.* […] *Against the floating realm of thought, would sounds by themselves yield predelimited entities? No more so than ideas.* ***Phonic substance is neither more fixed nor more rigid than thought****; it is not a mold into which thought must of necessity fit but a plastic substance divided in turn into distinct parts to furnish the signifiers needed by thought. The linguistic fact can therefore be pictured in its totality—i.e. language—as a series of contiguous subdivisions marked off on both the indefinite plane of jumbled ideas (A) and the equally vague plane of sounds (B).* (Saussure, 1916: 111-112)



We will come back to this stronger departure from traditional US-structuralist and generative architectures (both of which instantiate Item-and-Arrangement models, in the sense of Hockett, 1954 and particularly Schmerling, 1983) all throughout the present thesis, in topological and physical terms (to be developed in **Parts III** and **IV**). What we want to emphasize in the present section is a plea for *interactive* models, and against the isolation of any component in a model of the grammar: any linguistic phenomenon is to be described and explained as a function of intersection and interplay. Purely syntactic accounts based on features or alleged UG-principles are therefore rejected because of its inherently stipulative (UG-*ex machina*) character (see also Postal, 1974: xiv; 2004: § 13 for a more detailed discussion).

There are good conceptual reasons to adopt a framework like the one defended here: take cyclicity in syntactic computations as an example (with concomitant extraction filters, *à la* Ross, 1967)[92]. If something along the lines of Uriagereka (2002a) is at least partially correct in assuming that the linearization operation Spell-Out targets finite-state compatible sub-derivations (monotonically assembled *command units*), the Multiple Spell-Out (MSO) model *emerges* (to use the term that Uriagereka, 2012: 153 uses himself) naturally from the computationally mixed character of phrase markers, which include monotonic and non-monotonic structures (in Uriagereka’s account) or mixed computational dependencies that go well beyond the limitations of endocentric, rooted, labelled, binary-branching graphs and, in their local variety, define cycles as an emergent property (in ours). A MSO account is a partial answer to the ‘Hierarchy vs. Linearity’ problem: global semantic tendencies towards

[92] This is a central topic in our argumentation, and we will return to it in more technical (and, hopefully, explanatory) terms in **Part IV** below.

conservation[93] *clash* -pun intended- with local materialization requirements targeting minimal-maximal finite-state currents. Global semantic effects include the following (partially following Uriagereka, 2011: 2, ff.):

- Quantifier scope interactions (Sloan & Uriagereka, 1988)
- Ellipsis under parallelism (Ross, 1969a; Merchant, 2001, 2008; see also the criticism in Culicover and Jackendoff, 2005: 239, ff.)
- Long distance anaphora (see also Krivochen, 2012b for discussion on Latin LDA)
- Explicit / implicit meaning tensions (*explicatures / implicatures* respectively; Sperber & Wilson, 1995; Escandell & Leonetti, 2004)
- Lexical semantics and Aktionsart (Dowty, 1979; McCawley, 1968)
- Thematic roles (Gruber, 1965; Hale and Keyser, 2002; Krivochen and Kosta, 2013)
- Dependencies across constituents (the notion of 'chain' in Uriagereka, 2011: 11; also Martin and Uriagereka, 2014: 177-178), including the interpretation of multiple gap constructions, etc. (see Krivochen, 2015b for a different perspective)
- Interactions among the above

It is crucial to notice that a *phase* based framework could not accommodate these phenomena naturally, since the identity of *phases* has nothing to do with computational properties of phrase structure (unlike *command units*, which derive from the formal properties of the representation being built; or the kinds of dependencies identified in auxiliary chains in Spanish by Bravo et al., 2015; García Fernández et al., 2017) or semantic considerations (once Chomsky, 2007, 2008 explicitly rejected the possibility that propositionality and theta-domains signal phase boundaries, which would not have made much of a difference anyway as there is no unambiguous definition of 'propositionality' in Minimalism, for instance; see also Chomsky's personal communications with Angel Gallego in Gallego, 2010: 54-55), but rather with intra-theoretical requirements for feature-checking / valuation (making the notion of *phase* parasitic on the notion of Agree, since phase heads are the true *probes* for valuation purposes, see Gallego, 2010: 51, ff. for more details). In MGG, all roads lead to UG, which is an undesirable scenario since there is no independent evidence for UG (as a matter of fact, there is not even a fully explicit, unambiguous definition of UG or a detailed, falsifiable characterization of what exactly is there, after more than half a century of UG-related research). The reasoning from a *frustrated* perspective goes pretty much like this: we have a certain structure following from particular computational considerations which we have to express by means of different computational dependencies. For the sake of concreteness, call them *conceptual* and *morpho-phonological* structures respectively. While a single-component model would attempt to subsume one to the other, or just find an extraneous 'explanation' (e.g., transfer the complements of certain stipulatively chosen 'phase heads') a *frustrated* perspective tries to analyse the interaction between the relevant components, which in the case of semantic-morphophonological interactions in natural language involve taking local chunks of structure

[93] See Lasnik, Uriagereka and Boeckx's (2005: 53) *First Conservation Law*; Krivochen's (2011: 53) *Conservation Principle*[93]; see also McCarthy and Prince (1995) for a take on 'faithfulness constraints' in Optimality Theory, which apply conservation at local levels.

from the global derivation that follow a particular computational pattern (say, finite-state), and letting the relevant system operate with them. Cyclicity effects (which are *not* to be equated to *periodic impenetrability*, as emergent properties are not necessarily periodic if the system is non-linear: ouputs are not linearly related to inputs) are in this model a by-product of more basic computational requirements: everything you have to do, do it within a finite-state current, otherwise, you will have to establish a dependency between objects that are not in the derivational workspace at the same time.

We beg the reader to bear in mind the concept of *phase space*, which will be very important for our proposal: *a phase space of a dynamical system is a space in which all possible states of a system are represented, with each possible state of the system corresponding to one unique point in the phase space.* Phase spaces are *n*-dimensional, with each axis corresponding to a variable of the system (what some call the 'degree of freedom' of said system, each parameter being able to adopt different values), and can be used to model *fields* (in the physical sense; Zachos and Curtright, 1999), a concept we already anticipated will play a crucial role in both the theoretical and empirical aspects of the present work.

A relevant question now is: 'how to characterize output representations given such a system?' From a *dynamical frustration* viewpoint, outputs are always the result of the optimal resolution of the relevant tension at some derivational point D_T, given a set of conditions (technically called 'constraints' in the Optimality Theory literature, see Prince and Smolensky, 2004) $C = \{c_1, c_2, \ldots, c_n\}$ (and for some subject *S* in whose mind the system of constraints is represented). The 'frustrated' perspective is not incompatible with bidirectional approaches within Optimality Theory, in which the focus is on *limiting* the power of the generative engine (referred to as 'GEN' in OT literature), rather than on enriching the intra-syntactic apparatus by means of features and operations. In this respect, the following seems to be a good first step towards a concept of *optimality* within a frustrated perspective (from Jäger, 2002: 435; building on Blutner, 2000):

A form-meaning pair (f, m) is x-optimal iff
1. <f, m> ∈ GEN,
2. There is no x-optimal (f', m) such that (f', m) < (f, m).
3. There is no x-optimal (f, m') such that (f, m') < (f, m).

(Where *f* = form, *m* = meaning, and < = 'more harmonic than', in the sense of Smolensky and Legendre, 2006). It is not far-fetched, in our opinion, to claim that *x*-optimality is a way of approaching the resolution of a tension between systems, even though the nature of those systems (their characterization in terms of computational dynamics, for instance) is not discussed in the same terms we use (but see Smolensky and Legendre, 2006 for a very thorough investigation of the properties of OT-systems in formal and cognitive terms). It must be noted, however, that the resolution of the tension involves *f* and *m* conditions applying at different scales (thus giving rise to a *scale frustration*): *form* constraints operate at a *local* level, whereas *meaning* constraints operate at a *global* level; *conservation* targets CS-LF whereas PF adds information to locally determined chunks of terminal nodes. The latter, moreover, are closer to what OT understands as *faithfulness constraints*, which demand that input and output be identical in some relevant respect[94].

[94] This requirement makes faithfulness constraints compatible with a strict formulation of Minimalist Grammars under the Inclusiveness Condition IC (Chomsky, 1995: 228), but stronger: while the strongest

Going back to the tensions summarized in (86), the only opposition we have left is Simplicity vs. Accuracy.

It follows from the argument we have been building that it is impossible to have a nice, uniform PF-LF (or, simply, sound-meaning) interaction that simultaneously satisfies the output conditions of both systems since these components operate with *different kinds of computations*: finite-state for PF, and more complex for LF, the representation of which requires a higher-dimensional phase space (see also Tesnière, 1959). Complete accuracy is out of the question, even if only for the fact that structure needs to be 'Markovized' (without considering, for the time being, the possibility that this Markovization, which implies dimensional flattening in a sense we will see below, is dynamic and can apply multiple times during a derivation –e.g, via Multiple Spell-Out-) to be externalized, one sound at a time. Computationally, Spell-Out implies a simplification, strictly following the hierarchy of formal languages: if materialization targets finite-state currents (Uriagreka, 2002a, 2012), then each derivational current must be recognizable by a FSA regardless whatever kind of formal system(s) needs to be called upon to compute (more) global dependencies; that is, relations between distinct FS local derivational currents (Uriagereka's *command units*).

If the semantically driven derivation of natural language sentences is indeed interaction-based instead of function-based -which is one of the main points in the present work- then there is a stronger argument in favor of *simplification* being computationally unavoidable. The question is, is there anything like a 'simplicity metric' (i.e., a measure for simplicity) in a frustrated system? Our answer is 'no', at least not in the sense 'simplicity' is interpreted in perception studies (cf. Chater, 1996): as we have said, simplification is not a criterion for comparing candidates of any sort, but an unavoidable dynamic process at the very core of the CLASH model, triggered by the need to externalize structure. The reason is that there is nothing to compare, nor is there a variable measure: linearization of hierarchically structured phrase markers is (under the simplest yet strongest assumptions) either Markovization or nothing. If this is true, it is a deep, non-trivial property of the system: externalization of phrase markers involves as a fundamental step the ordering of terminals as a string (which does not mean that such a string does not have acoustic properties to be accounted for, like pitch movement, frequency, tone, among others). Let us go through it the other way around: sounds reach our ears sequentially, and only in that way can we make sense out of them (an observation made famous by Saussure, but traceable back to Aristotle). The right sounds arranged in a non-sequential manner simply make no sense. Take, for instance, [k^hæt˺]. Only if presented sequentially can we activate the lexical entry associated with the string of sounds that reach our ears. If we recorded each allophone [k^h], [æ], [t˺] separately, and presented them to a subject simultaneously, as a kind of 'linguistic chord', the result would just be non-interpretable noise. It seems sequentiality is a legibility condition on the PF side of the linguistic coin, *if PF is taken to be a linearized phrase marker*. How can we get to Type 3 grammars from there? Well, computationally, the simplest grammar that can give us pure sequences is a finite-state grammar, capturing all relevant properties of the sound sequence. If a Type 3 grammar can provide structural descriptions of sound strings *qua strings*, without assigning excessive structure to the strings (and also, equally importantly, without losing information), then why go

possible version of the IC bans any *insertion* during the derivation, the strongest possible version of faithfulness bans both *inclusion* and *deletion*. It should be obvious that, in an interactive model like the one we argue in favour of here, inclusion of new sources of information is necessary (see also Wegner, 1997). We do, however, want to have the least entropic process possible.

further up? In the absence of an independent requirement, for either theoretical or empirical reasons, we will stick to the picture in which sound strings are finite-state in nature.

Of course, saying that *sound strings* (*qua strings* and nothing more, this is paramount) are finite-state does *not* amount to saying that the phonotactic space is finite-state as well. Sounds are arranged on strings for externalization purposes, which give us finite-state properties, but the set of solutions for formally possible combinations of distinctive features restricted by articulatory limitations is most likely not finite-state. In fact, we think it is not even a metric Euclidean space to begin with, but an ultrametric space instead (an idea we have introduced above, and to which we will return in depth below), the result of a *dynamical frustration*, which gets topologically transformed as structure builds up, creating associations between objects and making their neighbourhoods intersect. Considerations of ultrametricity might be justified in terms of accessibility: for a language L, all sounds that the human vocal apparatus can produce are equally accessible when assembling the phonological inventory, even if not all are equally used (and some not used at all). Statistical considerations about occurrence of certain sounds have nothing to do with their accessibility in a formal system; if anything, occurrence is tangential to the properties of the system. Be that as it may, we do make a point of distinguishing phonological linearization (i.e., transforming a hierarchical structure into a string) from phonotactics, and stress that finite-state computation considerations apply, for sure, to the former, but the topology of the latter is a more complicated matter.

Returning to our considerations concerning evaluation criteria, what happens with *accuracy*? Is it a reasonable measure to take into account? Do we have something to compare and something else to compare it to? The answer to this, both in CLASH and in our modified version, is 'yes'. Recall the CLASH architecture, where the relevant portion has been highlighted:

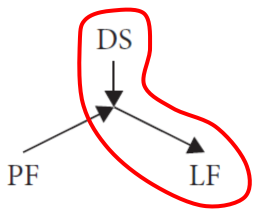


Figure 8': The CLASH architecture.

Let us concentrate on the enclosed part: the directionality is *from DS to LF*, uniformly. Uriagereka himself acknowledges that the arrows are not merely decorative: '*arrows are meant in all seriousness: as directional*' (2012: 266), a claim we share: those are trajectories that the dynamics of the system follow, and which drive the CLASH of representations with different computational natures. Taking into account an unavoidable amount of noise, the system should *conserve* information from DS all the way down to LF following the *Conservation Principle*. Uriagereka's DS is what here and in past works we have identified as a non-linguistic Conceptual Structure, whereas LF is more generally understood as encompassing both explicatures and implicatures (in the sense of Sperber & Wilson, 1995; Wilson & Sperber, 2003). PF is also more generally understood than in MGG: in this work it will refer to the dynamic Markovization of local structural domains (following the seminal idea in Uriagereka, 2002b). Let us take a look at a CLASH-adapted (and generative-semantics-inspired) version of the Multiple Spell-Out architecture model, assumed throughout Krivochen (2015a, b):

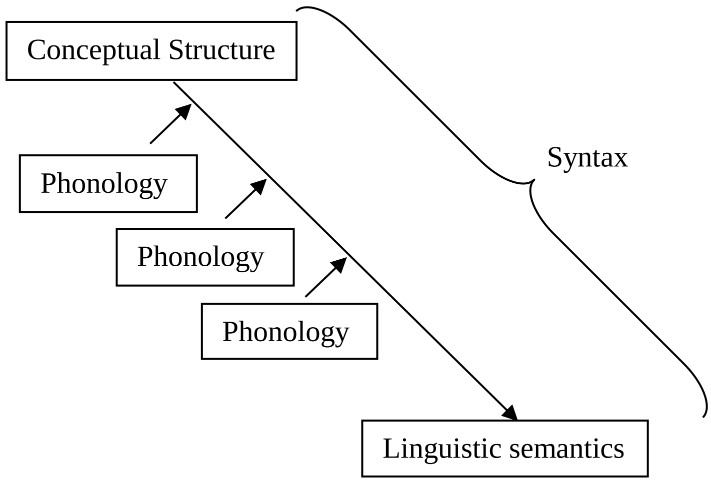


Figure 10: Modified CLASH architecture.

In this architecture, unquestionably inspired by Uriagereka's (making the MSO aspect a bit more evident) the directionality is even more obvious. Now, what exactly are we conserving? It depends on what we think the pre-linguistic unstructured 'conceptual soup' is, the information it contains, and the format in which that information is structured and linguistically instantiated, including the non-trivial issue of dimensionality. It must be emphasized that, since we consider 'syntax' to be pervasive in all cognitive processes, as essentially *relational*, we have it *all the way through*, not emerging from the frustration. This is the first difference from the 'traditional' CLASH: our conception of 'syntax' (and 'phrase structure', among other related notions) is not only broader, but also more dynamic (as we reject the imposition of *structural uniformity*, allowing for multiple kinds of computational dependencies within a phrase marker, aiming at *strong adequacy* in the sense of Joshi, 1985). Therefore, we are not constrained by the *binary-branching* axiom in structure building, and we can have *n* branches if the best (the one with the shortest description, and computationally simpler in terms of the Chomsky Hierarchy) structure to capture the semantic properties of a particular syntactic object requires so. However, as it stands, this idea is still quite vague. Consider the following sentence:

88) Who knows who believes Bob?

As can be seen in **Figure 10** above, a Conceptual Structure (the result of syntactically manipulating what Uriagereka calls the 'conceptual soup', and Saddy, 2018, following the gastronomic metaphor, refers to as a 'conceptual sabayon') is the starting point of a linguistic syntactic derivation, although CS *need not* be linguistically instantiated: a CS can be externalized as an action, or not be externalized at all (the generative semanticist reading this can approach CS in terms of 'deeper structures' as in Lakoff, 1965; McCawley, 1968, bearing in mind that generative semantics was a theory of language, not of cognition, and thus their structures were bounded by conditions on lexicalization). Therefore, a CS is *not linguistic in nature*, but *pre*-linguistic, more basic and primitive (but with a higher order of dimensionality), yet displays syntactic structure as there is no *generation / interpretation* separation in this framework (apart from some terminological shortcuts): very much in the line of Stroik and Putnam (2013), the so-called 'computational component' in MGG exists *within* the 'interface systems', which means that there is very little sense in talking about 'interfaces' at all, since there is no autonomous syntax that builds structure *outside* these systems and then handles such structures to those systems. To the very best of our knowledge, this is not a common feature in linguistically oriented CS accounts (e.g., Jackendoff, 1990, 2002; Johnson, 2004; Gärdenfors, 1997; Evans, 2009: 175, ff.; Lohndal, 2014: Chapter 5), but it is present, in some form, in Generative Semantics (e.g., Lakoff's 1970 'natural logic'; McCawley, 1971). Nor, unfortunately, is an explicit formulation of the specific structure and content of CS: neurocognitive accounts we are familiar with (e.g., Moss et al., 2009; Taylor et al., 2007) do not provide examples, and linguistically oriented proposals either make use of Jackendoff's 'tier' formalism or other semantic tools (e.g., lambda calculus, predicate calculus) which often do not capture the specific nature of CS (specific yet underspecified, since CS is to be used by more than one faculty). Let us assume a first approximation (for expository purposes) in terms of a Jackendovian logical form as follows, using a minimal number of semantic primitives:

89) [STATE know ([OBJECT who], [EVENT believe ([OBJECT who], [OBJECT Bob])])]

However, (89) is clearly not satisfactory as a CS for (88), insofar as there is no way of distinguishing the two occurrences of [who] other than by adding subscripts or some other form of diacritic element. Moreover, the structure does not include the non-assertive nature of the construal, which, if present in the syntax (e.g., by means of a complementizer C) has to be there in the conceptual structure level as well (so as not to violate the *conservation* principle, which is fundamental in any physical system[95]). Accounts like Acquaviva & Panagiotidis's (2012), or Johnson's (2004), among others, suffer from the same shortcomings, even if they differ from one another in important respects. This means that modality (at least, in terms of realis / irrealis, following Palmer, 2001) is to be present at CS, which intuitively makes sense. In Krivochen (2015d) we presented a review of Binding Theory in which entities (either sortal or eventive) are *type* variables, whose materialization depends on the construal they appear in, thus making Binding Theory an emergent property of syntactic construal and semantic interpretation; the overall perspective is not too far from Lees & Klima's (1963) transformational approach to pronominalization, in turn heavily based on Harris (e.g., 1957: 302, ff.). We used Δ for sortal entities, and Γ for eventive entities. Let us implement the proposal in this case, taking into account that, if who_1 and who_2 are interpreted as distinct, they are already distinct at the level of CS, by ConsP:

90) $\exists(\Gamma)$ & $\exists(\Gamma')$ | Γ = know(Δ, Γ') & Γ' = believe(Δ', Δ'')

It is to be borne in mind that [know] and [believe] stand for their materialized forms; at CS, all we would have would be variables to be assigned phonological signatures when (and if) they are Spelled Out and the arguments they take: the references we have mentioned before use capital letters for concepts (as opposed to linguistic elements, as in BELIEVE, see McCawley, 1968; Dowty, 1979; Acquaviva & Panagiotidis, 2012; Jackendoff, 2002: 6, 143, among others), this, we think, does not affect the argument at this point. In any case, it is important to bear in mind that these structures are semantic, which determines the wider contexts they can appear in, and the feasibility of their linguistic instantiation. However, (90) is still incomplete, as we are missing a crucial element, which is the irrealis modality, conveyed by the interrogative C head in orthodox generative grammar (from Chomsky, 1986a on incorporated into the X-bar template as a Complementizer Phrase CP). This element is to have scope over (90) as a whole, which is

[95] In terms that will be particularly significant below, when our field model is explicit, Feynman (2010: 14-6) claims that 'there are no nonconservative forces'. The apparent fact that some forces (like friction) appear to be nonconservative is an artefact of a Newtonian perspective rather than a more modern, quantum perspective, and of talking about forces rather than energy.

in turn a proposition (i.e., a set of predicate-argument relations). This is not unheard of in generative grammar; Fillmore (1968: 24) proposes the following rule:

Sentence → Modality + Proposition

And goes on to say '*the P*(roposition) *constituent is expanded as a verb and one or more case categories*', that is, predicate-argument relations. Using Searle's (1969: Chapter 2) notation for expository purposes, we will propose (88):

91) a. $p = \exists(\Gamma)\ \&\ \exists(\Gamma') \mid \Gamma = \text{know}(\Delta, \Gamma')\ \&\ \Gamma' = \text{believe}(\Delta', \Delta'')$
b. ?(p)

p is of course a shorthand for the whole proposition, which is not linguistic (it does not belong to a particular natural language, but to a metalanguage). Much discussion and interdisciplinary work pending, we will keep (91) as our preliminary formalization of CS, much research pending. Later on we will delve into the topology of conceptual structures (the 'conceptual soup / sabayon') in more detail (see also Saddy & Krivochen, 2016a; Saddy, 2018 for discussion). We will also attempt to explain how they get dynamically chunked, dimensionally flattened, and re-used in the derivation of globally cumulative structure.

Coming back to *conservation* issues, this CS is what we claim is to be conserved throughout the CS-LF path. The linguistic instantiation of CS can change the *format* in which information is presented, but it cannot delete information, nor can it tamper with the content. However, we cannot rule out *addition* of information, coming at least from two sources:

a) Non-linguistic cognitive systems (in tune with interactive computation: sensory information coming from the immediate situational context, interferences from the LTM, etc.)

b) Redundancy countermeasures given the system's entropy (including inflectional morphology, resumptive pronouns, etc.)

That is, we acknowledge the fact that, if the *language-as-a-physical system* thesis is taken seriously then derivations are unavoidably entropic. The role of whichever processes we assume take place *between* CS and LF is to *minimize* entropy (Uriagereka, 2011), in other words, operations must be justified in *faithfulness* terms (in the sense the word has in Optimality Theory). This is the global perspective. Locally, on the other hand (i.e., at each derivational step), things have to be looked at from a different perspective, as we are dealing with a different scale. If structure building is incremental (which does *not* amount to saying that structure building is *linearly and monotonically* incremental), then we have to come up with an extremely local condition to minimize entropy at each derivational step. In past works (Krivochen, 2012, 2014; Krivochen and Kosta, 2013) we have referred to such a cumulative condition as Dynamic Full Interpretation (DFI):

Any derivational step is justified only insofar as it increases the information and/or it generates an interpretable object for an Interface Level IL.

A derivation as a whole is thus a dynamical process, in which mutually orthogonal local and global (morpho-phonological and semantic) tendencies struggle, giving rise to cycles of decreasing entropy, generating an overall sinusoidal entropy pattern, with each period corresponding to a derivational cycle (an idea developed in Krivochen, 2014c; Saddy, 2018;

Saddy & Krivochen, 2016a, b). Of course, this struggle is not perceptible at certain scales (e.g., if we analyze individual constituents), but is to be found when looking at the process as a whole. This last observation; that is, the idea that local properties of a structure might not map directly to its global properties, is the first step towards a concept that will be crucial in the subsequent development of this work: a *manifold*. A manifold is an *n*-dimensional topological space that *locally* resembles a Euclidean space, and can thus be studied in this manner, taking advantage of the well understood properties of Euclidean spaces (Boothby, 1975: 6)[96]. For instance, the surface of the Earth is globally not a Euclidean space (rather, a Riemann space), but local regions of it can be chartered by using Euclidean geometry, via planes of dimension 2 (e.g., in an atlas). This is essentially what we did with our presentation of CS above, simplifying the dimensionality of non-linguistic material and expressing it in the relatively familiar terms of Jackendovian conceptual semantics. Can we improve that representation, making it more faithful to what we actually propose for the dynamics of cognition? We will attempt to do that, tackling the problem from a dynamical system approach to cognition, applying the theoretical tools of dynamical system theory (crucially, the tension between attractors and repellors) to the physical concept of a *field*, which is essentially a generalization over dynamical systems. The representations we will handle will be studied as manifolds of dimensionality *n*, which interact within a field and generate certain emergent properties from this interaction (in line with interactive models of computation). The concept of a manifold as seen from topology and approximations from dynamical systems theory will prove very useful in analyzing the problem of structure linearization, which we have phrased here as 'dynamical Markovization' (also see Lasnik, 2011; Lasnik & Uriagereka, 2012). In general terms, the cognitive workspace in which structures are generated and manipulated will be taken to be a *field*, whose mathematical and physical characteristics we aim to make as explicit as we can in the remainder of the thesis. The next chapter will be focused on the notion of *field* and how it can inform a theory of syntax, particularly under strongly derivational assumptions under cyclic conditions.

As Ol' Blue Eyes would say, "the best is yet to come".

Part III: (Strawberry) Fields Forever

Let us begin this chapter by defining what a *field* is. Informally, a field is an $\mathbb{R}^n$ space that has a value, or set thereof, associated to each of its points (see, e.g., Feynman, 2006, Vol. 2: § 1-2). For instance, imagine a room that has been divided in arbitrarily small areas of equal volume, each of which has a thermometer. The room is an $\mathbb{R}^3$ space, that has been *parametrized* (divided in smaller zones), and if we map the values given by each thermometer at each area of the room, we will get a *temperature field* for that room. Fields can be excited (e.g., by lighting a fire in one of the little areas), and the value associated to each point can change over time. Field theory is the discipline within Physics that studies how fields form, how they are characterized, how they interact with other fields, and how they change (with respect to time or any other independent variable we may want to consider).

[96] Technically, a manifold of dimensionality *n* (or *n-manifold*) is a topological space M with dimensionality *n* where the following conditions hold (Boothby, 1975: 6):

i) M is Hausdorff (i.e., distinct points have disjoint neighbourhoods)
ii) Each point of M has a neighbourhood that is homomorphic to the Euclidean space of dimension *n*
iii) M has a countable basis of open sets

Fields are often classified by their behaviour under transformations of spacetime, or by the number of components we need to exhaustively characterize them. Here we will only deal with two varieties, *scalar* and *vector* fields:

- Scalar fields (such as temperature) whose values are given by a single variable at each point of space. This value does not change under topological transformations.
- Vector fields (such as the magnitude and direction of the force at each point in a magnetic field, or direction and intensity of wind on a weather map) which are specified by attaching a vector to each point of space.

Fields oscillate in spacetime: they are defined by means of wave functions of the form e^{ikx} (for k a polynomial) in the complex plane (the conjunction of sine and cosine functions, more specifically). We will provide some formulae below, but it is essential to bear in mind that we are dealing with dynamic concepts, which change in time according to well-defined physical laws.

In the following sections, we will explore field effects at different scales, focusing on how the proposal that emergent properties of neurodynamics are best described field-theoretically can explain issues concerning 'what is and what can never be' (paraphrasing Robert Plant, 1969) in language *qua* open, dynamical, complex system under interactive computation assumptions.

3.1 Field theory and neurocognition

If we want to explore fields in the human brain and its computational emergent properties, it is crucial to seriously consider *scale* issues: What kinds of fields are relevant at the scale we are interested in? Can we apply concepts across scales without falling into contradictions or plain errors (of the kind we reviewed above, when considering naïve reductionism proposals between linguistics and neuroscience, for example)? How can we link mesoscopic effects to underlying oscillatory dynamics (and account for the differences in energy, as pointed out to us by Giuseppe Vitiello, p.c.)?

We will focus on the physical properties of wavefunctions, and how the study of those physical properties can help devise not only a theoretically desirable scenario, but also empirical predictions involving well-known facts of natural language as an oscillatory dynamical computational system (these facts featuring *locality effects* prominently); later on we will focus on the topological properties of fields where computational operations are hypothesized to take place. But first, there are further clarifications to make in order to have a non-metaphorical approach to the physical aspects of language as a cognitive faculty. We will deal with the concept of 'field' in both its physical and its topological senses: 'syntax' is globally understood here as a *set of topological transformations over the ground state of the lexicon*, which is an *ultrametric* space, of a specific kind we will deal with in depth in **Section 3.2** below. The 'syntax' is a set of parametrization functions over the sound and meaning continua, rather than a set of combnatory rules over discrete elements. In a word (see Saddy, 2018 for further discussion): symbolic computation emerges when we *limit* an initially unrestricted phase space via the imposition of *normality* (in the formal language theory sense) and *metricity* (topologically) over a *non-normal*, *ultrametric* phase space (as opposed as gradually *enriching* an initially impoverished proto-computational system, which is the MGG view). Physically, 'syntactic objects' are understood as perturbations in an initially completely Hausdorff[97] field once its ultrametricity is disrupted, their relations being defined in terms of operations over states of the field, for which we will resort to Dirac notation (also known as 'bra-ket notation'). Both takes on the notion of 'field' will be essential for the present discussion, and their interplay will prove useful, not only for the purposes of giving our theory a more firm basis, but also as a way to cross-check the provisional conclusions at which we arrive from either perspective: topological claims must be physically plausible, and *vice versa*. We will introduce the theoretical machinery below, but a note is in order (we will insist on this below anyway): unlike many studies in linguistics and psycholinguistics (among other disciplines), we will deal with mutually *orthogonal states*, as is customary in quantum mechanics (see, e.g., Feynman, 2006: Chapter 8), meaning base vectors describing possible states of a system are all at right angles. If two base vectors A and B are at right angles, the probability amplitude of the system in base state A to be measured in state B is zero (we will come back to this below in terms of Dirac notation, but bear this in mind). In terms we have informally introduced above with respect to Garden Path sentences -without loss of generality-, we will say that for any interpretative hypotheses A, B, ...*n* regarding the structure of sentence S (so, say A, B, …*n* are possible structural descriptions of S); A, B, ...*n* are orthogonal to each other (that is, A is orthogonal to B, C, ...*n*; B is orthogonal to A, C, ...*n*, etc.). The basic idea, introduced above, is that *we cannot hold mutually incompatible representations active at the same time*. Sure, we can go back and reparse a given problematic input (say, Garden Path sentences, Relative Clause attachment ambiguity, etc.), but that is actually evidence that for a time *t* we held hypothesis A as the structural description of S; when disambiguating (or coercing) unit U was perceived we realized we had 'screwed up', and *went back* to parse S under hypothesis B (see the Analysis by Synthesis proposal of Townsend and Bever, 2001 and much related work): that is crucially a *sequential* process, not a simultaneous one.

3.2 On Ultrametricity and the topology of the Lexicon

If field theory has a place in neurocognition, which we think it does (and a significant part of the literature agrees, to varying degrees, see for instance the articles collected in Spencer et al., 2009 and the references therein; also Haken, 1983; Schöner and Thelen, 2006; Dineva et al., 2008; Sandamirskaya, 2014, among many others), then there has to be a non-trivial representation of the relevant elements involved in mental computations in terms of fields. Recall that a field is little more than a collection of numbers (scalars, vectors, tensors of higher order…), each indicating the probability of finding a particular element at a particular point in an *n*-dimensional phase space. Thus, an electromagnetic field is a set of probabilities for finding electrical and magnetic activity (in phase with each other) at every point in the relevant phase space. We can map an electromagnetic field to neurocognitively relevant scales, if we are concerned with the possibility of finding electromagnetic activity associated with particular inputs in different points (or sets thereof) of the phase space. Of course, such an approach applies to any kind of stimulus, but we are particularly concerned about linguistic stimuli in this section, primarily because we want to compare and contrast our perspective with that of Uriagereka (2011), which, to the best of our knowledge is the first articulated attempt to deal

[97] A topological space *X* is *Hausdorff*, or T_2, or *separated*, if any two distinct points in *X* are separated by disjoint neighbourhoods. It is *completely Hausdorff* if any two distinct points in X are separated by *disjoint* closed neighbourhoods. The distinction between closed and open neighbourhoods is essential, since bringing points closer together can make their neighbourhoods intersect if these are open. Metric spaces are typically Hausdorff.

with theoretical linguistics from a field-related perspective in paper form. As is common practice in generative linguistics, Uriagereka's take on the matter is sometimes metaphorical (and conditioned by the axioms of the Minimalist Program), which is actually both an advantage and a disadvantage. It is an advantage (for us) because we can stretch the metaphor until it breaks, and thus test the limits of the formalism under strict Minimalist conditions. It is a disadvantage (for us, as well) because the possibilities for comparison are limited, since we do *not* want the reader to be left with the impression that our take on the matter is metaphorical at all. Classically, phrase structure trees (that is, structural descriptions for natural language sentences) have been claimed to be 'metaphors', but the metaphor has never been spelled out. The properties of tree representations as rooted, labeled, oriented graphs (McCawley, 1968) seem to have acquired more importance in recent developments in MGG and other formal frameworks (HPSG, LFG, and even CG), and while explicit mathematical analyses are still minority (and mostly focused on non-transformational theories, given their superior explicitness), there have been some studies on the topological properties of phrase markers within a generative perspective. We will now proceed to discuss some of these studies.

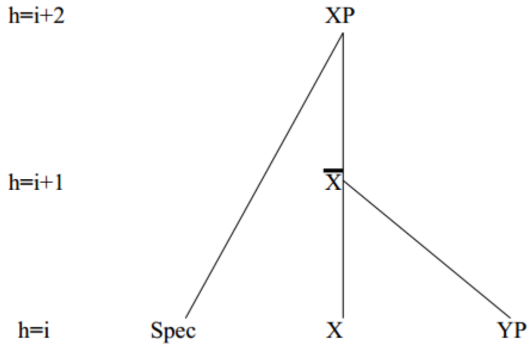
Fields can be looked at from physical and mathematical perspectives. In physics, a field is a quantity that has an associated scalar, vector, or tensor, to each point in space and time. We can then ask how those associated values change in time, what makes them change, etc. We can also ask questions about the topology of the field of our interest, and that is what we will proceed to do next. In mathematics, a field is a fundamental algebraic structure defined by two binary operations (multiplication and addition), two unary operations (the inverses of multiplication and addition), and two nullary operations (the constants 0 and 1, to which the other operations apply). The real, complex, and *p-adic* numbers constitute examples of fields in the mathematical sense. Topological fields are refinements of the mathematical notion of field, such that given a field K, a topology on K is a *topological field* if it is a topology for which the operations mentioned above are continuous maps (Warner, 1989). We will see that there are interesting empirical consequences of committing to specific assumptions about the space in which computations take place and how the 'atoms' of language are represented and related.

As part of his work on syntactic computation in terms of fields, Uriagereka (2011) combines considerations about ultrametricity in phrase markers (which are based on the work of Roberts, 2015) with the added assumption that syntactic categories involved in Merge operations are (perturbations over) ultrametric lexical fields; thus relating two important aspects that we mentioned just above. An ultrametric space is a set of points with an associated distance function d mapped onto the set of real numbers $\mathbb{R}$ such that:

92) $d(x, y) \geq 0$
$d(x, y) = 0$ *iff* $x = y$
$d(x, y) = d(y, x)$
$d(x, z) \leq \max\{d(x, y), d(y, z)\}$ (*Ultrametric Inequality*)

Ultrametric spaces have interesting topological properties. For instance, only a subset of isosceles triangles is allowed (acute isosceles), given the replacement of the *triangle inequality* that holds for metric spaces (in formal terms, $d(x, z) \leq d(x, y) + d(y, z)$, for x, y, and z vertices of a triangle) by the *ultrametric inequality* in (92). Another interesting property is that, for every point within a sphere, that point is the center of the sphere (given a constant distance function between distinct points), something that defies our conscious geometrical imagination, mostly confined to Euclidean figures and polyhedra. Roberts (2015) extends ultrametricity considerations to the X-bar template (comprising the axioms of binary branching, projection, and endocentricity), such that notions like Complement and Specifier are defined in terms of Merge distance from the head, as follows (from Roberts, 2015: 118)[98]:

93)



The use of ultrametricity in hierarchical trees is not novel: Rammal et al. (1986) report and analyze the application of this topology in the development on taxonomy trees in evolutionary biology. Distance functions between elements (nodes) are often expressed as a matrix; when that matrix satisfies the requirements of an ultrametric space, 'it follows that a dendrogram can be unambiguously built' (Rammal et al., 1986: 768). Let us consider such a matrix, where each number is an integer multiplied by the distance *i* between two points (represented in columns and rows). In this form, the ultrametric X-bar tree looks like this (taken from Roberts, 2015: 118-119):

94)

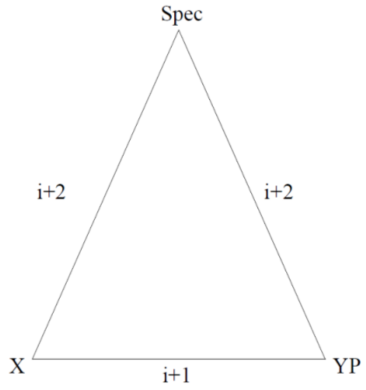
a)

$$\bar{\mathbf{X}} = \begin{array}{cccc} \bullet & Spec & X & YP \\ Spec & 0 & i+2 & i+2 \\ X & . & 0 & i+1 \\ YP & . & . & 0 \end{array}$$

---

[98] Hughes (2004: 149) distinguishes 'classical trees' from '$\mathbb{R}$ trees': the former allow branching at a discrete set of points, whereas the latter allow branching at all points. Formally, '*A real tree, or $\mathbb{R}$ tree, is a metric space (T, d) that is uniquely arcwise connected, and for any two points x, y $\in$ T the unique arc form from x to y, denoted [x, y], is isometric to the subinterval [0, d(x, y)] of $\mathbb{R}$.*' (Hughes, 2004: 152). Interestingly, it follows from Hughes' argumentation that considerations of ultrametricity seem to go better with $\mathbb{R}$ trees than with classical trees. The determination of the points at which branching is allowed for tree-based formalisms (like X-bar theory) is problematic, both from a topological and a syntactic point of view: why, if not by means of an axiom (e.g., stipulations over the alphabet the system works with), does X' branch but not X_0? In the discussion that follows, we will deal with classical trees, which are the object of Uriagereka's and Roberts' inquiry (although implicitly, since the distinction is never made in those works).

b)



Uriagereka imposes conditions over Merge based on the 'distance' between the categories involved in the operation, thus deriving the X-bar template from the ultrametric inequality applied to the triangles (Spec, X, XP) and (X, X', YP), which makes sense if one considers the fact that ultrametricity was first applied to taxonomic trees outside narrow topological studies; however, it is crucial to point out that tree ordering does *not* have to be identified with ultrametricity: both properties have to be proven separately. We see now that we have to worry not only about *generation* number, but also *branch length*, since we are dealing with topological spaces imposing a metric function over nodes, not just abstract mathematical structures as in our discussion of L-systems in **Section 2.2** above. Roberts (2015: 113) claims that '*the greater the ultrametric distance required the more complex a sentence is*', (in a move that is reminiscent of the DTC, and, more generally, of all context-free approaches to complexity; see Marantz, 2005; Ohta et al. 2013 for a linguistic perspective on the DTC) but he offers no definition of complexity (notice that the ultrametric condition is *criterial*, not definitional). It is crucial to bear ultrametricity in mind insofar as the Fib emergent from the tree-like development of an L-grammar in a topological space holds, within Uriagereka's model, if and only if the tree is ultrametric. For Fibonacci patterns to appear in the "X' geometry" (and not only as emergent properties given a set of rules), we have to make certain assumptions regarding the geometry of tree-like 2-D representations:

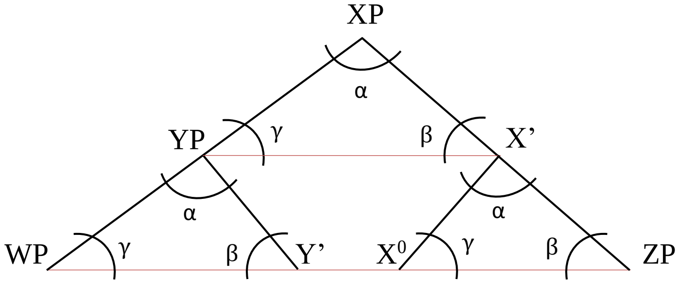
95)

1. There is a system of labeled projection in the syntactic workspace (before the interfaces access the syntactic objects)

2. There are *fixed* ultrametric distances relating nodes

Regarding the second point, consider the geometrized 2-D phrase marker in (94) below (Greek letters name angles, note also the red parallel segments):

96)



There are at least two different possible sets of assumptions that can be made in relation to this, quite independent from each other (see Lasnik & Uriagereka, 2005: 43, ff. for a slightly different take on the following assumptions):

97)

a) Stipulations over branch length: a branch is only as long as it takes to cross the closest parallel segment that stems from the closest branching node –as in the matrix (94 a)- (notice that angles are of no help, since there are infinitely many triangles at each point, all with equal angle measures). This is not problematic at all if the metric imposed to the space is ultrametric. The problem is that, in ultrametric spaces, distances *do not sum* (because of the strong triangle inequality[99]): thus, Specs, Heads, and Complements are all strictly equidistant.

b) Stipulations over the number of projections of a type inscribed in a parallel: includes an additional stipulation over the number of parallels, or, better put, stipulations over the regular distance at which a parallel is significant for the schema. This stipulation also pertains to the possibility of defining 'locality' in ultrametric spaces provided that distances do not sum.

As we can see in (96), the node YP must be the vertex of a triangle defined by the points XP-YP–X', the red lines being the relevant segments to consider for Fib to appear as an emergent property in the development of the L-grammar in a 2-D space, assuming that all nodes are distinct and there is no 0 distance between any two distinct points (see Uriagereka, 1998, 2012; Medeiros, 2008; Saddy and Uriagereka, 2004). As we have seen, however, if we are dealing with an L-grammar, we cannot make any assumptions regarding terminals / nonterminals since these are defined contextually, within a rule; and it is also not possible to determine labels that indicate how an object is to be taken for the purposes of further computations. Eliminating stipulation set (a) –which pertains to naïve metricity-, the segment XP-YP can be as long as one wants (if we are dealing with a metric space, which we will analyze in **Section 3.2.1**): it is plausible (though not desirable) to stipulate restrictions over monotonic Merge, but it is a whole different story when non-monotonic merge enters the game. Without even entering the realm of *n*-dimensional syntax or multidominance theories (Citko, 2011; see also Blevins, 1990 for a graph theoretical approach), Lasnik and Uriagereka (2005: 43 ff.) already present the possibility that the non-terminal YP "stretches" beyond the segment YP-X', even while sticking to Kayne's (1994) LCA (thus violating the ultrametricity condition of Roberts' matrix in $\mathbb{R}^2$): rather than distance, then, we are forced to say that it is the cardinality of each node (i.e., its 'generation number') that counts, but cardinality can only be determined by reference to a *label*, which we have argued cannot be determined for L-systems for they are *not* instances of normal grammars.

[99] A nice, straightforward illustration of this property is featured in Wikipedia (https://en.wikipedia.org/wiki/Ultrametric_space):

> *We want to prove that if* $\|x+y\| \leq \mathbf{max}\{\|x\|,\|y\|\}$*, then the equality occurs if* $\|x\| \neq \|y\|$ [for $\|.\|$ a length function, such that $d(x, y) = \|x - y\|$]. *Without loss of generality, let us assume that* $\|x\| > \|y\|$*. This implies that* $\|x+y\| \leq \|x\|$*. But we can also compute* $\|x\| = \|(x+y) - y\| \leq \mathbf{max}\{\|x + y\|, \|y\|\}$*. Now, the value of* $\mathbf{max}\{\|x + y\|, \|y\|\}$ *cannot be* $\|y\|$*, for if that is the case, we have* $\|x\| \leq \|y\|$ *contrary to the initial assumption. Thus,* $\mathbf{max}\{\|x + y\|, \|y\|\} = \|x + y\|$*, and* $\|x\| \leq \|x + y\|$*. Using the initial inequality* [the ultrametric inequality], *we have* $\|x\| \leq \|x+y\| \leq \|x\|$ *and therefore* $\|x + y\| = \|x\|$

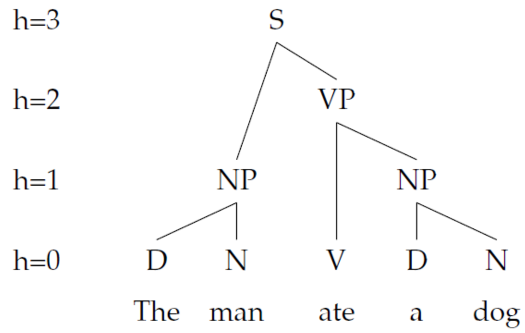
In interface terms, it is difficult to see how the two sets of stipulations in (95) would have any drastic interface effect, particularly when it comes to semantics: if Kayne's LCA is to be maintained, or more generally any proposal that relates linearization to syntactic structure by means of a direct-mapping function, (as Uriagereka seems to want), then some stipulation over the branch length may be necessary to establish one-to-one sisterhood relations when a non-terminal branches *n*-arily (for $n > 2$). We have shown (Krivochen, 2015a) that under LCA assumptions such a stipulation would be semantically and computationally necessary for finite-state structures like coordination and multiple adjunction (and, more generally, all instances of constituent iteration). If the reasoning along the previous lines is correct, then the Fib pattern we have pointed out would disappear from X' theory (actually, a good part of X' would go away as well). Is this bad news? We believe not. Quite the contrary, a different foundation for phrase structure geometry also displays Fib (and Fib-like) emergent properties (and can generate them given a simple concatenation function) without stipulations like (a) or (b). Moreover, a pattern, if really there, should be the optimal resolution of a tension, a CLASH –to use the acronym in Uriagereka (2012, 2014)-: we do not see such a tension in the algorithm generating the Fib tree (28) above, as it is crash-proof from the very beginning, only generating what is expected to be generated (a 'self-fulfilling prophecy' of sorts). Quite the same objections apply to the derivation of conditions over constituent dependencies established by Uriagereka (2011: 30, ex. 58), according to which, apparently, *shorter* paths lead to ungrammatical results, against most Minimalist *desiderata* (e.g., Chomsky's 1995 *Minimal Link Condition*):

98) a. What did [rumors about tortures] cause [reports about _]?
b. *What did [rumors about _] cause [reports about tortures]?

Granted, we should have to assume that we are dealing with a 2-D ultrametric space, and, moreover, with a *single* derivational space. It is relevant that the gaps appear within adjunct constructions, which, as Uriagereka (2002a) and Krivochen (2015a, b) show, are derived in parallel workspaces (both authors arrive at the same conclusion while departing from different considerations: Uriagereka's concern was to linearize non-Markovian strings, whereas ours was to account for semantic dependencies between objects in cases of so-called *displacement*). There is no reason to believe *workspaces* are related by means of distance, that is, by any *metric* function. There is no explanatory value in saying that adjuncts are 'further away' from heads than complements (assuming for a moment notions like 'head', 'complement', and 'adjunct' have extra-theoretical status). Rather, in past works we attempted to explain constraints on displacement (which, in our theory, does not amount to literal *movement* but to the instantiation of a structural or lexical *token* in the active workspace, see Krivochen, 2015b for details) by resorting to the assumption that in order to extract α of arbitrary complexity from a syntactic object SO and Merge it to β, both SO *and* β must be active in an 'episodic buffer' (Baddeley, 2007), or a working memory in more general terms, at the point of dependency establishment. In traditional generative terms, complements are *monotonically* assembled with respect to their selecting heads, whereas specifiers (and adjuncts) are not (Uriagereka, 2002a). The constituent [rumors about X] is in X-bar theoretical terms the 'specifier' of the [cause] primitive (syntactically represented by the category *v*), in semantic terms, it is interpreted as the external initiator of the event denoted by the *v*-V eventive complex. Therefore, there is an extra condition applying to [rumors about X], which does not apply to [reports about X]: not only do we have a workspace condition to account for because of adjunction (if we assume relative clauses, regardless their abridged character, are adjoined to the relevant NP/DP), but also the non-monotonic Merge of the specifier indicates the relevance of considering a separate workspace. Extraction from an adjunct contained in a specifier is thus predicted to be considerably more restricted than extraction from an adjunct contained in a complement (oddly enough, this is something Uriagereka himself notices in his 2011 paper, but he does not apply this reasoning to (95 a, b), which show precisely this pattern).

Crucially, ultrametric approaches to phrase structure are forced to set the monotonic / non-monotonic structure building distinction aside (which of course does not arise in decision or taxonomical trees). Let us consider, for instance, the tree representation Roberts, 2015: 119 proposes for [the man ate a dog], where there is no distinction between [the man] and [a dog] with respect to their derivation, (and see Uriagereka, 2002a, for arguments in favour of making just such a distinction)[100]:

99)



According to Roberts (2015: 119), (99) is ‘the correct tree’ representation for [the man ate a dog], because all nodes occur at the lowest possible height. Consider, also, the definitions of *domination* that Roberts assumes:

i) h(A) is higher up or at the same height on the tree as h(B) i.e. h(A) ≥ h(B)

*ii) it is possible to trace a path from **A** to **B** going only downward, or at most going to one higher node.*

In this framework, c-command is defined as usual, A does not dominate B, and B does not dominate A, but the first branching node that dominates A also dominates B. The problem is that the clause ‘or at most going to one higher node’ subsumes c-command to domination, for ‘going one higher node’, in a strictly binarily branching structure, means going to the first branching node. Let us consider the relation between [$_{NP}$ The man] and the VP. Here we have that *h*(VP) > *h*(NP), and it is possible to go from VP to NP by going up one level and then down…does that mean VP dominates NP? It is not clear that is a desirable conclusion, particularly given the importance that dominance has for modules of the theory like Binding. It is true, however, that, if theta-theory is taken in its GB sense, then having the theta-assigner structurally higher than the assignee is practical. Of course, the practicality of this depends on theta-roles being *assigned* by a head to a position in the tree (not unlike Hornstein’s 2001 approach to Control as movement in order to check theta-features, by the way). If theta-theory is tackled in a ‘Hale and Keyser’ way, theta-roles being read off configurational relations in a more global way, and arising at the syntax-semantics interface (rather than being assigned at D-structure, as in GB; or valued and checked, as in the MP), the supposed advantages of (97) in theta-theoretic terms vanish. We are assuming, crucially, that not having theta-roles as

[100] No animals were harmed in making this tree representation. The example is Roberts’.

independent primitives in the theory is better than having them, provided that we can account for the semantic interpretation of arguments in a purely compositional manner, with no substantive elements added to representations (*a la* Hale & Keyser, 2002 and related work; also Krivochen, 2012a: 77, ff.).

In any case, for V to theta-mark NP, they have to be in the same workspace at the moment of marking, since before that, the position occupied by NP is either empty (if the structure is taken as a template to be filled with phrases) or nonexistent (if trees grow via successive applications of generalized transformations, which create a Ø position adjoined to an object α yielding {α, Ø} and then replace Ø with β, thus creating {α, β} (Chomsky, 1995: 189; Kitahara, 1994: 50, and much subsequent work). The process by means of which NP is introduced at $h = 1$ and then somehow related to S at $h = 3$ is not made explicit, and -while we can speculate- it is a problem to be subsumed under the 'address issue' proposed by Uriagereka (2012: 75) as the problem of putting different derivational units together after Multiple Spell-Out; there is no *explicit* way of implementing Uriagereka's proposal to replace a separately Spelled-Out term M_K within a term K with an identical M atomized after S-O (in his proposal, M behaves 'like a word', not a syntactic compound but something akin to a syntactic terminal; Uriagereka, 2002a: 49, ff.) *in an ultrametric representation*. A correct implementation of the address issue requires that there be *zero* distance between points (after all, *identical* is not topologically defined in the formulation of the address issue, it just means that M_K and the Spelled-Out version of M must bear the same index and be equally complex), such that $d(M, M_K) = 0$. Even if this worked, we still need an additional specification about the distance between M (in this case, the NP) and its mother node, $d(M, S)$. But we have no way of calculating the hypotenuse of that triangle, for we do not know the length of the short side (note that the matrix formulation does not specify the ultrametric distance between a head X and the head of its complement YP; also, note that for the matrix to be applicable to the tree in (96), we would have to assume that D actually projects DP, and is the Spec of NP, and so on: there is no way of avoiding an exponential proliferation of labels most of which are useless for the purposes of further computations): all we have is the long side, which is 3-1. There is just not enough information to ensure that a connection between NP and S is even possible in a mildly derivational system (a GB-style model). If these problems already arise with a simple, fully overt declarative sentence, imagine how dense things could get with, say, parasitic gaps or ATB (even from a non-transformational point of view, since the address issue is still a problem for any system that implements derivational cycles).

Phrase structural ultrametricity constitutes networks, in Uriagereka's system, and the lexicon is indeed conceived of as a network of statistically weighted connections. More abstractly, an individual's lexicon is an underlying field of connectivities between unobservable states, which become observable only after their Markovization (i.e., Spell-Out). The topology of this field is ultrametric, and thus each element of the lexicon should be connected to every other by a constant distance $d(\#x\#, \#y\#)$, where x and y are terminals. In principle, there should be no objection to this assumption, unless linguistic experience is taken into consideration, and some connections between items are reinforced[101]. With reference to this point, Uriagereka (p.c.) insightfully says that

[101] 'Collocations', in terms of Halliday and Hasan (1976: 284), are elements that 'tend to co-occur', an informal yet statistically-related notion, too narrow for our purposes, as we claimed above.

The idea is that when you merge, say, "men" into "like arguments" (or some such), ***you are literally getting "men" to a proximity w.r.t. "arguments" that it would not otherwise have had*** *(as compared to, say, "men" and "boys" or "arguments" and "discussions", say). As a consequence of the merge, each of the relevant words (understood as information-density peaks within the space) will obtain new conditions. If the merge had never happened before in a speaker's mind, the contribution of each new word would be huge. For instance, if you have just heard for the first time "men like arguments", then you will learn that "men are such that they like arguments" or that "arguments are the sorts of relations that men engage in". When the merger has occurred already (enough to take the relevant specificity to be present in a subject's mind), the new association will strengthen the salience of given properties.* (our highlighting)

The model sketched in this passage is not unlike connectionist approaches to lexical networks, and it is likely such a proposal can be implemented in a connectionist network (Thomas, 1997; also Dell, Chang, and Griffin, 1999): the model Uriagereka proposes allows for the adjustment of activation weights, such that the relevant field is not only dynamic, but capable of 'learning'. However, there are some unsolved issues, not the least of which is the connection between sound and meaning: it is not obvious that 'hearing' a particular sentence or phrase should trigger an adjustment in the conceptual field, nor is it obvious that the network should ever syntactically relate, say, [men] with [like arguments] -i.e., Merge them in MGG terms- if it has never done so before. The latter problem touches on the issue of crash-rife vs. crash-proof grammars (Putnam, 2010; Stroik and Putnam, 2013; Putnam and Stroik, 2016), and bidirectional optimization algorithms (Jäger, 2002, among others): what triggers the connection between *n* (in Uriagereka's example, two) manifolds or field perturbations that have never interfered before? And, even if we came up with a 'first-time trigger', can we make it work again? We think the answer is 'yes'; moreover, it is straightforwardly modelable under field theoretical assumptions, particularly considering that what we could provisionally and pre-theoretically call 'words', or individual elements in the lexicon[102] are in fact defined as information-density peaks within the phase space ('perturbations', in a word), in a Dynamical Field Theoretical approach. We will come back to this below, once we have introduced some more notation and concepts about the operations we assume take place within the relevant phase space.

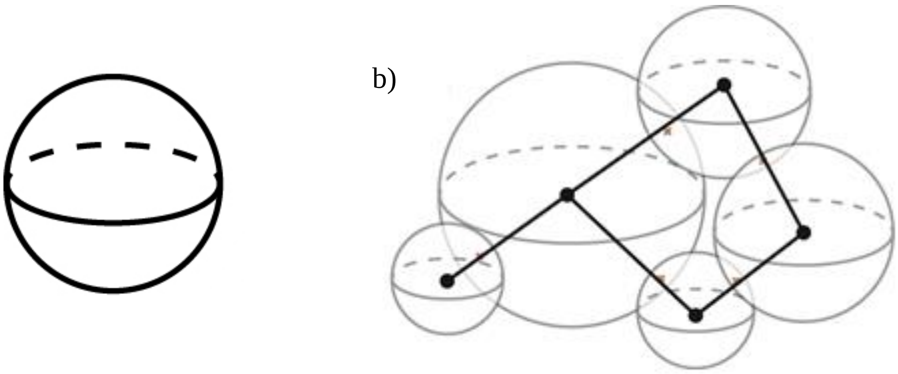
Going back to a point introduced above about the methodological isolation of the lexical ultrametrical field, including experience in the picture (along with, say, wider cultural factors) would lead to the development of more than a single lexicon-field; rather, it would lead to multiple fields, each structured as an ultrametric sphere (where all points are equidistant). Interactions between (or among) fields would, of course, be permitted, but predicted to involve a distance $d + n$, where d is the ultrametric distance between any two points within an ultrametric sphere, and n it the distance that separates fields. Both situations are illustrated below[103]:

[102] It might be worth saying that the theories that consider idioms as constructions stored in the lexicon are not incompatible with this formulation: we have not developed a theory of what gets stored and how. Conceivably, Construction Grammar approaches (e.g., Goldberg, 2006) –and Culicover's 'Syntactic Nuts'- are compatible with a field theoretical approach as well, if we can represent 'constructions' (atomized sub-derivations) as base vectors (and, *a priori*, there is nothing against that possibility).

[103] Images have been taken from https://conan777.files.wordpress.com/2011/09/kissing-1.jpg and https://conan777.files.wordpress.com/2011/09/kissing-2.jpg (no copyright infringement intended), where

100)

a) b)



However, we haven't appropriately justified our acceptance of ultrametricity in order to account for the topological properties of the relevant fields in their initial state, before being syntactically transformed. We will do this now.

3.2.1 Why ultrametricity?

In this section we will discuss and defend the choice of ultrametricity as the mathematical model for the topology of both the conceptual and the phonotactic spaces in their ground states (i.e., without excitations or interferences). The first question to be asked is: why not adopt a *metric* space to describe the ground state dynamics? A *metric space* is a specific kind of topological space in which the following properties hold (d is a metric function over a set of points, or simply put, *distance*):

101) a. $d(x, y) > 0$ if $x \neq y$
b. $d(x, y) = 0$ *iff* $x = y$
c. $d(x, y) = d(y, x)$
d. $d(x, z) \leq d(x, y) + d(y, z)$ (*triangle inequality*)

(for all $x, y, z, \ldots \in \mathbb{R}$)

Metric spaces define Euclidean topologies, including the perceptually familiar 3-D Euclidean space. Distances, except in special cases, are not only real (with zero imaginary part[104]), but also positive (i.e., $d(x, y) \in \mathbb{N}$), and they are not constant: two *distinct* points x and y can be arbitrarily near or far apart, but never have 0 distance (given 101 a, b). We will argue in the following sections that, although metricity is not an inherent property of the *conceptual* or *phonotactic* spaces for reasons of accessibility, among other things, 'syntax' is best described as a *topological* operation that *dynamically introduces a metric function between perturbations on an otherwise ultrametric field*. The linguistic workspace is, we argue, essentially *metric*[105]. This

they are used to illustrate a completely different mathematical problem. However, we found them amazingly suitable for our own purposes (consider, for instance, that in an ultrametric space, every point within a sphere is the sphere's center).

[104] That is, for a complex number $a + i \times b$ ($i = \sqrt{-1}$), we set $b = 0$ (thus, the whole imaginary term equals 0).

[105] For instance, Zwarts and Gärdenfors' (2015) analysis of locative and directional prepositions in terms of relations between NPs within a polar coordinates system can be read as an argument for the metricity of the linguistic space, since the notion of '(polar) betweenness' that is essential for their argument crucially requires points to be related by a variable distance relation. Even if their argument for polar coordinates was rejected, Cartesian 'betweenness' also rests on *metricity* as a ground assumption. In a

perspective entails, of course, that we do not conceive of ‘syntax’ as a set of generative rules. Before getting into details about the operations that take place in the topological workspace, however, let us justify our choice of ultrametricity as the initial topology of the mental workspace.

The first non-metric space to be taken into consideration will be the *pseudometric* space. A pseudometric space is defined as a generalized metric space (as the pseudometric space can be seen as an extreme case of metricity under special considerations) in which the following conditions hold:

102) a. $d(x, y) \geq 0$ (for $x = y$ and $x \neq y$) (*indistinguishability clause*)
b. $d(x, y) = d(y, x)$
c. $d(x, z) \leq d(x, y) + d(y, z)$ (*triangle inequality*)

(for all $x, y, z, \ldots \in \mathbb{R}$)

Note that we are missing identity as a condition for any two $x, y \in \mathbb{R}$ to have a distance 0: this is one of the essential properties of pseudometric spaces that we will be looking at. Unlike points in metric (and ultrametric) spaces, distinct points in a pseudometric space *need not* be distinguishable: a pseudometric does not require that the distance between two *distinct* points be strictly positive. The following is an example of a pseudometric space (taken from http://planetmath.org/exampleofpseudometricspace [accessed on 9/1/2015])

> *Let $X = \mathbb{R}^2$ and consider the function d: $X \times X$ to the non-negative real numbers given by*
>
> $d((x_1,x_2),(y_1,y_2))=|x_1-y_1|$
>
> *Then $d(x, x) = |x_1-x_1| = 0$, $d(x, y) = |x_1-y_1| = |y_1-x_1| = d(y, z)$ and the triangle inequality follows from the triangle inequality on $\mathbb{R}^1$, so (X, d) satisfies the defining conditions of a pseudometric space.*

For example, we can have a case in which $d((3,3)\,,(3,5)) = |3-3| = 0$; which illustrates the non-distinguishability clause. While most geometric accounts focus on the triangle inequality, our interest is set on the indistinguishability clause (102 a): if we considered the L-grammars for Fib and XOR as developing in a pseudometric space, we would need to have a special stipulation to prevent the collapse of any two points to 0, since such a situation would, among other consequences, make both sequences indistinguishable for any two members (and sequences of pairs of members, for a string). Moreover, we saw in **Section 2.2** above that operations of *collapse* and *percolation* (as well as Uriagereka’s *atomization* and *pruning*) are incompatible with the very nature of L-grammars due to their simultaneity and the lack of representational stability that comes with the absence of terminal-nonterminal distinctions, which sets them apart from any computational procedure based on Von Neumann’s architecture, including TMs (apart from the empirical fact that we cannot have 0 distance between, say, bacteria during reproduction: biological implementations of L-grammars are always in metric spaces and *require* topological distinguishability as a formal condition; their formal growth, however, fills an *ultrametric* space).

non-metric space –either ultrametric or pseudometric-, there is no way of formulating ‘betweenness’, for all points are equidistant.

This situation, which can be overcome in a purely formal scenario, becomes unsolvable when we consider the phonotactic and conceptual spaces, even though they are not *strictly* biological (but *emergent properties* of the activity of biological systems; this distinction is by no means minor). The reason is that distinguishability is a *necessary* condition for any mathematical model of those spaces. Empirically, categorical perception provides a good example of why distinguishability must be 'hard wired' into the model (see Harnad, 1987; also Goldstone and Hendrickson, 2009 for discussion and literature review), even if 'categories' are taken not as discrete, static points, but as attractors in the phase space where the manipulation of categorized symbols takes place (which is both physically and mathematically more sound than fuzzy-logic alternatives; Damper and Harnad, 2000; Spivey, 2007: 146, 154-157). Evidence of categorical perception of sounds has been found in humans even before birth, and it is a capacity shared with several other animal species, like chinchillas (for discussion at an introductory level, see Altmann, 1998: Chapter 3). Even though it is very unlikely that all (major) conceptual categories are innate, as strong nativists suggest, it has been experimentally found that the primary and more basic categories for at least colour and sounds are innate, even though their 'boundaries' are, predictably, shaped by experience, culture, among other factors (see, e.g., Roberson, Davies, and Davidoff, 2000, who claim that linguistic categorization *facilitates* recognition of colours, but also that there is a limit to what can be attributed to culture and lexis). Categorical perception, and its development, is intimately related to the configuration of the phonotactic space, defined as the space in which all possible states of the system are represented, with each point in an *n*-dimensional graph unambiguously denoting a state of the relevant system. In the case we are interested in, each point defines a discrete sound, and, for the purposes of exposition, each possible distinctive feature configures a dimension such that the allophone [p^h] would be defined (broadly speaking) by the matrix (+, +, +) along the axes *anterior*, *plosive*, *aspirated*.[106] In this way, each possible sound is unambiguously defined, and each language makes use of a subset of points within the phase space, establishing allophonic and phonemic distinctions as the phonological system is defined in that phase space for a language L. The selection of the relevant portion of the phase space (all of which is, in principle, available as it comprises all possible distinctive sounds a human can produce) is, in these terms, what constitutes phonotactic learning. Notice that this is a refinement of our initial hypothesis about what constitutes learning in terms of grammar assignment generalizations in **Section 1.5**: when formulating that thesis, we were concerned with structure assignment, for which the problem arises of multiple hypotheses ('infinitely many', according to Heinz, 2007: 25) compatible with any given input[107]. Here, each relevant element in the phonotactic space is unambiguously defined in the phase space, and structure (i.e., relations between the relevant

[106] Needless to say, the number of relevant distinctive features is considerably higher than those we have taken into account here, and if we are aiming at a rigorous characterization of the phonotactic space, each sound would have indeed to be defined in an *n* by *n* matrix taking into account *all* possible distinctive features and their combinations. That said, we will not attempt such cartography of the phonotactic space here, since our focus in the present discussion is simply to prove that distinctness is an essential condition of such a space. We thank Susan Schmerling for useful discussion about this point.

[107] Within a more narrowly linguistic framework, Ross (1969a: 261) argues that '*every sentence containing a sluiced question would be derivable from an infinite number of distinct deep structures*', in a non-trivial manner. We believe the problem is ultimately the same: given a sluiced question, can a TM implementing a transformational grammar match it with the corresponding interpretation (i.e., the corresponding D-structure / base-generated phrase marker) in polynomial time without resorting to some kind of external controller?

segments) arises only when the units have been identified. The problems of structure assignment and unit identification are thus complementary.

What happens in the conceptual realm? In principle, we would like the distinctness condition to hold for conceptual spaces as well as for the phonotactic space, thus making pseudometric approximations equally inappropriate for both. And, in principle, we will assume the distinctness condition applies for both, as a matter of fact; we thus cannot use the '≥ 0' part of the distance function in (102 a). The reason for this is that, just as we want to be able to define each possible sound in the phonotactic space unambiguously, we also want the building blocks of conceptual structures to be equally distinguishable and unambiguously defined given a certain phase space. However, at the level of implementation, we have certain problems: all in all, we can define the dimensionality of the phonotactic phase space in terms of distinctive features. Just as, given a plane (a 2-dimensional surface), all we need is two coordinates in order to precisely locate any point within that plane, given a sound, all we need is a number of coordinates equal to the number of distinctive features of the sound so that no information is lost, and so that localization is precise. The same reasoning applies to concepts, in principle, but to the best of our knowledge, there is nothing like 'distinctive features' for concepts. An objection to this last claim could come from the side of Conceptual Semantics (e.g., Jackendoff, 1983, 2002) and related approaches, but we must take into account that those approaches (just like Distributed Morphology –e.g., Acquaviva and Panagiotidis, 2012- approaches and Exoskeletal models –e.g., Borer, 2014-, both of which are concerned with the syntactic properties of roots) do not present an unambiguous formalization of the content of roots, even though they seem to agree that roots are more primitive than morphemes or words, in both a linguistic and an ontological sense. More importantly, as we have argued in previous works, we partially follow Boeckx (2010, 2014) in distinguishing *concepts* from *conceptual addresses* (i.e., roots):

> *Using an idea of Paul Pietroski's* […]*, we can think of these addresses as instructions to "fetch" or activate concepts. Like addresses generally, conceptual addresses are not meant to mirror the entities they point to. Think of snailmail or email addresses: they are silent on the physical specifications of the person they are tagging, or the arrangement of the furniture. They have a flattening, atomizing effect. This is what I want conceptual addresses to do: I want them to activate concepts, but they are not concepts themselves, they are only one (efficient) way to get to concepts. It's important to stress that even if, as I am claiming here, conceptual addresses are atomic, I am not making any claim regarding the atomic character of concepts themselves. I personally think it quite likely that concepts are highly structured, not just in human minds, but also in other animals', but it means that as far as human language syntax is concerned, we will be able to find precious little about the structure of concepts* (Boeckx, 2010: 27-28)

The core idea we agree with is that language does not actually manipulate concepts (which are defined in a higher dimensionality space than language can manipulate, as we will see below when analyzing the center manifolds of cyclic derivations near critical points as mechanisms that reduce the dimensionality of a syntactic manifold), rather, language manipulates instructions to fetch low-dimensional counterparts of concepts. The reason for this is that concepts play a crucial role in cognition, and thus not language-specific or anything of the sort. If this is the case, then there is no reason they are to be subject to the same 'decomposition' approaches. Linguistic roots are (or can be) 'decomposed' into elements that are considered to be more 'primitive' (like Cause, Event, Location, see, e.g., Mateu Fontanals, 2002, 2005; and most importantly McCawley, 1968 and related works in Generative Semantics), but this is not applicable to concepts, complex though they might be, because it is not clear that the same 'primitives' that apply the linguistic and conceptual domains, where, we have claimed, the former is more restricted than the latter. Since roots are dependent on concepts (we can even think of roots as the *Sinn* of concepts in a Fregean sense, that is, their 'mode of presentation', Frege, 1952: 57; which would fit in nicely with Boeckx's approach to 'conceptual addresses') but not the other way around (as concepts can, in principle and unless there is evidence to the contrary, exist even without being linguistically articulable). One approximation to something like distinctive features of the [+ / - *magnitude*] for non-linguistic notions is a Vendlerian approach to *events* (crucially, Vendler's classification can be extended from the narrow domain of verbs to apply to *roots* regardless their 'categorial' instantiation, with [die] and [death] sharing Aktionsart), such that we have the relevant dimensions *telic* and *durative* (Vendler, 1957; notice that the latter is actually a *temporal* dimension; and the former, a *boundary* on the latter). However, taking this approach as a general model would be too restrictive: nothing is said about sortal entities and, moreover, it is not clear whether concepts are specified with respect to their *denotata,* nor is it even obvious that they should have *denotata* (it is possible that *concepts **are** denotata* of *signifies*). The theory is also narrowly restricted to concepts instantiated linguistically (roots, or, in Boeckx's 2010 terms, 'conceptual addresses'), and it makes no predictions whatsoever about concepts themselves. These problems make it very difficult to *implement* (but crucially not to formulate) a multidimensionality approach to the conceptual space in comparison to the phonotactic space.

Now, do we want the relevant spaces to be *metric* in nature (as opposed to *metrizable*, a property we will come back to below)? The scales of analysis (or 'levels of organization', to use a term more in line with physics) are also different, and thus call for separate arguments. This requires at least a brief argumentation regarding why we would reject metric considerations applying to the phonotactic and conceptual spaces. The reason was partially laid out above: (non-biased) *accessibility*. Since metric spaces are based on variable distances; that would mean that some elements are 'further away' from any arbitrary point (say, for concreteness, the origin of coordinates, be it Cartesian or polar, see e.g., Zwarts and Gärdenfors, 2015 and references therein) than some others. That is, *from* that point, some elements would be more easily accessible than others, because they are *closer*. As we have said, for the initial, undisturbed state of the conceptual and phonotactic spaces, this is not a formally desirable situation even though purely statistical considerations might (mis)lead us (in)to think(ing) that some elements are indeed more easily accessible than some others because they occur more frequently: this kind of consideration does not pertain to the structure of the field itself, but rather refine the properties of the relevant parser and the heuristics it uses to find the relevant element within the ultrametric field (if it finds anything at all). However, frequency-based parsers fall short when it comes to explaining structural sensitivity, as argued by Saddy (2009) by using strings generated by means of the Fib grammar (whose derivational and representational peculiarities we addressed in **Section 2.2**). When describing the properties of the fields, metric considerations are not quite appropriate, either theoretically or empirically. We take the arguments in this section to be pointers at least towards the advantages of an ultrametric formalization of the phonotactic and conceptual spaces, and, narrowing our scope to language, also of the lexicon[108]; crucially,

[108] While it should be obvious by now, we cannot overemphazise the importance of bearing in mind that the lexicon is narrowly linguistic, whereas the conceptual and phonotactic spaces are not. Rather, conceptual and phonotactic structures can be linguistically instantiated via a process that, as we will argue (and illustrate) below, requires a dimensional flattening, and also possibly a change in the metric

however, this topological characterization holds *iff no syntactic operation targets elements belonging to those spaces*. The characterization of conceptual and phonotactic spaces that we have been making in the last paragraphs is arbitrarily static, and does not take into consideration factors external to the respective fields that can make perturbations interfere in different ways. The reason we adopted this position is that, before characterizing a topological transformation over those spaces, we wanted to have a picture of the 'pre-derivational' (pre-syntactic) state of the relevant fields, *before a metric function is imposed over those fields*. This picture will allow us to have a more precise set of tools to define what *syntax* is and how it performs a dynamical metrization of ultrametric spaces.

A further argument in favot of ultrametricity as the topological model for the pre-syntactic state of the workspace (its *ground state dynamics*, see also Saddy, 2018) comes from considering the very origin of such workspace: recall that in **Section 2.9** we introduced the concept of *dynamical frustration*, the optimal resolution of the clash between opposing tendencies. This concept arose in the context of the study of spin glasses, which are 'disordered' magnets, lattices in which electrons are subject to a pairwise antialignment constraint, which makes the system locally frustrated (since an electron has to be changing spin permanently in order to maintain the antialignment with its neighbours, a situation we illustrated in (80) above). As Rammal et al. (1986: 771, ff.) argue, '*The crucial ingredients in these models* [of spin glasses] *are disorder and frustration*.' Disorder is understood in its usual 'entropy' sense -see Caracciolo and Radicatti (1989) for discussion about the entropy of ultrametric dynamical systems-. Frustration is one of the core topics of the present thesis. Since the system *as a whole* cannot achieve a stable state, instead, multiple locally optimal solutions are found by actants in the dynamical system. If, as we have argued here, 'natural language' is essentially the result of a dynamical frustration between global and local tendencies (corresponding to semantic and morpho-phonological requirements, respectively), it makes sense to think that frustration can generate an ultrametric space which is progressively transformed by means of syntactic operations. As we will see, derivations can be modeled to proceed following a *least effort* path, zig-zagging through the workspace from energy minimum to energy minimum. This process is much better captured if we do not assume from the beginning that the topology of the 'syntactic workspace' is static. A derivation, then, would proceed cyclically from ultrametric to metric and to ultrametric again (see Saddy and Krivochen, 2016a; Saddy, 2018 for more discussion), as syntactic cycles (which correspond to *manifolds*, an idea we already developed in Krivochen, 2015a, 2016c) are completed and reach a critical dimensionality value that spans the center manifold, squeezing the system through a faded dimensional attractor, into a metric space of lower dimensionality but richer in distance functions. The ground state of neurocognitive dynamics is an ultrametric space which is not only high-dimensional but also unrestricted, lacking hard conditions. Given such a space, there will be an infinite number of *n-dimensional* manifolds which, provided the topology of the space is disrupted – as from external input - [109]they may intersect. In other words, for while in an ultrametric space the distance function d between a point $x \in$ X and a point $y \in$ Y (for X and Y manifolds) is fixed and > 0, the

considerations regarding the topology of the relevant workspaces. That is, the arguments for the ultrametricity of the phonotactic and conceptual fields are not to be automatically extrapolated to other fields at the same or a different scale: that would imply a strong uniformity thesis, against which we have extensively argued not only here, but also in previous works.

[109] See Schöner (2009: 33) for some discussion about internal and external perturbations on dynamical systems.

intersection of X and Y at x, y means that $d(x, y) = 0$. Interpretation, we will argue, can only occur in a *metric* space bounded by hard conditions over usable outputs, since linguistic elements are closer to those they are to be interpreted in relation to (*trajectors* and *landmarks*, in the terms of Zwarts and Gärdenfors, 2015: 10)[110]. The idea that the representational space is a Hilbert metric space and structures are essentially defined in vector fields is also a core assumption of Gradient Symbolic Computation (Smolensky et al., 2014), however, GSC does not implement a computationally mixed engine nor does it deal with varying metrics; the programs are potentially compatible, yet not equivalent. In the case of adverbials and prepositional constructions in natural language, the possibility of having intensifiers ('*very* close') and comparatives ('*farther* away') which locate elements in relative terms to one another and to landmarks calls for a variable metric function over a Euclidean space rather than for ultrametricity (e.g., Zwarts', 2003 *polar coordinates* proposal). In this sense, relating elements *qua* field perturbations is simply making them interfere: such interference does not take place if the topology of the field is untouched, because the ground state is a weakly associative, ultrametric field. This conception of syntactic derivations, in terms of oscillatory dynamics, will be expanded on below.

Before expanding on the nature of syntax and the transformations it performs over an initially ultrametric space, however, we would like to expand a bit on the properties of the generative component assumed in linguistic theory *qua* formal grammar, and the relation between the structure of languages as represented by tree diagrams and topology by means of well-known mathematical-topological structures: fractals.

3.3 Some more topology: fractal considerations[111]

An important point we have been asked about many times concerns a possible fractal nature for the object we are modelling, as well as a possible equivalence between recursion and fractal objects. Before getting into the discussion, let us clarify what are customarily taken to be the basic properties of fractal structures (based on Mandelbrot, 1983; Falconer, 2014; Stogartz, 1994: Chapter 11; Lapidus and van Frankenhuijsen, 2006):

103) a. They display structure at arbitrarily small scales
b. They cannot be described by '*traditional geometric language*' (Falconer, 2014), either locally or globally

---

[110] Consider, for instance, the formalization of the meaning of the prepositions 'inside' and 'outside' in terms of polar coordinates (Zwarts and Gärdenfors, 2015: 11), for x = distance between a *trajector* (a.k.a. *figure*, see Zwarts, 2003) and the *space of a landmark* (a.k.a., *ground*) $S(L)$; θ = angle between the trajector and the x-axis; ϕ = angle between the trajector and the positive z-axis; and r_L is the radius of the landmark L:

$$\text{inside}(L) = \{\langle x, \theta, \phi\rangle \in S(L) : x \leq r_L\}$$
$$\text{outside}(L) = \{\langle x, \theta, \phi\rangle \in S(L) : x > r_L\}$$

Since landmarks are assumed to be circles in this work, if the linguistic space (as opposed to the ground state of cognitive dynamics) was ultrametric, then every point would be at a distance $x = r$, neither inside nor outside $S(L)$, or, rather, if $x = 0$, every point would be the *origo*. The semantics of prepositions (and, more generally, localist theories of cognition) thus *requires* metricity.

[111] We are particularly grateful to Ana Bravo and L'udmila Lacková for questions about the fractal nature of language, which have led us to write the present section.

c. They display self-similarity (either strictly geometrical / topological, or statistical)

d. Their dimensionality is often[112] a non-integer

e. They are not defined in Euclidean metric spaces, such that $d(x, y)$ can indeed range from 0 to ∞

f. They are almost always[113] obtained via iterative procedures

From our perspective, these are issues to take into consideration before giving a definitive answer to the question whether a relevant natural object is a fractal in a way that concerns us. We will add two more questions, which arise when considering the problem from the point of view of linguistic theory:

104) a. Do we adopt a top-down or a bottom-up approach?

b. Does the algorithm generating the object ever halt?

Let us consider (103) and (104) in order. Self-similarity is quite self-explanatory, at least in mathematical terms: informally, we say that a fractal contains copies of itself at arbitrarily smaller scales. The copies, it must be noted, might bear only a resemblance to the whole, in which case we talk of a *statistical* self-similarity, not a strict *topological* self-similarity. For instance, the boundaries of a Mandelbrot set are different kinds of Julia sets for different critical values. We can perceive the difference in the *geometrical* representation of the set, that is, when the set of complex numbers $\mathbb{C} = (c, c^2 + c, (c^2+c)^2 + c, ((c^2+c)^2+c)^2 + c, (((c^2+c)^2+c)^2+c)^2 + c, ...)$[(114)] is drawn in the complex plane, generating the following well-known figure:

[112] This depends on the notion of 'dimensionality' we consider. In general, Hausdorff dimensionality (defined as the ratio between natural logs) is the path to take, and we will adopt that formulation. In any case, there *are* fractals with an integer number of Hausdorff dimensions: a Hilbert or a Peano curve filling an *n* dimensional space will have *n* dimensions. Perhaps less trivially, Julia sets have a Hausdorff dimension 2 for some critical values, and so does the boundary of the Mandelbrot set (Shishikura, 1991). Mandelbrot (1983: 15) also considers Brownian motion to have integer dimensionality, yet it is a fractal. In general, we can say that every object with non-integer dimensionality is a fractal, but there are fractals with integer dimensionality.

[113] The graphical representation of the motion of a Brownian particle is fractal, but not obtained by means of recursive or iterative procedures. The same seems to be the case with most (if not all) naturally occurring fractals, including coastlines, rivers, and so on. Falconer (2014: xviii) also opens the possibility of these kinds of fractals when claiming that '*In **most** cases of interest,* F [a fractal set] *is defined in a very simple way, **perhaps** recursively*' (our highlighting).

[114] More succinctly, the Mandelbrot set is the set of values of *c* in the *complex* plane for which the orbit of 0 under iteration of the complex polynomial (i)

i) $z_{n+1} = z_n^2 + c$ (where c is a complex constant of the form $a + i \times b$)

remains bounded (Falconer, 2014: 245), never approaching infinity no matter how large *n* gets (see Mandelbrot, 1982; Falconer, 2014: 243, ff.).

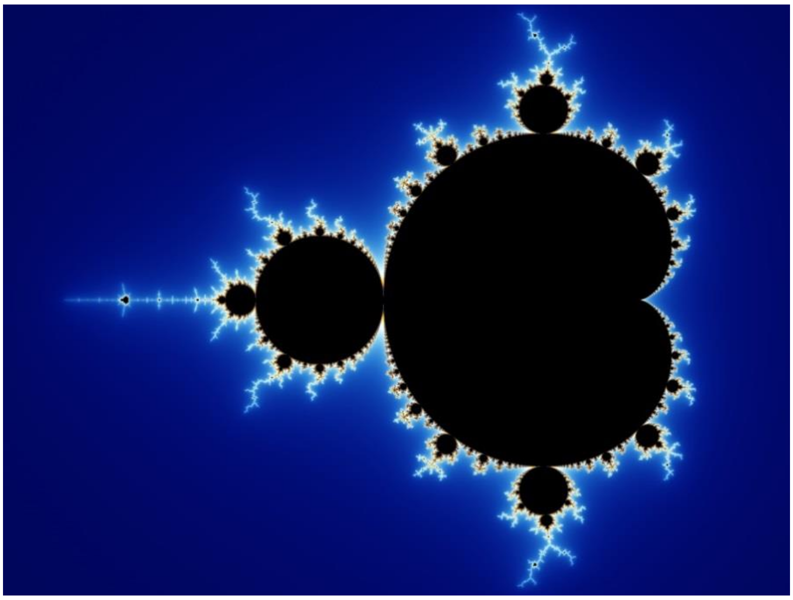
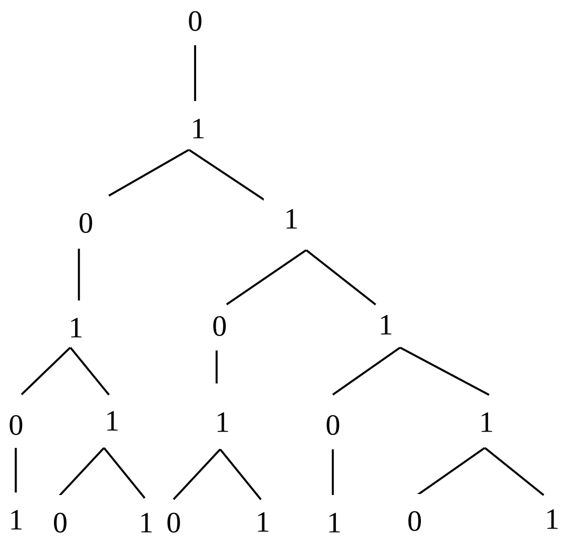


Figure 11: Mandelbrot set with a continuously coloured environment. [taken from http://commons.wikimedia.org/wiki/File:Mandel_zoom_00_mandelbrot_set.jpg]

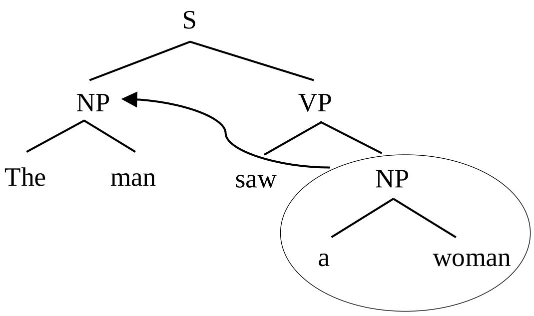
The graphical representation might be misleading, insofar as it might lead us to think we are in the presence of a *real* object (whereas in fact we are graphing *complex* numbers, with a nonzero imaginary part. What we are seeing is really a mix of the *imaginary –y* axis- and *real –x* axis-planes; see Falconer, 2014: 244). Depending on where we decide to zoom in, we will find varying degrees of self-similarity, but it is crucial to bear in mind that we are dealing with a mathematical structure, all of whose points are defined in the complex plane, and which does not interface with physical requirements. However, linguistic structures have *content*, which further complicates the picture: does self-similarity include specifications of, say, the *category* of terminals within an arbitrarily determined portion of the graph? Let us consider the Fib tree in (28) above, repeated here for the reader’s convenience:

(28’)



This graph will be useful for illustrating two properties of mathematical fractal constructs: on the one hand, a linguistic tree cannot just replace 1 and 0 by any symbol of its alphabet (say, for the sake of simplicity, #N#, #V#, #P#, within an eventive domain). Syntactic structure in natural language is not just a template that is filled with terminals and nonterminals, rather, we have proposed here and in previous works (Krivochen, 2015a, b) that syntactic structure ('phrase structure', in MGG terms) in natural language is neither uniform nor given *a priori* by axioms, but semantically-driven; and the transformations affecting terminals in terms of grouping (what we have referred to as *collapse* in our discussion about L-systems above, also called *fusion* in Distributed Morphology approaches, see Halle and Marantz, 1993) are conditioned by morpho-phonological possibilities of materialization: do not group what you are not able to materialize (McCawley, 1968, 1971). Strict self-similarity implies complete structure-preservation, which means any replacement of a terminal or nonterminal yields a legitimate object: this is clearly not the case for linguistic structures, which are subject to interpretation at both *sound* and *meaning* systems separately and targeting different minimal domains. We have dealt with the linguistic situation before, in the consideration of the phrase marker (30), repeated here as (30'):

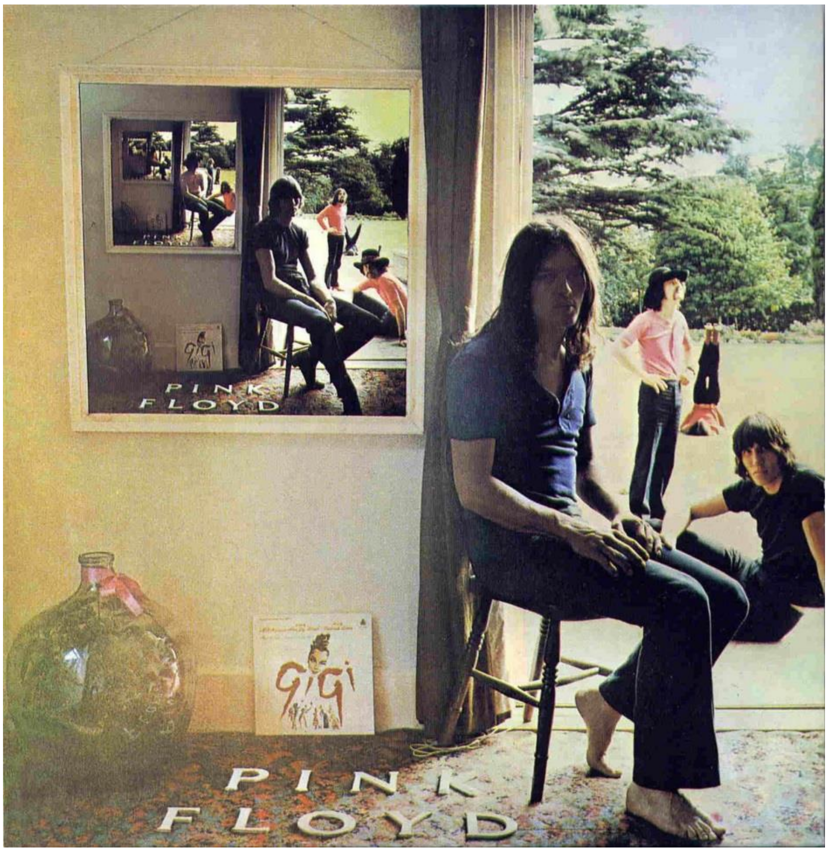
(30')



Needless to say, the result of such a mapping operation (moving the circled |NP| to a target position also specified categorially as |NP|, displaying both categorial and graph-theoretical identity) would be neither semantically nor phonologically equivalent to the input of the operation (in contrast to (28'), which yields the same interface result –the emergent Fib pattern- even after the mapping has been applied), the result being 'well-formed' (whatever that means within linguistics nowadays, particularly for frameworks that do not substantively dissociate syntax from semantics, as this one, or syntax from phonology, as that of Schmerling, in press), but only if the content of the terminals is not taken into consideration, which is not a possibility if we are working with natural language (it is worth noting that Emonds, 2007: 344; 2012: 32, fn. 7 justly critizices the weakening of *structure preservation* made in Chomsky, 1986a, which would allow for the substitution in (30') and many more unrestricted adjunctions).

So, while it is possible to say that, from the textual to the phonemic level there is structure (and there have been proposals that the structuring ultimately obeys the same principles, see for instance Van Dijk's 1992: 141, ff. textual 'superstructures', which were originally inspired by generative tree diagrams; Chomsky and Halle, 1968: Chapter 2 for the structure of morphophonological elements in terms of phrase structure grammars –like sentences-), we must take into account that such structuring does not respond *only* to formal, mathematical principles, but also to the possibilities (and demands) of interpretative systems, which, as we will see below, greatly influence computations in terms, for instance, of *halting* processes. Let us see some examples of content-sensitive self-similarity:

a)



b)

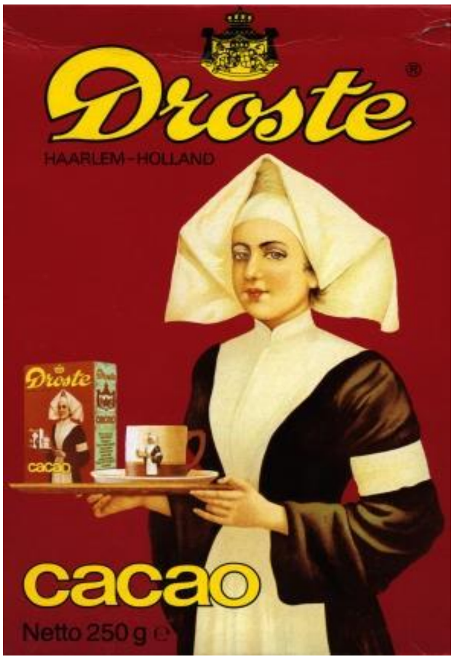
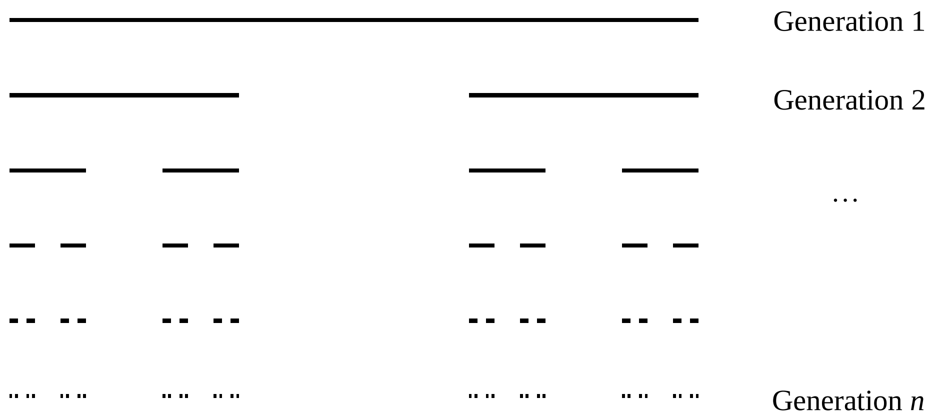


Figure (a) is the cover of Pink Floyd's (1969) record *Ummagumma* (LP version); figure (b) is the well-known box of Droste cocoa, 1904. There are interesting things to note in both images: (a) presents a recurrent *format*, meaning that the terminals and nonterminals involved in operations have the same structural relations among its elements (we have used this concept before, see Krivochen, 2011: 38-39), but the *content* of the terminals varies: considering only the person sitting in the chair in figure (a), in successive iterations we have David Gilmour, Roger Waters, Nick Mason, and Rick Wright. Self-similarity is, in this case, partial, or ‘statistical’ (using Falconer’s terms). Or is it? Let us consider the Droste cocoa box. Here, self-similarity appears both in *format* and *content* (since each generation of the structure gives us a reduced version of the whole), but there is a non-trivial matter: can we really say the nurses in the first and second iterations are ‘the same’ in any interpretatively relevant sense? The answer is not straightforward (and the question might appear to the reader to be more philosophical than mathematical), but contrast (a) and (b) with (c), a Cantor set:

c)



The Cantor set is generated by taking a segment of length *L*, and subtracting a central segment of length *L*/3 at each generation[115]. Zoom in on a segment by a factor 3 at generation 2, and you will get a segment that cannot be distinguished from that of generation 1. Our point is that, whatever notion of self-similarity we assume, we *do* want to be able to say that the kind of self-similarity involved in (a) is slightly different from that in (b), and they are both miles apart from (c). The crucial factor, we think, is that (a) and (b) are *interpreted*, whereas (c) is not. Structure preservation when mapping operations enter the game in natural language is not limited to *format*; since terminals are interpreted, *content* is also crucial. The degree to which structure preservation as a holds as a homomorphism condition over structure mappings is limited by the so-called ‘interface systems’, that is, by requirements of *interpretability*.

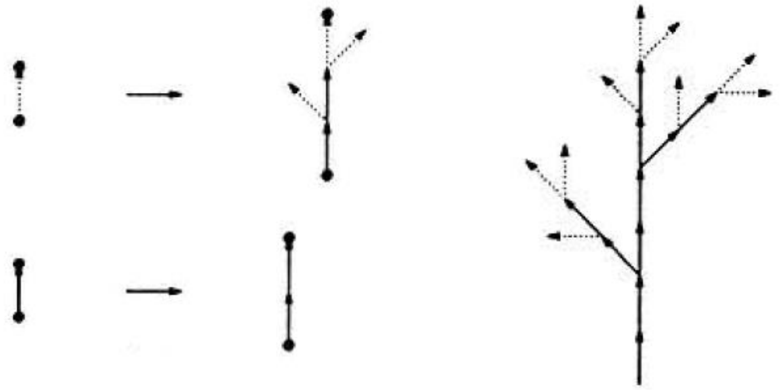
If the defining feature of fractals is their dimensionality (which is the most prominent feature of Mandelbrot’s 1983 definition), an interesting issue arises for phrase markers representing linguistic structures (which we have analyzed in Krivochen, 2018a). Consider the following grammar, an extension of classical L-systems (taken from Smith, 1984: 2):

105) Alphabet: {1, 0, [,]}
Axiom: 0
Rules: 0 → 1, [0,] 1, [0,] 0
1 → 1, 1
[→ [(opening boundary)
] →] (closing boundary)

[115] The Cantor set is a fine example of non-integer dimensionality: $\frac{\ln 2}{\ln 3} = log_3(2) \cong 0.6309$.

Smith (1984: 2) presents a graphical version of the rules, from a bottom-up perspective (i.e., from the initiator to an n^{th} generation):

d)



Production rules

Generation n = 2

Computer-generated graphics need a specification for branching angle: customarily, either 60° or 90° is used (but this is immaterial to the present discussion). In any case, the Hausdorff dimensionality of the tree is given by the formula $\log_2(2) = 1$, meaning the tree is unidimensional. Interestingly, as the number of allowed branches increases (which needs a change in the rules, sure), Hausdorff dimensionality increases as well, and for a uniformly 3-branching tree we have $\log_2(3) \approx 1.5849625$. For four branches, predictably, we have $\log_2(4) = 2$. This whole discussion might seem far-fetched, but it is not, crucially, for any theory which assumes that phrase structure graphs have any mental reality. This is the case for MGG if either Kayne's (1994) LCA or its theorematic version first presented in Uriagereka (1998: 202, ff.) is taken to be a neurocognitively real structure linearization algorithm (that is, if the LCA actually describes the way in which mental symbolic representations get dimensionally 'flattened' and linearized to be externalized as moving layers of air, as would be expected if MGG is a true 'biolinguistic' model); it must also be the case for the ultrametricity related proposals if they attempt to achieve implementational adequacy, which is an important part of Uriagereka's enterprise insofar as he derives many crucial syntactic conditions over local dependencies from deeper physical principles. There is another point to take into account (and we will come back to it below, in more depth and more explicitly): no matter how many iterations of the algorithm we carry out, *dimensionality is a constant for uniform tree fractals* (the number of branches being the determining factor to change Hausdorff dimensionality). However, we will argue the derivational system has two dependent variables: entropy and dimensionality. In Krivochen (2015a) we argued against a uniform template for phrase structure, this automatically yields a system with variable dimensionality, since Hausdorff dimensionality should be calculated *at each generation* to have a true approximation of the structure of the fractal. Here, we expand on that argument, setting our focus on the proposal made there (which, in turn, owes much to Uriagereka, 2002b), and will argue that the linearization of chunks of structure (so-called *cycles*) is motivated by considerations of dimensionality, which impose a limit over what can be derived 'in a single stroke' within a derivational space. Anticipating discussion in **Section 3.9**, which follows the proposal made in Krivochen (2015a), we will argue that predication relations increase the dimensionality of the syntactic manifold, until reaching a critical value which triggers the fading of a dimensional attractor, in turn forcing the system to 'squeeze' through the fading dimension, yielding an object of lower dimensionality. This will be argued from both a theoretical and a neurocognitive point of view.

Now, let us go back to the initial question: is language a fractal? Or, rather, is there any aspect of language which could motivate the use of fractal geometry to study its properties and / or its physical and neurocognitive substratum?

Mandelbrot (1983: 15) defines a fractal as '*a set for which the Hausdorff–Besicovitch dimension*[116] *strictly exceeds the topological dimension*'. Roughly, the topological dimension (a.k.a. Lebesgue covering dimension) of a space is defined by the number of intersections produced by the minimal amount of covering figures that fill the space (for instance, if we want to fully cover a curve with circles leaving no point uncovered, there will be overlaps between at most 2 circles)[117]. More generally, the topological dimension of $\mathbb{R}^n$ is n. Admittedly, Mandelbrot's definition omits space-filling curves (e.g., Peano, Hilbert…), but it is a good approximation, and is customarily used (though criticized) in technical texts on fractals (see the references in this section). An important feature of L-fractals (a fancy way of saying 'language fractals') is that *their Hausdorff-Besicovitch dimensionality is not constant*, for it increases with predication relations such that '*for n dimensions of α, f(α) = n+1*'; an idea we will come back to repeatedly. And this variability in Hausdorff dimensionality, if the proposal made in Krivochen (2015a) –expanded on here- is on the right track, is due to the fact that L-fractals are *interpreted*, and there are non-trivial limits determined by physical considerations. One might think that the impossibility of iterating the Droste pattern exactly as it is, is only due to the necessary finiteness of the material on which the pattern is drawn. But let us consider the Pink Floyd cover: at generation $n = 4$ we have run out of band members[118] (and, interestingly, the guys at *Hipgnosis* ended the iteration with a print of the cover of the band's first album, *A Saucerful of Secrets*[119]). What does this mean? Well, as will be obvious below (hopefully), there is a physical limit over dimensionality increase, which is not reducible to the finiteness of whatever physical workspace we operate on: in the Floyd case, we have exhausted the alphabet and the possible combinations. A further iteration would need to repeat a configuration, which would be redundant for an interpretative system. The boundary element *S* (the cover of *Saucerful*) delimits a cycle within the structure. The cover of *Ummagumma* is, then, *locally fractal*. Our model of syntactic relations based on $\mathbb{R}^n$ manifolds presents the very same kind of local fractal characteristics, with the additional feature that the Hausdorff dimensionality of the L-fractal is not a constant, but varies cyclically depending on predication relations (Krivochen, 2015a: 560, ff.) established at each derivational step (which we will *artificially* quantize for purposes of having some concrete values for t to work with in our differential equations), thus adopting a bottom-up perspective for methodological purposes. This perspective follows from a

116 Or just 'Hausdorff dimension', see above.

117 Another take on 'topological dimension' is Poincaré's, according to which a space or an object has dimension n if it can be parametrized by another, of dimension n-1. Thus, for instance, a cube has dimension 3 because it can be divided in two sections (called *ana* and *kata*) by a plane, whose dimension is 2. For our purposes, related to the varying dimensionality of L-fractals within a derivation, the definitions of topological dimension are roughly equivalent.

118 We are not counting Syd Barrett, who was no longer a member of the band at this point, or Snowy White, who joined the band for the Animals (1977) and The Wall (1980-81) tours. We are working with the alphabet of band members available at t = 1969.

119 In the CD version, the cover of *Saucerful* is replaced by an iteration of the first generation reduced by a factor 8. This yields a strict self-similarity, but its mathematical infinity does not illustrate our point, so we have chosen the LP cover version instead.

‘derivational diachrony’ (i.e., it does not overlook the time dimension in derivations, along the lines of the development of L-grammars, which are biologically oriented), which is better suited for our purposes and our tools (which include differential equations) than looking at the final manifold and trying to analyze the formal properties of the object and the operations which generated it. This latter alternative can make us lose sight of interesting considerations of varying dimensionality and interpretability which arise from the online dynamics of the computation.

For the sake of closure, let us go back to the characteristics of fractals in (104)[120]: which of those apply to L-fractals? The distinction between local and global processes is essential here, since the notion of *cycle* is not accidental in language but, if we are anywhere close to correct, derived from the physical constraints that computations are subject to. So, we do not find structure at arbitrarily small scales, nor can we extend the manifold to infinity, because the system is not only embodied, but in fact *an emergent property of interactions between physically bounded elements* (just like trees, rivers, leaves, and other natural objects commonly regarded as fractals). Within the relevant scale, however, we do find iterative processes, guided by the interpretative systems. The Hausdorff dimensionality of the L-fractal is also consistent with the mathematical framework, displaying both integer and non-integer dimensionality in a cyclic fashion (more on this below), until reaching a critical value which triggers the fading of a dimensional attractor. Self-similarity is to be relativized to *format*, not *content*, which is in line with our previous considerations in **Section 1.6** about the non-applicability of a ‘digital’ version of structure preservation to operations on these objects (*contra* Gallistel and King, 2010), which is particularly relevant for a theory of structure mapping (which we have outlined in Krivochen, 2015b, and will expand on here in **Part IV**, focusing on the notion of ‘island’ and their cognitive reality). If the object we are dealing with were structurally uniform (as is axiomatic in MGG), then Mandelbrot’s (1983: 153) concept of ‘*subfractal*’ would apply, meaning the fractal dimension of the object would be uniformly equal to 1, and thus the tree would *not really be a fractal* (see also Smith, 1984: 2). But we have seen dimensionality is not a constant, nor is it the number of branches (i.e., the number of either terminals or nonterminals that depend structurally on a nonterminal), a number that depends on interface considerations (see the discussion of iteration and coordination in Krivochen, 2015a, which implies multiple branching structures depending on semantic relations between the terms involved in the relevant structure building process). The notion of ‘*non uniform fractal*’ (also due to Mandelbrot, 1983: 154) is better suited to describe the kind of objects to which L-fractals belong, *objects displaying fractal properties within a dynamically bounded space*. Let us see an example, limiting ourselves to MGG-like tree structures for the time being. Consider the derivation of (106) –admittedly, a stupid example, but we are interested in illustrating form rather than substance-:

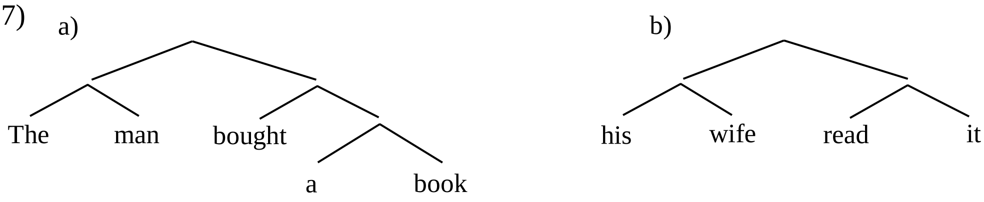
106) The man bought a book and his wife read it.

[120] Which we repeat here for ease of reference:

a. They display structure at arbitrarily small scales
b. They cannot be described by ‘traditional geometric language’ (Falconer, 2014), either locally or globally
c. They display self-similarity (either strictly geometrical / topological, or statistical)
d. Their dimensionality is often a non-integer
e. They are not defined in Euclidean metric spaces, such that $d(x, y)$ can range from 0 to ∞
f. They are almost always obtained via iterative procedures

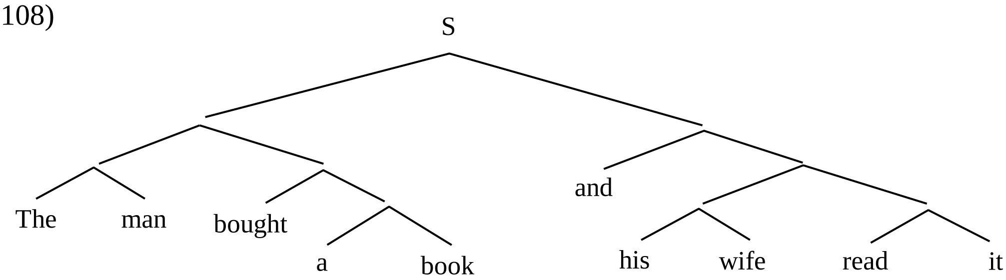
We will come back to the structure of coordination in **Section 4.4** below, so for the time being it will just have to be assumed that at some point we are dealing structures like (107):

107)



Each of the structures in (107) has a (Hausdorff) dimensionality of 1, considered as trees: $\log_2(2) = 1$. We will see that things are more complicated because we are in fact dealing with manifolds that only locally display dependencies as simple as those diagrammed in (107). However, consider now the coordination of (a) and (b), which we diagram in (108):

108)



If this monotonically binary branching structure for coordination is assumed, *and* combined with the idea that tree structures have some topological reality, then dimensionality stays the same, for the number of branches does not change. Notice that there is a hypotactic relation between the conjuncts, such that elements of the second conjunct refer back to members of the first one ([his] and [it]), moreover, the events are interpreted as if ordered in time (even though there is no overt indication of such order), such that *e*(buy) < *e*(read), and this order cannot be tampered with without changing the meaning of the sentence and even its truth value[121]. These might lead us to adopt a *strictly* hierarchical representation with *uniform* binary branching for coordination, which we have referred to as *monotonic Merge* (Uriagereka, 2002a). The relation between conjuncts is antisymmetric, for (106') is ungrammatical:

106') *His wife read it and the man bought a book

However, it would be a mistake to assume *all* coordinations share the same structure. Symmetric coordinations (see Schmerling, 1975 for discussion) seem to have a different, 'flat'

[121] In terms we will introduce in **Section 3.9**, elements that co-refer (which can be of arbitrary complexity) are defined by the same state vector, and literally point to the same attractor within a cognitive phase space. Topologically, it is illustrative to think that the phrase marker *warps* (in the sense of Uriagereka, 2002b; also Martin and Uriagereka, 2014) so that the nonterminal node immediately dominating [a book] intersects with the second conjunct and replaces the terminal [it] for semantic purposes. Note that such a proposal has consequences for the interpretation of Right Node Rising examples like [The man bought, and his wife read, a book about…], when the NP is sufficiently 'heavy', if we actually have *the same vector* in two structural positions. The interpretation of such vector, thus, depends not only on the specific components of the vector but also on the structural context in which it appears. These considerations have at least two far-reaching consequences: (a) the computations that define linear ordering and materialization of terminals are radically different from those involved in semantic interpretation, and (b) that by warping the phrase marker we increase the dimensionality of the manifold from 2 to 3.

structure (which we will thoroughly analyze in **Section 4.4** below; see also Krivochen and Schmerling, 2016). Consider (109):

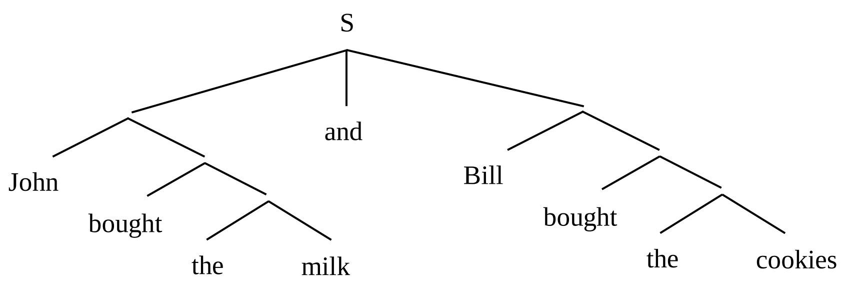
109) (talking about afternoon tea) John bought the milk and Bill brought some cookies

Note that (109') is also perfectly acceptable (since there is no particular order in which the purchase of milk and cookies should be presented, and there is no cross reference within the sentence):

109') Bill bought the cookies and John bought the milk

This possibility of reordering (among other things we will not get into here) seems to point towards a different structure, more like (110) (see Culicover and Jackendoff, 2005; Krivochen, 2015a for details and discussion):

110)

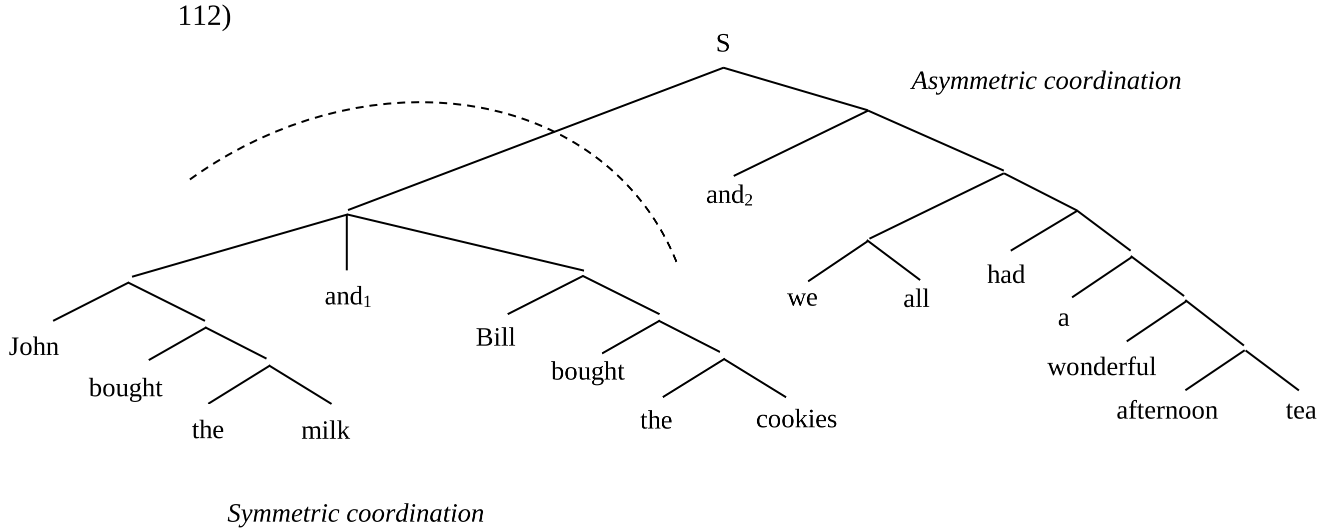


Note that there is no dominance or asymmetric c-command relation between the conjuncts, they are in a perfect paratactic relation which correctly predicts (109') to be grammatical. Also, consider that a phrase marker like (108) for (109) would assign 'too much structure' (Lasnik, 2011) to the string, and incorrectly predict syntactic and semantic asymmetries between the terms of the conjunction (Schmerling, 1975). The structure to choose is then a 'flat' phrase marker, whose terms are in turn monotonically assembled sub-trees. But now the (Hausdorff) dimensionality of the tree has changed: $\log_2(3) \approx 1.5849625$, in a single derivational step (when the two terms of the coordination are conjoined): in derivational time, the dimensionality of the L-fractal we are working with *changes dynamically* depending on the structural relations that are established among its constituent parts. But let's go one step further, and consider (110) embedded in a bigger structure:

111) Bill bought the cookies and John bought the milk, and we all had a wonderful afternoon tea

The terms of coordination are two once again, and the relation seems to be asymmetric: we had a wonderful afternoon tea *after* (and probably also *because*) John and Bill contributed with milk and cookies respectively. How can we represent that? Well, (112) is an option:

112)



If the first term of the 'big' coordination [and$_2$] is taken as a unit for the purposes of further computations after the cycle is closed (which would be the expected situation under strongly cyclic considerations, to which we will return below particularly in **4.5**), then when we assemble the phrase marker in (112) we are back at dimensionality 1, because the essential skeleton of (112) is [A [and$_2$ [B]]] (with A there being the symmetric coordination as a whole). If our proposal in Krivochen (2015a) that phrase markers display this kind of variability in the number of branches, attending to semantic considerations (e.g., the necessity of representing scope relations between elements, the possibility of having ATB-like extraction phenomena and relating operators and variables, etc.), then *the dimensionality of an L-fractal is not static*, and while being essentially a representational characteristic of tree representations (which does not apply to the 'Klein bottle' phrase markers we will argue in favor of), we see that we have to specify the derivational point at which we will perform the relevant measurement, and, if possible, also make the derivational history of the syntactic object to be analyzed fully explicit. In our conception of syntactic computations, 'mixed' phrase markers like (112), displaying both finite-state (paratactic [and$_1$]) *and* phrase structural dependencies (hypotactic [and$_2$]) are not an exception, but rather the norm. Adjunction, iteration, coordination…all those phenomena (and some others) require, in our perspective, more dynamical (and interactive) take on syntax than has been the norm in MGG. A mixed computation perspective can effectively address the limitations of phrase structure grammars noted by Chomsky (1963), Ross (1967), Lasnik (2011), and Lasnik & Uriagereka (2012) (among others) without the need to resort to transformations; at the same time, this perspective can bring together insights from dynamical systems theory and theoretical linguistics in a natural way (Krivochen, 2016b).

With respect to metric considerations over L-fractals, we have made a case in this very work for the ultrametricity of the conceptual and phonotactic fields, but we will see in more detail how the presence of relational elements in language disrupts that ultrametricity, imposing different distance functions to different *n*-grams of elements for the sake of interpretation. L-fractals (*qua* emergent products of a dynamic oscillatory computational engine) are thus objects whose properties cannot be fixed beforehand as they are subject to change as the derivation unfolds, and their nonlinear, dynamical properties make them fascinating objects from the perspectives of both physical / mathematical modeling, and neurocognitive explanation.

3.4 Back to fieldwork[122]

While field theory is essentially probabilistic (that is, a field is expressible as a set of probabilities associated with an *n*-tuple of coordinates and a relevant magnitude or set thereof), a leap of faith would be required, we think, in order to concede that the interactions between elements in the linguistic field(s) are themselves probabilistic: after all, elements to be derivationally manipulated are not randomly chosen; if they were, we should have a go at each utterance several times (more than would be computationally tractable[123]) before hitting the right formulation. The field might be probabilistic in nature (even more so under quantum field theory assumptions), but this does not mean that we have probabilities all the way down (or, rather, all the way up, from the field to the derivation). In fact, if Schöner (2009) is on the right track, we would expect the system to be quite a bit more ordered than a purely statistically based system would predict, and this is so because order does not emerge from an external controller or the manipulation of variables, but from the dynamics of the system itself, which self-regulates around attractors / repellors. Thus, changing the dynamics of the system means moving attractors, relatively stable sets of points within the phase space. The relations between the system as a whole and these attractors is not uncontroversial, however, and there are, at least, two important proposals in relation to this problem (see Spencer, Perone and Johnson, 2009: 109):

a) The system goes *near* stable attractor states, but not *into* them since the continuous nature of the change of the system over time won't allow it to be in a stable state for 'any significant amount of time' (Spivey, 2007: 26). The continuous input the system receives prevents it from 'falling' into a stable attractor.

b) The system goes *into* stable attractors, having enough flexibility to 'escape' the stable attractor and change the dynamics. This proposal requires '*an active mechanism that can destabilize this state when the situation demands*' in order to make the system go after achieving local stability (Spencer, Perone and Johnson, 2009: 109)

Option (b) seems to require an *extra* mechanism (in order to destabilize the system), and this mechanism must in turn be aware of external 'situational demands' that require the system to be destabilized so that the dynamics can further evolve, as it is not obvious how it would be built into the system itself. Moreover, the picture Spencer et al. propose does rest on some unclear stipulations over simulated resting states for neural fields, which apply to applications in robotics (Dineva, Schöner, and Thelen, 2008), but whose translation to human neurocognitive terms is far from straightforward. Our own proposal is closer to Spivey's, which mathematically requires us to consider the relevant attractors as asymptotes, making the notion of *limit* (of the relevant function we are considering) crucial for the formalization of the system, as we will see explicitly below. A derivation (as a process), so to say, makes the system 'zig-zag' between attractors and repellors, without falling into attractors and never staying in a state for too long (which relates to Spivey's 2007 argument against an inherently static notion of 'mental state').

[122] Get it? Because we're dealing with *fields*…oh, nevermind.

[123] There is a way of posing this problem in terms of the P vs. NP problem: after all, the 'grammaticality' problem is easily checkable (when presented with a sentence, any speaker of a language L can determine whether it belongs to L or not fairly quickly), but, as we see, it is not obvious that it is easily solvable. We will not pursue this line of reasoning here because it would be tangential to our point, at best, but it sure is an interesting way to think about language and its implementation in an automaton.

This is because inputs (from the system itself as well as from the exterior; Schöner, 2009: 33) are permanently influencing the processes taking place within a workspace.

The interpretation assigned to a particular phrase marker, which is the result of particular perturbations to the ground state of the lexical field and the establishment of dependencies between these perturbations by means of interference (Uriagereka, 2011), depends on the dynamics of the system: Fib patterns (along with XOR, Lucas, or whichever patterns the reader might think of) are also emergent properties of the combination of specific *restrictions* over structure generation (e.g., 'have a weak and a strong term in your grammar', in the case of the Fib system) and over the alphabet the system can manipulate (e.g., {1, 0}, {*a*, …, *z*}, as in the case of all automata we have reviewed, only some of which incorporate extra elements with structural significance, like ¢ and $ in an LBA). What does this mean? It means, as we see it, that we can flip the orthodox views on language 'evolution' (from a computational perspective) upside down: instead of adding layers of computational complexity until reaching the level of sophistication required to generate all and only the sentences of natural languages NLs (as has been common practice in so-called 'biolinguistic' studies, more often than not, those intimately associated to Chomskyan generative grammar; Bickerton, 2007, e.g.), we start with an unrestricted non-linear generative system, and add conditions over its emergent products via other systems interfacing with the generative system, like sound and meaning, each of which is a field of its own, with specific topological properties and requirements over allowable inputs. The resulting product, the optimal resolution between the dynamical frustration thereby created in a particular context of externalization, is what we call a 'sentence in a natural language' (more on this from a Zipfian perspective, below).

Uriagereka (2011: 23) takes things slightly differently, in a very intriguing way. For starters, he assumes the following condition holding for structure building:

> <u>*Reduction of Entropy at Merge*</u> *(REM): Two fields* ψ_α *and* ψ_β *combine via Merge if and only if* $S\{\psi_\alpha, \psi_\beta\} < [S\{\psi_\alpha\} + S\{\psi_\beta\}]$

Note that the condition (which relates the entropy *S* corresponding to two field perturbations that interfere via Merge) includes a biconditional, meaning that there is *no* instance of Merge that does not reduce entropy. While this is in principle desirable (and, in fact, can be seen as the interface-defined side of the Extension Condition + Full Interpretation Principle coin that shapes derivations in MGG –at least in theory–), it sure has to be relativized taking into account interface requirements at different derivational points, and also contemplating the possibility that $S\{\psi_\alpha, \psi_\beta\} = [S\{\psi_\alpha\} + S\{\psi_\beta\}]$ at least for a 'derivational turn' (see, e.g., Putnam, 2010: 6 on the concepts of 'hard' and 'soft' crash[124]), which we will do below. For the time being, we assume that the REM condition applies by cycles, in consonance with the MSO model first developed by Uriagereka. The point of a linguistic derivation is, we agree with Uriagereka, to reduce entropy, but this reduction is not, in our opinion, monotonic or uniform (an idea we proposed in Krivochen, 2014c), and does not occur in an autonomous syntactic component, but *within* the sound and meaning fields, which configure the phase space for the relevant linguistic computations once they limit the unrestricted possibilities of an L-system instantiating an

[124] Michael T. Putnam (p. c.) comments in this respect that "*My feeling now is that the only "hard crash" that essentially must be avoided is the one at the end of the derivation as a well-formedness constraint as such*." Interestingly, this implies that measures of optimization *during* the derivational process are 'soft', thus allowing the system to hold a momentarily non-convergent piece of structure in the episodic buffer to see if a local operation can introduce an element to solve the ill-formedness of that piece of structure.

ultrametric space of the kind we have discussed above. Moreover, in our model (see also Saddy, 2018) entropy reduction takes place *locally* as a by-product of metrization and dimensionality increase, by cycles. The emergence of cycles, which are the *locus* of entropy reduction, is intimately linked to the strong adequacy condition over grammars: cycles are defined as local units with uniform computational dependencies. In an interactive system that permits mixed computation, local domains are closed when we need to jump up or down the Chomsky Hierarchy; it is within a computationally uniform local derivational unit that entropy reduction takes place. After a cycle has been completed, entropy goes back up.

In Uriagereka's model, the REM depends on the combining fields' being off phase when they combine via Merge (2011: 24):

> Destructive Interference [DI]*: Only if two fields ψ_α and ψ_β are out of synch when they combine will REM conditions be achieved.*

We find the DI condition too strong as originally formulated, insofar as constructive interference (i.e., field reinforcing) can explain some psycholinguistic effects involving asymmetries between facilitatory effects on the one hand, and neutral and impedimentary contexts on the other, as we will see below. It is also not explained why, if a N and a V are to combine, their respective fields must be off phase, other than by resorting to REM and DI. The REM is also responsible for combining two kinds of lexical fields within the lexicon, which is overall understood as a vector field (Uriagereka, 2011: 56): lexical and functional/grammatical fields. It is to be stressed that the idea that '*concrete words are in fact designated (arbitrarily paired) stability points in that network* [of vectors]' can be formulated and developed independently of the REM requirement, despite Uriagereka's claim that REM is the 'deeper principle' from which combinatorial properties between lexical and functional elements follow (for instance, by making a partitioning mechanism explicit for both *form* and *meaning*). So far, it is not even clear that we are dealing with *vector* fields, as it is possible that at least part of the relevant computations, particularly those related to *semantics*, involve *tensors* of an order yet to be determined. We will indeed work with vector fields here, but we need to stress that the nature of the relevant fields is ultimately an empirical issue, and further research might prove us wrong in this choice.

It must be emphasized that, since Uriagereka's (2011) was the first approach to linguistic derivations in terms of fields combining in a non-Euclidean space, we do think it is a major contribution to the study of the physics of language (see also Saddy, 2018 for a more recent perspective). What we will attempt to do is combine Uriagereka's insight with Schöner's non-linear, dynamical systems approach to cognitive processes, taking things beyond the level of metaphor. It will be useful to present a different, yet related approach to interference in cognition, which is based not on *continuous* fields and wave interference (as Uriagereka's, Saddy's, and ours is), but on the competition among *discrete* elements triggering an output representation. After presenting the feature interference model as an example of the role of interference in traditional cognitive science in **Section 3.5.1**, we will introduce our own modified version of Uriagereka's *field* account in **Section 3.5.2**.

3.5 When waves collide: on interference

3.5.1 Interference in (traditional) cognitive science

The concept of interference is a common one in inquiries about the nature of working memory (WM) and the representations it manipulates. Items held in WM are defined as sets of features, which interact while active. Interference occurs

> *because different items are represented as different patterns of activation across the same set of features, their representations can interact and thereby degrade each other* (Oberauer and Kliegl, 2006: 606).

The characterization of elements in WM or conceptual structures as bundles of features is a commonplace in the literature: Taylor et al. (2007), Taylor et al. (2011), Caramazza et al. (1990), to mention but a few, assume concepts have an internal structure, which is expressed by means of features like ‘has eyes’ or ‘has a nose’ (Taylor et al., 2007). Those features are not exclusive to a particular entity, but are shared among representations. Thus, in these accounts, a particular entity can be defined as a specific (unordered) set of features, specificity resting on the particular combination of features composing the relevant representation, rather than on the features themselves. Since features are shared, the activation of a feature is not a straightforward, clean issue: if feature bundles are represented as patterns of activation over partially overlapping sets, interference is almost inevitable. The dynamics of the interference model allow for competition, in the form of representations competing for a feature which is associated exclusively with the winning representation (by means of a process that Oberauer and Kliegl do not formalize), such that for *n* elements in the WM, each element undergoes the interference of *n*-1 elements (Oberauer and Kliegl, 2006: 607), in a non-stochastic process. The process is graphed as follows (numbers within the network indicate activated features at each node; note that the added results equal the input):

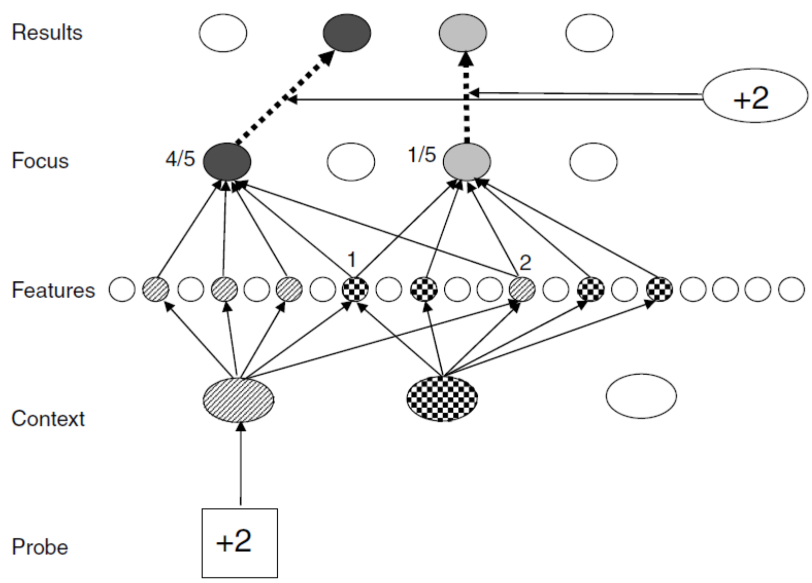


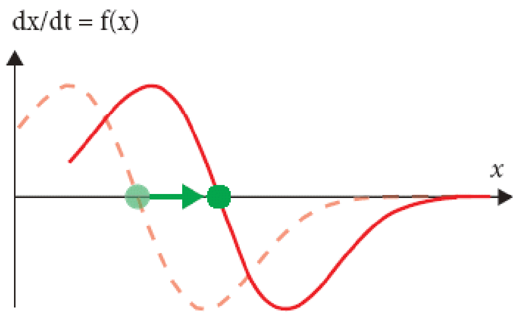
Figure 12: The Interference Model (taken from Oberauer and Kliegl, 2006: 608)

The model consists of four layers of units, one of which (the feature layer) is assumed to have sparse distributed representations. The other layers are sketched as localist representations, but this is not meant to rule out that they use distributed representations as well. The items currently held in WM are represented in the feature layer as sets of active feature units. Feature units belonging to the same item are bound together by sharing a common phase of firing, doing so in synchrony. They also fire in synchrony with a context unit in the context layer, which binds each item to a representation of this item's location. However, not only is the nature of the competition process is left obscure, the very nature of the competing features is also unclear. From a theoretical point of view, the discreteness associated with features is not clearly justified: what exactly is 'a feature', and how is it acquired / manipulated?

It must be stressed that we are not arguing against the concept of interference itself, but the particular instantiation of the concept in a feature-based model (and, more in general, discrete units manipulated algorithmically). After all, if interference is due to simultaneous firing, the firing pattern can be described in field terms as well: synapses are electrochemical processes, and the brain can be mapped in terms of the probability of finding electrical (electromagnetic) activity at any point for a particular time -bear in mind that neurotransmitters are ions, thus, electrically charged-. It is relevant to point out that Xiang et al. (2002) have found via magnetoencephalography (MEG) that the presentation of stimuli consisting of moving sound sources and static sound sources (thus, varying with respect to the Doppler effect) evoke magnetic fields, mapping ingoing and outgoing magnetic fields in the superior temporal and right parietal cortexes (Xiang et al. 2002: 5-7). Since electric and magnetic fields are in phase in EMR waves (they reach minima and maxima together), we can conclude that there is electrical activity as well, although in an orthogonal plane (electric fields and magnetic fields are mutually orthogonal). By mapping neural firing patterns via noninvasive brain imaging techniques, we can have access to empirical data that is crucial for the falsifability of the field-theoretic model, or at least of its secondary predictions. Indirectly, we can gather neurocognitive evidence in favor of the 'syntax-as-a-topological-operation-over-fields' theory, apart from its conceptual appeal, possibilities of interaction with other alternative formal frameworks (Harmonic Grammar, Survive-Minimalism, among others), and essentially interdisciplinary character.

Schöner (2009) claims that neurons and neural networks are naturally described as dynamical systems (cf. Kornai's, 1985 *finite state* approach), with a point or set of points as attractors. The stability required to have a cognitive representation as an emergent property of neural activity is guaranteed by the attractor, towards which the function *tends*. Crucially, and we will come back to this, stability is something towards which the system tends, but we will argue against conceptions like Spencer et al.'s (2009), which assume (mental) states are the result of the system *going into* stable attractors. Rather than stability, then –and particularly within a *dynamical frustration* approach-, we talk about *dynamical instability*, bound by physical constraints over the neurocognitive substratum of the system.

Of course, the system is dynamic enough to allow for attractor points to change, given external perturbations (which are inevitable if we are dealing with an open system). The situation of change over time -thus the differential equation- is graphed as follows (from Schöner, 2009: 5, Fig. 2.2):



The system is, furthermore, capable of including repellors, which are points that neutralize attractors, and may even ‘collide’ with them making them disappear. Needless to say, the kinds of interactions allowed by dynamical nonlinear systems like Schöner’s are quite a bit richer than the feature-based model of Oberauer and Kliegl. Consider, for instance, what interference would look like under dynamical assumptions: each possible state of the system is mapped as a function; say, each activated representation (of arbitrary complexity: it does not matter for present purposes whether the candidates are lexical entries or representations of the logical form of a sentence) is mapped as a different dynamic system, with its own subroutines and its own attractors (if the representations are complex enough, even separate workspaces, see Uriagereka, 2002a; Krivochen, 2015a for discussion and details of multiple workspaces in syntactic derivations). However, if there is competition between multiple output candidates, they will interact at some point during the derivation. How to represent such interaction? The concept of *phase* will be useful here.

Phase is the difference in degrees or time between two waves at a particular point in time. Two or more functions are said to be *in phase* if they share frequency and phase, or if their phases are related by multiplication by a positive integer. When *n* functions are in phase, their interference is said to be *constructive*, as peaks meet peaks and troughs meet troughs thus adding their relative intensity (peak amplitude), in the following way:

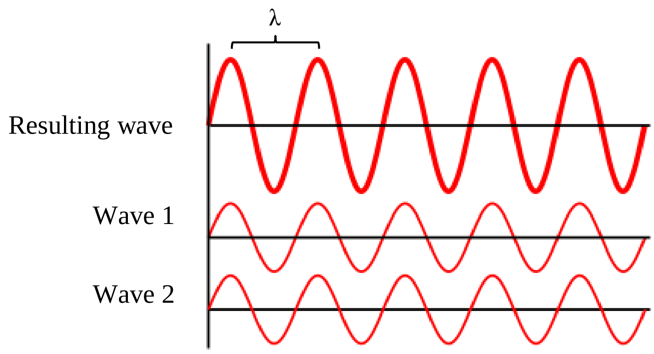
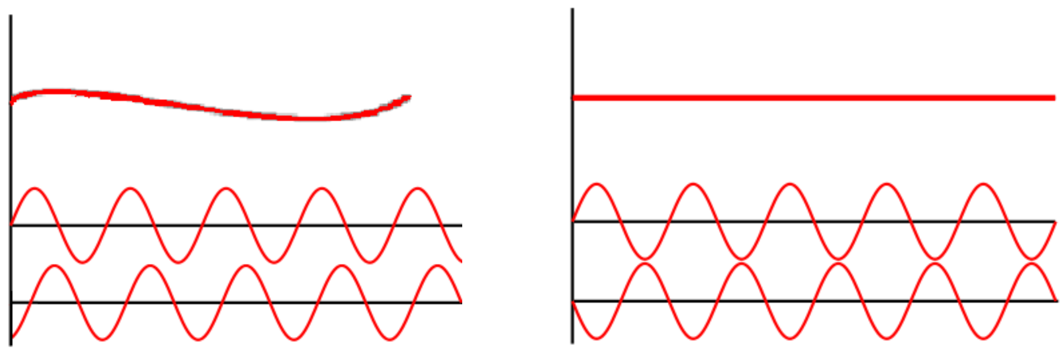


Figure 13: Constructive Interference

If the relevant functions differ in frequency and / or degrees at a particular point in time, such that peaks do not meet peaks exactly, the waves are said to be *out of phase* or *off phase*. The corresponding graphical representation (purely illustrative, needless to say) is as follows:



Partial destructive interference

Total destructive interference

Figure 14: Destructive interference patterns

Uriagereka (2011: 25) claims that only total destructive interference can lead to '*definite propositions*' and rigid designators, by virtue of nominal fields being *completely out of phase* with respect to verbal fields. He assumes that the only way to diminish entropy is via destructive interference between fields of different natures. The flat state would be the result of a series of interactions; presumably, every linguistic derivation would terminate with a flat wave (i.e., total destructive interference). We will now present a different, contrasting scenario.

3.5.2 Interference revisited (featuring Entropy and Cycles)

Consider now an alternative scenario, in which the dynamics are slightly different (but we still assume a classical generative alphabet, for the sake of concreteness). For starters, the example taken by Uriagereka (an ergative construction, 'things happen') does not include any procedural element (i.e., no prepositions P, tense T, causative light verb *v*, or complementizers C). This is crucial, because, if procedural elements convey instructions on how to relate conceptual elements (which are defined as the perturbations over the lexical field), no procedural elements means no relational instructions. In such a scenario, the interaction between constituents would be direct, without any mediation via procedural elements. However, the result would be a severely underspecified semantic representation, in which there would be no way to encode figure-ground dynamics (which arise in the presence of P), or causativity (which arises in the presence of *v*), nor could we have sortal or eventive entities (which depend on the presence of determiners D, and tense T instructions respectively). The next question is, do procedural elements influence the wave interaction, given the ultrametric conception of the lexicon illustrated in (100 b) above? We believe they do, moreover, we argue that more procedural elements in the construal mean more interference, and the disruption of the normal ultrametric topology of the lexical field (following the insight of Uriagereka, p.c. with respect to how syntactic relations 'bring elements together'). As a *metaphorical* example of this scenario, consider the well-known double-slit experiment: coherent, monochromatic light with wavelength λ from a single source (or multiple sources, see Pfleegor and Mandel, 1967, but let's keep it simple for the time being) passes through two slits of roughly λ opening, and travels a distance *D* towards a photosensitive screen. Since the opening of the slits is roughly λ, the light waves refract, and since we have more than a single slit, we end up with two sources of refracted light emitting waves with a constant wavelength λ in all directions forwards, travelling *D*. Let us represent this as follows:

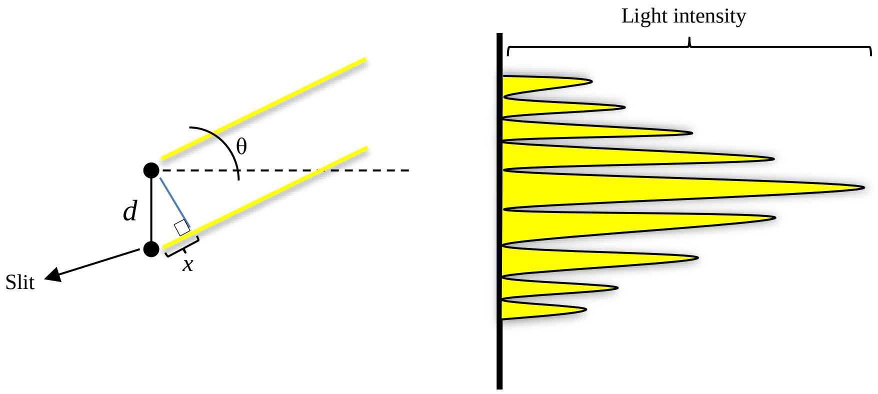


Figure 15: Double Slit Experiment

Distance $x = d(\sin\theta)$, and if $x = n\lambda$ (where n is any positive integer), then the waves will be in phase after distance x, and the result will be constructive interference -i.e., there will be a bright fringe in the photosensitive screen-. If, on the other hand, $x = (2n\text{-}1)\frac{\lambda}{2}$ (i.e., an odd number of half-wavelengths), the waves will be completely out of phase, and complete destructive interference will be the result -in this case, a dark fringe in the photosensitive screen-. An interesting condition emerging from the double slit experiment is that, without slits, or with just a single slit, there is *no interference*. The same thing happens if we have two or more slits whose aperture is far greater than λ, or far smaller: in the former case, the wavefront just passes by the slit, without any refraction effect; in the latter, the wavefront is *reflected*, and no wave goes through. The condition is clear: the slit opening must be ≈ λ. But this is not all: the more slits (assuming they are all at distance *d* from each other and are all of width λ), the more destructive interference, since we have more 'sources' of light from every direction towards the photosensitive screen. The bright fringes will be much brighter as we increase the number of slits, but there will be a greater dark space between them, and, as we increase the number of slits, the number of bright fringes decreases (but they become brighter).

How does this impact on Uriagereka's conditions on Merge? If we have only lexical categories involved in the process, there will be no interference between the respective wavefunctions at all, because there are no slits (in topological terms, if there is nothing to trigger the disruption of the ultrametricity of the lexical field, interference is of course impossible since the distance function between any two perturbations of the field is constant). '*Wait a minute*', the skeptical reader might say. '*If two rocks are thrown in a pond, they will create circular wavefronts, which interfere*'. True, but although that is a good example of wave interference, it does not involve fields. Moreover, it is dependent on the medium and its properties (e.g., density): if we throw two rocks at roughly the same velocity through air (a less dense medium), no such effect occurs. What we claim here is that even though it makes theoretical and cognitive sense to describe lexical categories in terms of perturbations of a field (ultimately, an electromagnetic field, if the proposal is to have any neurocognitive plausibility). If the ground dynamics of cognitive computation is an ultrametric field, categories do not interfere *by themselves* (in the absence of perturbations, which impose a metric over that field), and stipulating a *Merge* function within a particular theory of structure building as recursive combinatorics does not make things any better if we want to know *what* happens and *why*.

Schöner (2009: 15) claims that '*the fields themselves are assumed to form dynamical systems, consistent with the physiology and physics of the corresponding neural networks*', which do not need to be localized in the traditional, static sense: after all, emergent properties depend on an interaction between actants in a system, and for as long as that interaction is *local*, defined in terms of accessibility (i.e., α can interact with β if α can directly or indirectly act upon β or *vice versa*)[125], for us to limit interactions when they are possible given an external trigger needs special stipulations to be added. The dynamics of the relevant interactions are non-linear, and the interference effects depend on contributions to the field dynamics that are independent of the current state of the field at any given moment, what Schöner refers to as 'inputs' to the field. *Relational elements* (in the sense of Mateu Fontanals, 2002; 2005), a.k.a. *functional categories* (Abney, 1987) or *procedural categories* (Sperber and Wilson, 1995; Escandell and Leonetti, 2000, 2004) *are the 'slits' that make waves interfere in specific ways*. Let us give an example: the wavefunctions of the elements [book] and [table] can be described independently as different excitations of an ultrametric lexical field, but there is no reason to make them interfere, as they are not linked in any way, and the distance function between them is the same as that between any other elements. Now, in an expression like [the book is on the table], there is a figure-ground dynamics, [book [on [table]]], describable in terms of spatial *coincidence* between the entities involved (Hale, 1986). The figure and the ground are certainly related; in fact, *figure* and *ground* are themselves relational concepts (rather, 'categories', as briefly mentioned in **Section 2.10** above), defined in terms of one another. In the case of figure-ground dynamics, it is the preposition that generates interference between the relevant wavefunctions, independently of the properties of the generative operation we assume (be it Merge, Concatenate, Unify, etc.). From this argument comes the idea that *'syntax' as a topological operation* (or set thereof) *over field perturbations is intimately linked to the existence of relational categories*, be they narrowly linguistic or more generally cognitive relational primitives (primarily locative, following the theory of Anderson, 1977, see also Talmy, 2000). This line of reasoning can, in principle, provide a provisional answer to fundamental questions about argument structure and syntactic-semantic construal (a crucial topic we will return to below, when dealing with questions of dimensionality). One such question, perhaps the most pervasive, has to do with the number of participants involved in a construal: typologically, there are no known languages in which a construction has more than three arguments, corresponding to External causator ('effector', in Van Valin's terms, see Krivochen, 2015c for discussion), Theme, and Location –other constituents being considered 'adjuncts', peripheral elements in the syntactic-semantic construal-. A possible answer to the question 'why is this so?' is 'because there is a finite number of functional/procedural elements that license structural positions for arguments, once we run out of functional categories, we run out of structural places to put our arguments in'. That is not a nonsensical answer, but the problem is that currently there is no consensus on the number or nature of functional/procedural elements. For instance, and just limiting ourselves to the dynamics of the verb phrase and its extended projection, some authors assume a simple *v*-V dynamics (Kratzer, 1996; Chomsky, 2008; Krivochen, 2015c), others assume a Voice-*v*-V dynamics (not necessarily in that order, see Kosta, 2015; Harley, 2012), and there are some who add a further Appl(icative) head (e.g., Pylkkänen, 2008) or AntiCaus(ative) head (Kosta, 2015) among other proposed functional projections. Quite the same happens in the locative domain PP (with some authors splitting PP

[125] Notice that the notion of 'distance', be it metric or not, is of little help at certain scales: as Einstein himself put it, quantum entanglement is a '*spooky action at a distance*'. To this day, there is no consensus about what quantum entanglement actually is (we just know it is a fact).

into *PathP* and *PlaceP*), and so on (the 'explosion' of the category Inflection into several functional categories –Agreement projections, mainly- in the late '80s and early '90s is also revealing in this respect). Therefore, there is no *principled* (i.e., non-stipulative and independently motivated) way to establish the number of relevant structural positions, or formulate hard constraints with respect to how many arguments a predicate can take. We will thus tackle the issue from a different perspective: given an ultrametric approach to the topology of the lexicon prior to external perturbations, and assuming such an ultrametric space is describable in terms of fields, assigning a number or set thereof to each point (where the number of elements in the set depends on the number of dimensions we assume for the lexicon), then Conservation and 'slit-dynamics' taken together can predict the facts. We agree with Uriagereka in considering 'words' excitations of a field (actually, morpho-phonological words as well as 'conceptual addresses'), but we implement the dynamics of structure generation quite differently. For Uriagereka, a derivation is uniformly aimed towards entropy reduction, such that complete destructive interference is desirable (so that the system tends towards '*highly organized criticalities*', Uriagereka, 2011: 56)[126]. For us, as argued in Krivochen (2014c), there is no such thing as *uniform* entropy reduction in a strongly cyclic model based on mixed computation, because values of derivational entropy 'reset' to a value lower than maximal entropy (i.e., the value of entropy at $t = 0$) by a factor we shall mysteriously call n for the time being, once a syntactic cycle has been completed. The notion of 'cycle' or local, atomized domain for the application of operations, will be of crucial importance here, since locality considerations in physical systems lead us to propose that *only elements within certain local boundaries* (what we will refer to as 'cycles') *can produce interference effects*. More specifically, we have proposed in Krivochen (2015b: 280) that a dependency can be established between *token*-elements α, β, … (where either #α# or |α| for any α, β, …) if and only if α, β, … belong to the same derivational workspace at the time of establishing the dependency; this ultimately amounts to the notion of *accessibility* mentioned above (see also Uriagereka, 2002a: 53-54 for a characterization of this view in terms of a 'radical' version of MSO). In the framework introduced here, *we extend this condition to all interactions and define such interactions as interference patterns among elements that are ultimately perturbations in an electromagnetic field*. Derivational cycles, in turn, can be defined from two perspectives:

113) a. *Computationally*, as maximal-minimal finite-state compatible monotonically assembled units for purposes of structure linearization (Uriagereka, 2002a, 2012)

b. *Informationally*, as sub-domains of locally decreasing entropy in a discontinuous oscillatory function

(113 a) should already be obvious to a reader familiar with Uriagereka's MSO model, and we have also taken advantage of the computational definition in our (2015a) analysis of generative operations: this definition arises naturally in a computationally mixed system. The point we

[126] Interestingly (and despite its being still a minority approach) there is a healthy variety of opinions within the 'physics of language' approach, covering all logical possibilities. For instance, David Medeiros (p.c.) has commented in this respect that "My […] intuition strongly suggests that, *qua* natural system, it should rather maximize entropy. I had read something recently about extending this to the idea that natural systems tend to maximize *future* entropy, (which turns out to make different and better predictions in some domains)". The reader has thus been exposed, although to different extents, to 'uniform entropy decrease' (Uriagereka), '(uniform?) entropy increase' (Medeiros), and 'entropy reduction *qua* derivational cycles' (Saddy, 2018; us) proposals in the course of the present work.

want to make about (113 a) is that it is insensitive to the content of the terminals, as long as they can be *locally* structured in a monotonic fashion. If, as we have argued above (along the lines of the MSO model), linearization of structure is expressible in terms of dynamic 'Markovization' of that structure (Lasnik & Uriagereka's 2012 'dynamical flattening'), once a minimal monotonic unit has been assembled in such a way that the following derivational step involves a non-terminal (thus, the relevant unit is also 'maximal'), that portion of structure is linearized by means we will detail below. Therefore, the elements contained in that flattened unit are no longer available for purposes of further computations (cycles cannot be tampered with, but, as we will see, some aspects of past cycles can be accessed after the relevant derivational unit has left the workspace), and we can safely say that a *cycle* has been terminated. In any case, the main idea is that *cycles arise from a fundamental architectural tension* (a *dynamical frustration*) *which underlies a computational system that oscillates up and down the Chomsky Hierarchy locally*. The motivation for such an oscillation cannot be found in the abstract properties of the system alone, though. Unbounded generation (as is the case in an L-grammar) has no limits, and defines an infinite phase space[127], as we saw above (and was also noted by Saddy, 2018): unless somehow limited by hard conditions and bounded by representational stability (e.g., in the form of *labels* or *categories*), the computational system gives us no usable outputs. This situation radically changes when we put such an unbound generative system in the phenomenological world: a purely abstract generative engine needs no cycles (the famous 'halting problem', which, as we saw, also arises in the context of *computational frustrations*), as there are no architectural or implementational limitations. If that system is implemented *within* some other, substantive systems like sound and meaning –with their mutually orthogonal requirements-; limitations immediately arise, with cycles being emergent properties of the dynamical resolution of the tension between the unbounded possibilities of the grammar and the emergence of attractors within a limited and topologically specified phase space guiding interpretation routines.

The definitions in (113), needless to say, interact within the system: the introduction of a particular element or set thereof in the derivation may define a computationally uniform (i.e., monotonically assembled) domain, and *at the same time* locally reduce entropy, at that derivational point (which amounts to saying that a cycle can be defined either computationally, or informationally – semantically based-, *or both*). For instance, in the mixed-computation approach to auxiliary chains presented in Bravo et al. (2015), modals, phasals, and first-position auxiliaries (which we referred to as 'lexical auxiliaries') close domains in terms of transmission of temporal and/or aspectual information, but they can *also* take part on different kinds of dependencies: (i) a monotonic chain of modification over the lexical verb constituting a single domain, or (ii) a computationally mixed chain, in which the modal takes a sub-domain as a complement. These situations are illustrated as follows (the lexical auxiliary in **bold**, semantic modification relations are represented by arrows *from* the modified *to* the modified element):

[**Aux 1** [Aux 2 [Aux 3 [Lexical Verb]]]] E.g.: **Puede** estar siendo torturado
Can$_{3Sg}$ *be*$_{INF}$ *being tortured*

[Aux 1 [**Aux 2** [Aux 3 [Lexical Verb]]]] E.g.: Ha **tenido que** ser ayudado
Has$_{3Sg}$ *had to be*$_{INF}$ *helped*

[127] So does unbound deletion, as has been pointed out in the literature in relation to the generative power of the Standard Theory and its extensions (e.g., Berwick, 1984).

The first example features all auxiliaries modifying the lexical verb without modifying each other (they all 'auxiliate' but none is 'auxiliated'); in previous works we appealed to a Unification grammar (Shieber, 1986) to derive the dependencies in this example: *puede*(*torturado*), *estar*(*torturado*), *siendo*(*torturado*). The second example is radically different: to begin with, the perfective auxiliary 'haber' only modifies the modal, and the complex [perf + modal] modifies the lexical verb without caring about Aux 3 (which is part of what we have called the 'extended projection' of the lexical V in García Fernández et al., 2017). Crucially, this example only works if Aux 3 is passive 'ser': the dependency here is mildly-context sensitive. We refer the reader to the discussion in Bravo et al. (2015); García Fernández et al. (2017) for details, but the crucial point here is that the definitions in (113) are not mutually exclusive at all; in fact, they are just two different ways of looking at the process of chunking a representation into local bits. It is crucial to note here that if we were committed to a single level in the CH (say, uniform context-freeness), only a single format of structural description could be generated: that, in turn, predicts that a single kind of dependency could be established among elements in a structural description, contrary to fact. Sometimes, as we have pointed out, cycles are defined only by reference to the change in computational dependencies (as in the case of *coordination* types, to be analyzed in **Section 4.4**; or Uriagereka's 2002a approach to non-monotonic structure building), sometimes it is the introduction in the derivational space of a particular kind of procedural information (like 'modal' or 'aspectual-phasal'); and sometimes, the introduction of such a unit changes the minimal computational device required to generate a structural description of a string, and thus a cycle in the sense of *both* (113 a) and (113 b) is defined. The system thus characterized is much more dynamic than its MGG counterparts, which define cycles in narrowly syntactic terms only (e.g., *Barriers*, Chomsky, 1986; or *Phases*, Chomsky, 2008). By making the conditions over the delimitation of accessible domains explicit in formal and physical terms, we are characterizing a conception of locality that is more flexible than most available options nowadays, but also defines a phase space restricted by means of *hard conditions* at both the levels of physics (and substratum) and computation.

A feature that is common to Uriagereka's approach and our own is that cycles are *minimally interpretable, maximally generable units*. This, which might seem a mere aphorism, is a way of summarizing what a cycle is and how (and why) it arises, given a computationally mixed system with generative-interpretative dynamics. Unlike Chomskyan cycles (barriers, phases…), which arise from theory-internal stipulations over both building and labeling (a trend that is present even in proposals that present themselves as radical alternatives to MGG, like Boeckx, 2014: 40, ff.), cycles in the two senses in (113) are much more fundamental -since they arise from computational and physical properties of the system, respectively- and dynamical -insofar as we cannot say, *a priori*, whether a syntactic object is a cycle or not, as it depends on the legibility conditions of the systems that are supposed to read that object. Cycles are minimally interpretable units because nothing smaller than a cycle can be assigned an interpretation in terms of sortality, eventivity, or relations between them. And, they are maximally generable, because once the critical dimensional value 3 for the structure being generated is reached, a dimensional attractor fades due to the inherent limitations of the system, and *no further structure can be built up in that derivational current*. This is more dynamic than representational alternatives because we do not depend on the inclusion of a particular 'head' in the derivation (Complementizer and transitive 'little *v*', as in Chomsky's 2008 *phase theory*; see also Gallego, 2010 for an introduction), nor do we need to stipulate a computationally uniform system of structure generation (unlike Adger, 2012; Chomsky, 2013; Boeckx, 2014, *inter alios*). Moreover, our cycles are not narrowly linguistic (note the absence of strictly linguistic concepts in (113), like *phrase*, *head*, *maximal projection*, etc.); rather, they derive from the dynamical topological structure of the mind we have hypothesized here and the properties of syntactic operations -which change that topological landscape-. Thus, the present approach to cycles applies to any cognitive faculty, and more generally to any dynamical system implemented with finite resources.

An interesting question concerning the motivation for cycles arises when we consider the following quote from Rabinovich et al. (2014):

> (…) *chunking is grouping or categorizing related issues or information into smaller, most meaningful and compact units. Think about how hard it would be to read a long review paper without chapters, subchapters, paragraphs, and separated sentences. Chunking is a naturally occurring process that can be actively used to break down problems in order to think, understand, and make improvisation more efficiently.* (Rabinovich et al. 2014: 3)

In our framework, linguistic cycles are not really motivated by efficiency considerations; rather, efficiency is a by-product of cycles, which arise from deeper physical properties of the system determining the system's possible dynamics. There seems to be an empirical question to address in this respect: could there be an embodied cognitive system that, given the generation-interpretation and topological limitations ours has, but infinite time and memory, does not display cyclicity? Note that we are asking about the *possibility* of an *embodied cognitive system*, not a computer or a 'brain in a vat' situation. We can (and is customarily done, in fact) model a TM without appealing to cycles or local domains at all, and if we go down to, say, an LBA, 'cycles' are delimited by the specific symbols ¢ and $, not by the dynamics of the system[128]. Experimental evidence has shown that the 'magical number' 7 ± 1 *objects* put forth by Miller (1956) is actually to be refined to 4 ± 1 *chunks* (Cowan, 2001). The specific number of chunks is not the center of our attention here, but rather the fact that they are there. Since, to the best of our knowledge, there has been no study of chunks in topological terms, and since it has not been proposed that chunks could vary in their dimensionality and establish interactions and dependencies with other chunks of equally dynamic dimensionality[129] (as we propose here, with an upper boundary of D = 3 based partly on perceptual considerations and partly on topological properties of phrase markers), we will not deal with this issue here, but we do hope the present work inspires future research on chunking from the perspective advanced here.

[128] Arguably, Chomsky's version of Phase Theory (2001, 2008) mixes both symbol-triggered cycles with considerations about the dynamics of the system, insofar as phase heads (the relevant symbols) are the only ones that trigger feature valuation processes, which in turn prevent uninterpretable / unvalued features from reaching the interface levels, which would make the derivation 'crash'. See also Gallego (2010: Chapter 1) for introductory discussion.

[129] In purely topological terms, consider the operation via which we can build a Klein bottle by means of gluing the edges of Möbius strips: which is the dimensionality of a Möbius strip? Its Euclidean dimensionality does not coincide with its Hausdorff dimensionality, and there are also issues of perspective playing a role here: are we operating *within* the surface of the strip or are we looking at the strip inscribed in, say, an $\mathbb{R}^n$ space? If we are building manifolds, this is a non-trivial question, which can be approached from the perspective of a dynamical frustration arising when we attempt to relate *n* strips, an operation we think is analogous to that of relating cycles derived in parallel for purposes of global interpretation (and it is only an 'analogy' because it is not clear that the topology of cycles is that of a Möbius strip, but we thank Thomas Stroik for suggesting the possibility to us).

Mathy and Feldman (2012: 347) propose a conception of data chunking based on minimal description length. Let $d(s)$ be the minimal length description (in bits) for the string s. Then, according to these authors, $d(s) = K(s)$, the Kolmogorov complexity of s. The description function $d(s) \approx |s|$ if $d(s)$ ($|s|$ being the length of the string) is *minimal* (meaning, the minimal length description of a string cannot be much larger than the string itself). Optimally, $\forall(s)$, $K(s) \leq |s| + c$ (a constant) (see, e.g., Li and Vitányi, 1997 for discussion and further development)[130]. Focusing on strings of numbers, Mathy and Feldman hypothesize that simpler strings –consider a telephone number, which is normally chunked in some way or another- are easier to remember. There are some issues with this approach that we will briefly review here. To begin with, complexity is based on the length of the structural description of an object, which in this particular case is a string of numbers. Interestingly, it is not obvious that strings of numbers are subject to the cyclic considerations we have been discussing, or that they derive from any of the conceptions in (110) about cycles (since there is no semantics-morphophonology tension, and it is crucial to bear in mind that in our proposal, cycles emerge from a dynamical frustration). This makes it difficult to generalize the results (or even the methodology) to, say, linguistic stimuli, where the interplay of conditions over externalization and semantic interpretation plays a huge role. In our proposal, cycles arise as the optimal local solution to a dynamical (scale) frustration; it is not clear what the motivation is for chunks within a strict Kolmogorov-complexity / minimal description length framework. It is also not clear from Mathy and Feldman's discussion whether chunks can, in turn, be hierarchically organized: this process of chunking disrupts the linear function these authors assume relates length and 'rememberability': a Garden Path sentence like 'The horse raced past the barn fell' might be shorter than an SOV(+Adjuncts) sentence, but this does not necessarily correlate in a linear way with number of cycles (as these are not determined beforehand, but dynamically *during* the derivation) or with parsing efficiency (which partially takes us back to the discussion of Rabinovich et al., 2014). A minimal description length approach might be useful for comparing several Markovized (more generally, 'atomized', see below) sub-units among themselves, but it is unclear how it would apply to computationally mixed phrase markers, either in language or in any other cognitive capacity. The fact that our cycles are determined by the existence of an unsolvable tension between systems, and that they also depend on topological considerations (related to the dimensionality of the manifold the system is building) should be enough at this point to distinguish *cycles* from *chunks* as the latter are customarily understood in cognitive science. This does not mean they are incompatible notions at all; as a matter of fact, it seems that chunks are (at least in Mathy and Feldman's system) the simplest case of cycles, applying to strings. Feigenson's (2011) distinction between *objects*, *sets*, and *ensembles* is, in our opinion, better equipped to deal with the complexity of language, and, more generally, those systems in which a dynamical frustration arises. Rosemberg and Feigenson (2013) tested the hierarchical organization of working memory ('chunk nesting'), as well as the possibility of keeping more than a single level of organization active at a single time (e.g., an object *and* a set, Rosemberg and Feigenson, 2013: 619) in infants in terms of *objects, sets*, and *superchunks* (ensembles), when they had to remember the position and properties of a certain number of items. Spatial and relational information comes into play here (the position and properties of yellow balls in relation to one another and with respect to the platform they were on top of, in addition to black foam-covered boxes), unlike strict number-sequence experiments.

[130] Informally, '*As an upper bound, a maximally complex sequence of N items will require about N slots to encode it, while at the other extreme an extremely repetitive string may be compressed into a form that is much smaller than the original string.*' (Mathy and Feldman, 2012: 347)

The general problem within which these considerations arise can be formulated in the following terms (Rabinovich et al., 2014: 1)

> *What are the mechanisms that transform the extremely complex, noisy, and many-dimensional brain activity into a rather regular, low-dimensional, and even predictable cognitive behavior, e.g., what are the mechanisms underlying the dynamics of the mind, including chunking?*

A good part of our thesis can be summarized in this question. Our answer, which we have partially developed, is related to two main characteristics of the system we are dealing with:

a) The existence of a dynamical frustration
b) The physical limitations on structure generation imposed by the neurocognitive substratum

The specific instantiation of the mechanism that derives cycles in a topologically dynamical system, implemented through the *center manifold theorem* (which we will review in **Section 3.9** below), depends on *both* (a) and (b). The implementation of a mechanism that can reduce the dimensionality of a manifold once a critical value is reached (either via the manifold's inner dynamics, as each derivational step that builds structure increases the manifold's dimensionality; or by making manifolds intersect, as if we were building a Klein bottle by means of gluing two Möbius strips together; see Saddy, 2018 for discussion) within an overall cumulative architecture in which these lower-dimensional manifolds can be inputs to further operations is one of the main features of our oscillatory computational dynamics. Interestingly, a dynamical frustration would arise even in a system with an infinite memory tape or operating capacity (the concept of *computational frustration* of Binder, 2008); and while physical limits over the emergent computational properties involve memory limitations, it is not obvious that finite memory alone gives us cycles in the sense studied here. Consider, for instance, PDAs (or PDA+s): their memory is limited to a first-in first-out regime (see Uriakerega, 2008: 228 for discussion within a linguistic paradigm), which should derive a strict and static form of periodic cyclicity, of the 'every phrase is a phase' kind (Epstein and Seely, 2002, see also Boeckx, 2010, 2014 for discussion). The point is, a mechanistic computational implementation of a generative (fully explicit) procedure need not capture the physical properties of the system, but the physical theory, in its computational aspects, must be mechanistically implementable. Let us consider an example of what we consider to be one of the best such mechanistic architectures currently available, the Queue-Left-Stack-Right (QLSR) architecture developed in Medeiros (2015a, b; 2016):

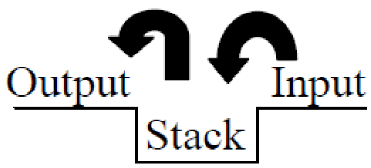


Figure 16: abstract stack-sorting architecture

Medeiros (2016) describes the sorting architecture as follows:

- *input list is transformed to output list with aid of last-in, first-out memory stack.*
  - *Compare top of stack to next symbol in input*
  - *Pop stack to output until input is larger, then push input to stack.*

- *The algorithm correctly sorts some sequences, but not others.*

The Input consists of a set of elements with integers assigned to them, which justifies the use of a sorting procedure to get ordered sequences (Knuth, 1998). Based on a universal hierarchy of projections (á la Cinque, 1999 and much related work), the QLSR architecture takes an input modelled upon linear order, and mechanistically maps it onto an output representation modelled upon a universal syntactic-semantic hierarchy. The sorting mechanism sorts some sequences (i.e., correctly maps orders into underlying cartographical hierarchies), but not others; and achieves an impressive range of descriptive adequacy: for the case of DP-internal orders, the QSLR architecture derives the facts noticed by Cinque's (2014) survey (dark coloured cells are orders that the QSLR architecture cannot sort, and they are also not attested):

Dem Num Adj N 103 = 18.6% 319 = 20.8%	Dem Num N Adj 52 = 9.40% 121 = 7.88%	Dem N Num Adj 22 = 3.98% 38 = 2.48%	N Dem Num Adj 18 = 3.25% 50 = 3.26%
*Num Dem Adj N 0	*Num Dem N Adj 0	*Num N Dem Adj 0	*N Num Dem Adj 0
*Adj Dem Num N 0	*Adj Dem N Num 0	Adj N Dem Num 5 = .90% 15 = .98%	N Adj Dem Num 29 = 5.24% 78 = 5.08%
*Dem Adj Num N 0	Dem Adj N Num 24 = 4.34% 36 = 2.35%	Dem N Adj Num 69 = 12.5% 132 = 8.60%	N Dem Adj Num 16 = 2.89% 23 = 1.50%
*Num Adj Dem N 0	Num Adj N Dem 16 = 2.89% 40 = 2.61%	Num N Adj Dem 47 = 8.50% 188 = 12.2%	N Num Adj Dem 25 = 4.52% 41 = 2.67%
*Adj Num Dem N 0	*Adj Num N Dem 0	Adj N Num Dem 11 = 1.99% 24 = 1.56%	N Adj Num Dem 116 = 21.0% 430 = 28.0%

Cinque's (2014) sample. 553 genera; 1535 languages.

Table 2: Most common orders within the DP and QSLR generability.

Medeiros (2015b) makes a strong claim about the division of labour between structure building and structure mapping: "*transformations (mapping word order to meaning, not trees to trees) are enough*", but he is forced to assume a basic order as a desirable output, to which actual outputs of the process are compared. Thus, each element in the input is assigned an integer *i*, and the mechanism sorts elements in the input to get an output in which integers are ordered, with a last-in-first-out stack mediating between input and output representations. This entails accepting a certain cartography for syntactic structures as a basic order, like SVO for clauses, or D#AN for the DP. This is part of the reason why we claim that the QLSR architecture is not a theory of language, but a mechanistic implementation of such a theory, and, given a set of inputs and a set of permissible outputs, a reliable tool for testing derivational procedures, as long as such procedures are expressible in terms of computable functions.
The model's simplicity and computational convenience reside in the fact that it can be implemented with only five lines of code (Medeiros, 2016):

While input is non-empty,

If I > S, Push.
Else Pop.
While Stack is non-empty,
Pop.
[Where:]

- *I: next element in surface order.*
- *S: element on top of memory stack.*
- *>: a > b if a is deeper in the cartographic hierarchy than b (e.g., N > Adj > Dem).*

Notice that the cartographic hierarchy *must* be *independently* motivated, for the QSLR architecture does not derive it (this is evident in the last line of code). This means that the QLSR architecture requires a linguistic theory (or any other kind of theory) to provide it with atomic elements to build input sets with, and a (recursively enumerable) set of permissible outputs.

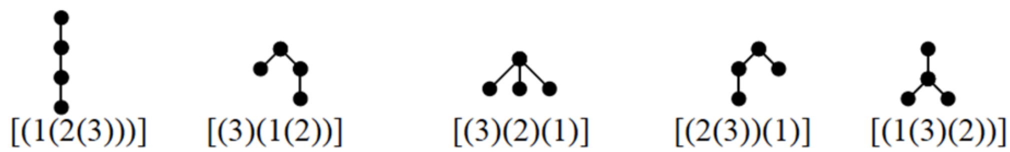
A potential problem for the mechanistic implementation of a computational theory via QLSR consists on the exact nature of cycles and the way to relate cycles that display different computational complexity, following the ideas presented in Lasnik (2011), Lasnik and Uriagereka (2012), Krivochen (2015a), among others. Independently, when analyzing the properties of the stack sorting architecture, Medeiros and us came to the conclusion that it is necessary to include a 'placeholder' symbol (which we referred to elsewhere as a 'joker') in an input set, which can at the level of output be replaced by the output of a parallel stack (something akin to Joshi's 1989 Extended PDAs with multiple parallel stacks)[131]. This placeholder, which we will identify as [$Ø_n$], can be assigned any integer, and can be replaced by any other symbol that has come as the output of a parallel stack, by an operation that is very much like a Generalized Transformation. Let us see an example of the kind of procedure we are proposing could capture a mixed phrase structure model within the QLSR system:

114) Target: The three old, old, old man (cf. the trees in (26) above)

Input A:	Input B:
[The$_1$]	[old$_1$]
[Three$_2$]	[old$_1$]
[$Ø_n$]	[old$_1$]
[men$_3$]	

Both Input sets are parallel, and it doesn't really matter which integer is assigned to the elements in the Input set B, as long as they are all equal (thus, they cannot be relatively ordered). We assume that in such a situation, the system delivers a 'flat' structure, which is compatible with Medeiros' (2016) assumption that the sequences of Push and Pop define a graph, more specifically, a Dyck tree. If all elements in an input are assigned the same integer, there is only one Dyck tree of all possible orders of three constituents that captures that dependency without any addition of extra structure (taken from Medeiros, 2016):

115)



[131] What follows is our own extension of Medeiros' architecture, and, unless explicitly stated, the vision we present here is not to be identified with his.

Note that only the third tree actually presents a strictly flat dependency in which there is no scope or hierarchical relation between the elements of the string: that goes well with our proposal for a Markov syntax for iteration. What we propose is that the third Dyck tree is inserted into the terminal node occupied by the placeholder [$Ø_n$] in the parallel stack via a GT. The value of the placeholder's diacritic integer is determined contextually, depending on the categorial nature of the subtree to be inserted (basically, its 'label') and the position of that specific projection in the universal hierarchy. Now, here is a problem, because it is not clear that a purely generative, mechanistic procedure should (or even could) be label-sensitive. Medeiros (2016, p.c.) has noticed the same problem:

> (…) *a potentially long, internally-complex "constituent" (phase) can effectively be the leftmost (say) element in the next cycle up (e.g., a DP within a CP). That said, these reference pointers then do need to get sorted and end up written to an output tape eventually. Which brings up the issue of labelling* (…)

It might be possible to say that the label of a construction of arbitrary complexity does not matter as long as it is realized as a terminal node (recall we are working with a grammar in CNF), which is fine for purely formal purposes, but for natural language purposes, labels *qua* syntactic-semantic objects seem to be needed. At this point, it is not clear. Moreover, the specific mechanism by means of which a GT could be implemented within QLSR is not straightforward, as Medeiros (p.c.) correctly pointed out. What turns out to be interesting is that the system does not require inter-tree transformations (i.e., there are no transrepresentational rules), because all sortings occur only within an Input-Stack-Output cycle. That means that the QLSR architecture naturally implements a radical version of MSO, and if it is further extended by means of a placeholder symbol, it seems our broader physical conception of linguistic derivations in a cognitive system is actually equivalent in generative power to QLSR when it comes to mechanistic implementation[132]. Unlike Medeiros, we conceive of the QLSR architecture as the necessary limit to which physical and neurocognitive approaches must converge when they are translated to code language, manipulation of discrete symbols via a finite sequence of steps in polynomial time. Of course, we have argued that there are crucial aspects of language which do not fall within this category, and thus the stack sorting architecture cannot apply to them, because they are of a very different nature. Moreover, it is not clear that QLSR can handle non-normal computation, given L-grammars' lack of representational stability: if there are no labels, the dynamics of placeholders becomes much trickier. Placeholders, insofar as they encode cross-cycle dependencies, are the units that seem to push the computational requirements of the architecture beyond strict PDA limits: it is the placeholder symbol that requires an operation of *adjunction* in the terms of Joshi (1985: 209, ff.) for its interpretation. Let us diagram the procedure of β adjoined at node X within tree γ (taken from Joshi, 1985: 209):

[132] At this point, a mathematical proof of that equivalence would be welcomed, but it has not been devised. Mainly, because QLSR only captures the mechanical aspects of derivations, those which can be modelled using functions in α-machines (Turing, 1936). However, even within Medeiros' architecture it seems to be the case that some configurations *need* to be ambiguous (e.g., [I saw the man with the binoculars], in which the abridged relative clause can be attached high or low), and such ambiguity points towards c-machines as a suitable model. More generally, the space of inferences and associations is not function-computable, but is captured by analogic operations over the cognitive phase space in the field-theoretical approach.

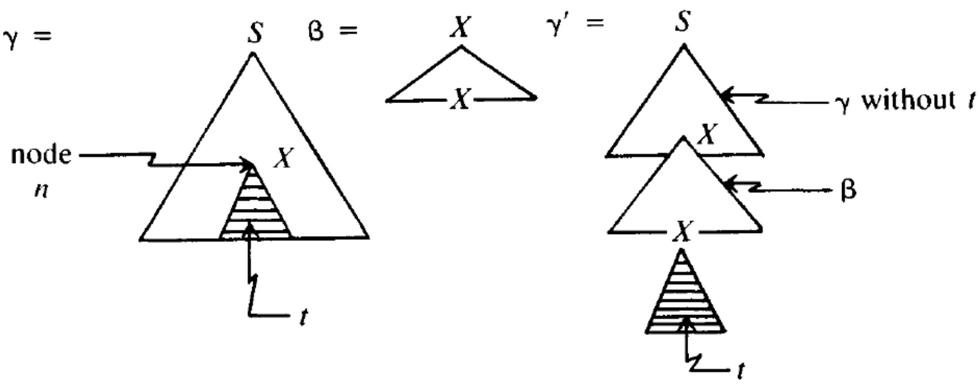


Figure 17: Adjoining operations in TAGs

For the operation of *adjunction* to work, both γ and β must contain a node X, which has to be a root node (topmost) in the case of β: adjunction identifies both, and inserts the auxiliary tree β into γ, yielding γ' (so it is basically an asymmetric operation, which targets a location within a tree and replaces it with an auxiliary tree whose root node matches a node in the target). It is important to point out that there can be material in γ which is not affected by the operation (which Joshi notates as *t*), even after X ∈ γ is replaced by the auxiliary tree β. That is what, we argue, occurs in the case of placeholders within a sorting derivation. Note that in (111) there is a 'remnant'within γ (Input A), [man_4] in this particular case; Joshi's formalism seems to be compatible with (but not equivalent to) the QLSR architecture (a point that Medeiros has noted). In the terms we have managed in the present revision, the operation *adjunction* operates over Dyck trees, and looks for placeholders within an Output representation.

This extension, although in principle possible, is not free of problems. Open questions (particularly from the perspective of code implementation) remain, including how to solve a situation in which an Output representation contains more than a single placeholder: these are underspecified and could in principle be replaced via *adjunction* by any other Dyck tree. If the replacement of placeholders was followed by a further sorting, in which the integer of the placeholder was defined contextually, the problem could potentially be solved, at least formally. This means that any sorting including placeholders has to be performed *twice*. The specific consequences of this for the mechanistic implementation of the architecture are still unclear. Medeiros (p.c.) also identifies a problem pertaining to where placeholders 'point to': '*A question, though, is where this reference pointer* [placeholders are in fact reference pointers, since they point towards another constituent or subderivation] *is written to: either directly to another output tape, or back to input. I think the right answer is back to input, as an element effectively in surface order, to be (queued or) stack-sorted in the matrix cycle*'. In the present extension of QLSR, placeholders are replaced by adjunction directly in an Output tape, which is in turn the Input to a further sorting. The output tape, crucially, must allow tampering: once the placeholder has been assigned an integer depending on the cycle it is instantiating, the place it occupies in the overall order is relationally established, via a *mildly Context-Sensitive* procedure. However, there are potential timing problems when relating substrings. In Medeiros' words, '*given surface strings A and B, adjacent to each other, each with its own analysis -- does A go in B? Does B go in A? Are either or both embedded into things which occur further left or right, so A and B don't attach directly at all?*' This is *not* a trivial problem, because it pertains directly to the asymmetric nature of *adjunction*: how do we determine which tree is the target of the operation? We have assumed that stacks are parallel, so the only element yielding asymmetry is the presence of a placeholder in one of such outputs. But if the placeholder-containing Output and the auxiliary tree are derived in parallel, it is not clear the auxiliary tree will have been fully derived by the (derivational) time the main tree is completed. Trivially, any A, B are embedded in an *ad hoc* C. But less trivially, we find ourselved facing a timing problem: does A contain a placeholder that calls for an element of B (let us say 'a term of B', such that B can be the term itself)? Does neither of them contain a placeholder (and thus there are no intra-tree relations, in which case the null hypothesis, that is, that A and B are embedded in C, holds)? It is possible that the architecture itself cannot sort this out, unless specific rules about derivational timing (something akin to early Minimalism's *Procrastinate*), or hierarchy between placeholders / rules involving placeholders are explicitly added. It is not clear, however, how this hierarchy (which is essentially the kind of constraint structure we find in OT / Harmonic Serialism), could be implemented codewise. The hierarchy solution, however, is not straightforwardly applicable to the case in which we have *embedded placeholders* (a placeholder is to be replaced by a structure in turn containing a placeholder), particularly if stack tapes are parallel. In this sense, we also have to note that the problem of putting cycles back together (the *address issue* in Uriagereka's 2012 terms) has the concomitant problem of determining how units are kept active for retrieval in order to be adjoined to placeholders, and how many units –and of which complexity, should there be a limit- can be kept *distinct* and active[133]; apart from the problem already pointed out of multiple placeholders and the necessity to somehow distinguish them via diacritics (such that not any sub-derivation can replace any placeholder freely). In this case, fixing the order in which the placeholders are replaced by subtrees via adjunction seems to be the only way to go, but exactly how to establish a timing hierarchy is still an open question.

Summarizing our brief discussion of the QLSR architecture, and taking into consideration problematic issues pertaining to global and cross-cyclic effects –alluded to above-, the crucial aspect is that it seems to be the case that a *mechanistic implementation* of a derivational system must be compatible with QLSR at least within local derivational horizons (i.e., without considering cross-cycle dependencies). Moreover, it seems equally important that the predictions made by a theory of the dynamics of syntax as a cognitive mechanism be explicit enough to be translatable into a form that is QLSR-implementable. We will see that, in the case of the framework exposed in the present thesis, those aspects which are modelable in classical computational terms are also QLSR-compatible, and the TAG-extension of Medeiros' architecture can perfectly model empirical aspects dealt with in this work, including the syntax of iteration and our discussion of coordination (see also Krivochen and Schmerling, 2016).

Once the computational requirements for a theory to be implemented via code have been outlined, let us resume our more programmatic discussion. A general point we have to make is

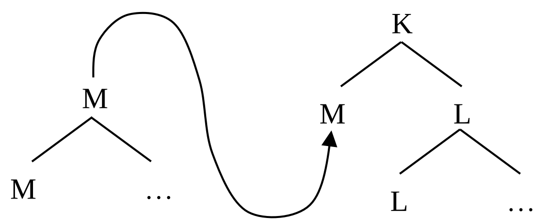
[133] In cognitive terms, this problem seems to be related to Feingenson's (2011) distinction between *objects*, *sets*, and *ensembles*. Objects are internally opaque, thus, the distinctness condition does not hold. The set of computational Outputs should contain mutually distinguishable units to be adjoined to different placeholders at different levels of embedding. The connection is promising, but we have to bear in mind that, in the context of the present work, the QLSR architecture is not interpreted as a cognitive theory, but a computational-mechanistic set of procedures. In contrast, Medeiros (2017) presents a slight reworking of QLSR, which he calls ULTRA, as a psychologically real parser, a view with which we disagree for reasons discussed in **Part II** (given the fact that ULTRA still implements mechanistic procedures, and is essentially a PDA).

not only that linguistic computations give rise to cycles, but also that it is also necessary for any system in which such chunked computations are implemented that it has a way to somehow put cycles back together to get an interpretation of a derivation as a whole. This process is something that can happen either on-line and in tandem with chunking (as in Montague-style grammars[134]) or after all chunks have been derived (in turn, either separately or in parallel) and identified ('labeled'). How to do this is *not* trivial: Uriagereka (2008, 2012) refers to this as an '*address issue*', because there are symbols within representation that point towards other symbols (or whole sub-representations which are instantiated as a terminal for the purposes of further computations, a process we will refer to as 'atomization' later on). Within an LCA-driven MSO architecture, Uriagereka (2012: 75) formulates the issue as follows:

> <u>Address Issue:</u>
>
> *Whenever a phrase marker K is divided into complex sub-components L and M, for K to meet LCA conditions of Multiple Spell-Out, the daughter phrase-marker M that spells-out must correspond to an identical term M within K*

Such a formulation (at least in the 'radical' interpretation of the MSO system, which we think is the most interesting one both theoretically and empirically) implies a nonterminal |M| = {*a, b*, …*n*} monotonically assembled leaving some kind of 'trace' of its original structural position, call it #M#. Why does this trace *need* to be a terminal? Because, otherwise, the resulting phrase marker K would not be LCA compatible, there being a symmetry point between M and L. Visually,

116)



'radical' Spell-Out of Specifiers
(Uriagereka, 2012: 170; 2002a: 49-52)

In the 'radical' version, '*each spelled-out CU* [command unit] *does not even merge with the rest of the structure, the final process of interphrasal association being accomplished in the performative components* [Conceptual-Intentional and Sensory-Motor]' (Uriagereka, 2002a: 49). The crucial issue is that there must be some instruction left after the succession of pieces of structure, for either the generative component (in the 'conservative' approach) or the interface components (in the 'radical' approach, which is closer to our own proposal here) to re-assemble the chunked structure, in the kind of modified Y-model Uriagereka assumes (which is compatible with a semi-autonomous syntactic component). We will see that this is just one of the arguments in favor of global conservation tendencies, which require 'remnants' of previous derivational steps (sort of 'representational crumbs', in a Hansel-and-Gretel kind of syntax) to remain in the active workspace. We will also come back to the *address issue,* insofar as it is one of the main problems to be solved from a cycle-based theory: how to put together what was once chunked?

[134] Thanks are due to Susan Schmerling for discussing nuances of Montague grammar with us, including the role of semantic interpretation and its timing in relation to the building of syntactic structures.

The Issue is a consequence of having cycles in the computational system: if there is only one derivational space and objects are built entirely in a single move, without chunking, the Issue never arises. Going back to the definitions of cycle presented above, the informationally-based (113 b) is neither obvious nor uncontroversial. Unlike (113 a), (113 b) is content-sensitive insofar as it deals with the information conveyed by the relevant terminals *and* their syntactic relations: (113 b) arises from a mixed computational perspective, which, as we have seen, features a syntactic mechanism which (in natural language) is shaped by semantics (in traditional terms, 'semantics is generative'; see McCawley, 1968 for useful discussion). Also unlike (113 a), it is dependent on *time* (arguably, a purely computational definition like (113 a) is time independent, particularly under generative assumptions, see Chomsky, 2007: 6), but arguably not dependent on *space*, understanding *space* in a purely topological sense: (113 a) holds independently of the geometry of the representations and the characteristics of the workspace they are derived in. Being linearization-based, (113 a) makes assumptions about the relevant geometry of linearizable (and non-linearizable) objects which do not directly affect (110b). Before expanding on the relations between these two conceptions, which are mutually irreducible (to the best of our knowledge), it will be interesting to briefly review another derivation of the concept of 'cycle' in human language: let us call it the *Zipfian cycle*.

Zipf (1949: 35, ff.) explores the applicability of a (classical) harmonic motion equation (based on a Taylor series) for the '*group behavior of* (…) *a vocabulary of words in the stream of speech*' (1949: 37). In Zipf's terms, there is an *optimization* process between competing and opposite principles, in the case of a harmonic oscillator, relating size of the oscillator and frequency, '*in which the saturated harmonic equation will precisely reveal itself*' (1949: 38). Applied to language, Zipf formulates two principles (admittedly informally), or rather Forces, which he calls 'Unification' and 'Diversification':

117) <u>Force of Unification:</u> minimize the number of words (to 1), maximizing their frequency (to 100%) and associated meanings

<u>Force of Diversification:</u> maximize the number of possible words, minimizing their frequency and associated meanings (tending towards a bi-univocal relation between words and meanings)

These 'forces' are usually identified with principles of economy centered in the speaker and hearer respectively (and have had long-lasting echoes, including Horn's 1988 Q and R Principles; and Uriagereka's 2012 treatment of the different perspectives within his CLASH model). Based on quantitative evidence from the Ulysses and the Iliad, Zipf claims that '*in the language of those two terms we may say that the vocabulary of a given stream of speech is constantly subject to the opposing Forces of Unification and Diversification*' (1949: 21), a perspective of conflict resolution and optimization that is not alien to Harmonic grammars or bidirectional constraint interaction models (e.g., Jäger, 2002; also Blutner, 2000; Blutner & Strigin, 2011 for a pragmatic perspective), or even syllable structure as the emergent of a Fib grammar within the CLASH model (Uriagereka, 2012: 286-287, for a dynamical frustration perspective on the interaction between Unification and Diversification, which we expand on here). His research on the relation between these opposing principles, and the optimal way to solve that tension leads him to say that

> *If we explicitly assume that the harmonic seriation $F \cdot Sn$* [expanded in the form of a Taylor series] *represents a fundamental principle that governs the number and frequency*

> *of usages of words in speech, then we can only conclude that a given speaker 'naturally' selects both the topics of his conversations and the words with which he verbalizes them in such a way that the resulting frequency-distribution of his continuing stream of speech will meet the exigencies of our equation, F · Sn, without 'too little' or 'too much' talk. And this, in turn, means that inherent in the stream of speech is a dynamic unit which we may call a closure (or a cycle, or a rhythm) which might be defined roughly as the length of speech during which a particular group of verbal tools has completed its collective behavior once. What else this closure may signify we do not yet know* (…) (Zipf, 1949: 38-39)

Like Uriagereka, Zipf bases the resolution of the tension on linearization requirements (thus the frequent mention to the 'stream of speech'), interestingly, a decade before Tesnière's (1959) considerations about the different (and conflicting) natures of semantics and morpho-phonology from a structuralist perspective. In Zipf's perspective, '*we may even visualize a given stream of speech as being subject to two 'opposing forces*' (Zipf, 1949: 21), identified above, and which determine the quantitative properties of a string (e.g., type-to-token ratio at word and syllable levels). To the best of our knowledge, Zipf's work is the very first attempt to characterize a computational notion of cyclicity in natural language, and, what is more, to link it to a fundamental tension rooted on deeper properties of the system from which language emerges (even though their nature was unclear at the moment: where do Unification and Diversification come from?), as well as considerations of usage. It must be highlighted, furthermore, that the notion of cycle Zipf assumes has strong cognitive correlates, since '*schizophrenic speech can almost be characterized by the absence of closure*' (1949: 39). Whether these considerations actually apply or not to specific pathological conditions (and see Zipf, 1949: Chapter 7 for more details) is not relevant for our purposes, insofar as we are setting our focus on the physical architecture that both *generates* such a tension between orthogonal components (the 'hiatus' or discontinuity, in the words of Uriagereka) and *solves* it locally (i.e., via local domains) in an optimal way. Interestingly, while we *can* in theory force the system to favor one interface over the other (or one force over the other), those are, according to Zipf (1949: 284, ff.) characteristics of pathological conditions (autism, schizophrenia) rather than the normal state of affairs in cognitive development. This is to be expected if a dynamical frustration is at the core of both cognition and its physical bases; and also if cognitive impairments impacting on language are *mapping* impairments (Kosta and Krivochen, 2014b) which directly affect the available resources to solve the frustration between conceptual and morpho-phonological requirements. In past works we have hypothesized that

> (…) *each materialized* [i.e., 'externalized'] *linguistic stimulus is the best solution to the tension between what the speaker wants to say, that is, the information he wants to convey (information that is syntactically structured); and the means he has to externalize that meaning, which implies a structural flattening from hierarchy to linear dependencies.* (Kosta and Krivochen, 2014b: 41)

The connections between our conception of the cycle, the notion of dynamical frustration, and Zipf's theory should be, at least, apparent. In the present work we have focused on the 'mentalist' aspect of the process and the role of cycles as emergent properties of the derivational dynamics we have described (in turn constrained by more basic physical principles), but a more complete picture of the implementational level of the theory should of course include considerations of language use in context, which, in the terms we have been presenting in these last paragraphs, amount to choosing the best candidate c_x (from a set of possible candidates C =

$\{c_1, c_2, c_3, \ldots, c_n\}$, *à la* Optimality Theory; see Prince and Smolensky, 2004) in a bidirectional system that solves the linearization-hierarchy conservation tension in an optimal way *for a speaker S, given finite resources* (working memory, time, lexical knowledge…) *in a communicative/situational context*. In such a dynamics, the tension between the mutually orthogonal principles Q and R (Horn, 1988), which present competing economy principles based on the speaker and the hearer, should also be included in the picture. This yields a multidimensional dynamical system in which several phase spaces intersect and interfere to result in an externalized, linearized structure which conveys meaning *for a subject, to another subject, at a moment in time, and in a situational context.* That said, we will keep formalizing what happens 'within a speaker's head' from a physical perspective, although the importance of an extension of the theory along these lines cannot be overstated.

3.5.3 Reflections on a classical problem under non-classical assumptions

Let us resume our discussion of derivational cycles in a classical step-by-step architecture along physical terms; this will be an introduction to the purely field-theoretical perspective on derivations we will present in **Section 3.6** and following. Assume, for the time being –and following usual *proof-theoretic* practice-, that we have a set of elements drawn from the Lexicon to work with, and a workspace within which operations apply to those elements in order to yield more complex structures, which are then interpreted by systems of sound and meaning. Assume, moreover, that structural complexity is obtained exclusively by recursive combinatorics, implemented through an operation *concatenate* which just puts things together. These assumptions will be revised and corrected in **3.6**, but as a transition between current syntactic theory and our modified derivational engine, this will do. If entropy reaches its maximum level when all elements are equally likely to enter the derivational space[135], any restriction regarding the set of elements that can be merged in the following derivational step (see Putnam's 2010: 8 definition of "Soft crash", which includes this notion of *local derivational unit*[136]) will make entropy decrease, since the system is asymptotically tending towards a fully *ordered state* in derivational cycles: the function describing the behavior of such a cyclic system is, naturally, periodically discontinuous. Why? Because if entropy h in derivations is plotted by means of a function $h = f(x)$, and there are cycles in the sense of either (110 a) or (110 b) (or both!), then when the values of x (where x denotes derivational steps) approach a derivational step in which a cycle has been finished (say, at a derivational point D), $\lim_{x \to D} f(x)$ is undefined at D[137]. Let us look at a linear example whose mechanics will be familiar to those readers with an MGG background. We will modify some aspects of these mechanics below, in terms of a sum-over-paths perspective. For the time being, and using a widespread assumption, a linguistic derivation starts when a finite set of items (a Lexical Array) is drawn from the Lexicon, the full set of conceptual and procedural elements represented in a speaker's mind, in order to convey a particular CS (Uriagereka, 2011: 59 for a related perspective). The

[135] Notice that this holds independently of the specific topology that is assumed for the derivational space: as we have argued in Krivochen (2014c [2016]), it is also true for binary-branching trees of the Minimalist kind.

[136] Let us cite the whole definition: '*If a syntactic object α cannot be interpreted at an IL in any and all of its features, α is neither useable nor legible at IL, iff α cannot be combined with another local derivational unit that repairs the violation(s) of α*' (Putnam, 2010: 8)

[137] Relevantly, this can be also phrased as 'the function $f(x)$ is not differentiable at D'.

question whether elements are drawn from the Lexicon all at once or step by step is not relevant for the present discussion (but will be addressed below): we might not need a Numeration or a Lexical Array (or any such concept) at all, optimally; nor should we need sequentiality to guarantee well-formedness. However, for the sake of clarity and intelligibility for linguists, we will for the time being assume that there is a finite set with which we derive a sentence, even though the concepts of Array and Numeration is highly problematic (see, e.g., Stroik, 2009: 44; Jackendoff, 2011: 275). Each one of the items featured in the relevant array is a *type*, and each of their instantiations in a derivation is a *token* (see Krivochen, 2015b for extensive discussion and implementation; also Chomsky, 2013 for an orthodox proposal). A *token* can thus be defined as a *type in a syntactic context*. Given this scenario, let us consider the following (unordered) array of terminals:

118) *Array* = {D, P, √α, √β}

(where D = determiner, conveying sortality; P = proposition, conveying location; √*x* = LF-underspecified root)

Of course, any of these elements could enter the workspace W first (if no stipulations are assumed), so the *first step in a derivation has the maximum entropy*. Following standard practice, let us assume entropy values between 0 (complete order) and 1 (complete disorder). Non-trivially, no mental computation actually ever adopts values of $f(x) = 0$ or 1, because complete informativity and complete disorder are effectively computable but not effectively computed, each for different reasons that arise only when the system is implemented in a mind. Therefore, 0 and 1 constitute limits for the relevant function, and the value of h (i.e., $f(x)$) at x = 1 (the first derivational step) is arbitrarily close to 1 (technically, we will say that $\lim_{x \to 0} f(x) = 1$, provided $h = f(x)$); non-trivially, the space in which the relevant elements are represented *before* a particular derivation is Hausdorff, that is, any two different (i.e., topologically distinguishable) points have different neighborhoods (Bourbaki, 1966)[138]. Let us naïvely assume, also, that the Numeration is accessed only once per generation, and the structure-building operation proceeds binarily (à la Kayne, 1984; Chomsky, 1995, 2013; among many others). We will see in the following section that none of this (which is customarily assumed in MGG) is actually necessary to derive the crucial aspects of derivations: cycles or any kind of locality condition on the one hand, and periods of entropy decrease on the other. Let √α enter the derivation first[139]. This makes the global entropy of the system decrease, since not all elements in the initial Array may enter the derivational space and establish a dependency with √α while satisfying the principle we have called *Dynamic Full Interpretation* (DFI), very roughly a step-by-step relativization of Uriagereka's REM. Depending on the conceptual structure that this syntactic derivation is to embody, the possibilities for P vary, but assume we want to build a full-fledged

[138] More technically (and equivalently), "*Any singleton set {x} ⊂ X is equal to the intersection of all closed neighbourhoods of x*" (Bourbaki, 1966: 75).

[139] See Di Sciullo and Isac (2008); Stroik and Putnam (2013); De Belder and van Craenenbroeck (2011) for examples of different criteria to determine which element enters the syntactic workspace first. All of the references assume some kind of featural defectivity in the element (e.g., this element lacks a selection feature, which implies that it is selected by another element, but does not select a complement itself), which we do not.

locative structure, containing sortal entities related in a central / terminal coincidence manner[140]. If this is the case, the possibilities for P ≈ 0, since the conceptual structure requires sortal entities; and the possibilities for √β ≈ 0 as well since roots are semantically underspecified and cannot be interpreted by C-I unless they appear within a larger structure containing a procedural element (see, e.g., Krivochen & Kosta, 2013: 89-90; also Panagiotidis, 2014 for a somewhat different approach to root underspecification); therefore, their inclusion would not yield a drastic interface effect, if syntactic operations are semantically driven. This means that we have an optimal situation: there only one *type*-candidate whose instantiation as a *token* satisfies DFI, in a strongly derivational model. Stipulative though this reasoning might seem, it is not too different from saying the system is dynamic, learning-sensitive, and tends towards zones within the phase space limited by the sound-meaning systems where interpretative attractors are more likely to be found (which is a more accurate depiction of the relevant condition). In any case, the following derivational step would then be:

119) *Concatenate* {√α, D} = (√α, D)

Note that should we have introduced D first, the results would have been the same since generation does not care about order or the "side" of the tree in which symbols are inserted, unless one is willing to concede that the computational system $C_{(HL)}$ is *both* generative and interpretative. For the purposes of future computations, this structure will be interpreted as a sortal entity, where sortality, as a semantic effect, is triggered by the local relation between the procedural element D and the root √α (note they are both *terminal nodes*). Having thus a 'sortal entity' in the workspace (in traditional terms, a Determiner Phrase or Noun Phrase), the situation changes: even though merging another D token is not possible if DFI is to be satisfied (as it would be superfluous), we still have, *in principle*, three logically possible candidates, with which the possibilities for each rise from 25% to 33.33…%. This increases the entropy again and, if we had no way to determine what must come next, the derivation could not continue; the computation would not (could not!) be *cumulative*. In previous sections (and works) we have argued in favor of a pre-linguistic conceptual structure that is instantiated linguistically via the *Conservation Principle*[141] (hereafter, ConsP) mentioned previously in connection with the structure / content tension:

---

[140] In past works (e.g., Krivochen, 2015c) we have argued that central / terminal coincidence is read off an underspecified P depending on the substantive content of the elements present in the positions of figure and ground, thus being determined componentially at the semantic interface.

[141] Lasnik, Uriagereka & Boeckx (2005: 53) borrow *Conservation Principle* from thermodynamics, and state the following law:

> **1st Conservation Law:**
> *All information in a syntactic derivation comes from the lexicon and interpretable lexical information cannot be destroyed.*

The problem with this law from our perspective is that it makes use of lexical information taken from a pre-syntactic, monolithic, and nonredundant lexicon, which is the norm in orthodox Minimalism (Jackendoff, 2011), but with which we will not work. Moreover, this formulation is limited to linguistic structures, whereas ours is wider in scope.

120) ***Conservation Principle****: information cannot be eliminated in the course of a derivation, but must be instantiated in the relevant interpretative system in such a way that can be read and it is fully preserved.*

If this pre-linguistic conceptual structure was *locative*, then a {P, {D, {√α}}} (i.e., <P, D, √>) construction is to be built in a workspace W. However, we can also derive the better candidate from purely post-syntactic interface conditions, which is another (perhaps less controversial) option. Assuming a strong version of Brody's (1995) *Radical Interpretation Thesis*, each element in a representation must receive an interpretation and each syntactic position must be associated to an interface interpretation; and, we will add, if the element conveys procedural instructions as to how to relate conceptual elements, this information must be represented syntactically, so as not to lose information (and therefore incur a violation of the ConsP). Crucially, *no other element may be allowed in a representation*. We have P in the Type-Array, which conveys relational *locative* procedural instructions, in terms of *central-terminal coincidence* between a Figure and a Ground (Hale, 1986; Hale & Keyser, 2002; Mateu Fontanals, 2002). If we introduced any other token into the derivation, the procedural information conveyed by P would be lost, since there would be no available instruction to relate two (sortal) structures, typically of the type {D {√}}, at the semantic interface: any other element, procedural though it may be (say, the causative functional node *v*), would not be locative, and the information conveyed by the CS would be modified. So, the combination of ConsP / DFI favors the introduction of a P token in W[142]. Crucially, as sortal elements are related by means of predication relations, the space loses its Hausdorff characteristic: the neighborhoods of syntactically related elements in the phase space *qua* sets of points are no longer disjoint, but mutually interfering (that is, the intersection between the neighborhoods of any two distinct points in a syntactic derivation is not the empty set)[143]. In Uriagereka's *kite* model, the intersection between neighborhoods of distinct points *x* and *y* is given by the folding of the space onto itself (see also Uriagereka, 2002b for a related 'warping' proposal); here, the introduction of relational procedural elements in the derivation makes lexical fields interfere. Importantly, the two proposals are not mutually incompatible; rather, they are complementary (with Uriagereka's providing a topological background for our derivational assumptions). The derivational dynamics we describe here follow a self-organized interaction between the generative operation and the semantic component, such that there is no need to stipulate the order of operations in terms of highly specified feature matrices (cf. Stroik and Putnam, 2013; also numerous works in HPSG and other lexicalist models), in terms of Van Geert (2003: 650), '*each step in the process creates the conditions for the next step*', there being a dynamic bidirectionality between generation and interpretation at each step (which we have quantized artificially, per our warning above), with the interpretative systems 'peering into' the workspace to see if the most recently derived unit can be assigned an interpretation, thus constituting a cycle. The process is not strictly deterministic, but results from the interaction between systems, cancelling out derivational paths that violate the principles and / or represent a deviation from

[142] This could be regarded as a *weak formulation* of the theory. A *stronger formulation*, in favor of which we will not argue here –but we have done so in Krivochen, 2014c- would claim that #P# is the *only* token whose introduction in W would satisfy *both* ConsP and DFI. For purposes of the present discussion, which is to define the relevant objects that constitute cycles, both formulations are equivalent.

[143] Topologically, syntactic relations transform a *completely* Hausdorff space into a Hausdorff space, weakening the separation condition. We will come back to this below, in more explicit terms.

the CS guidelines[144]. Now, we obviously want the process to be computationally tractable, to be completed in polynomial time. What is the *halting* instruction for such derivational procedures? A crucial difference from Turing's α-machines (and related controller-free automata) is that the output of a TM is not necessarily interpreted; therefore, unless there is an explicit halting instruction, the machine can go on and on. This problem has also been formulated in the context of a much wider theory, Tegmark's (2007) *Mathematical Universe Hypothesis*. Assuming, as we have pointed out above, that the external physical reality is a mathematical structure (the *Mathematical Universe Hypothesis* MUH) and augmenting that thesis with the assumption that such a structure is defined by computable functions (the *Computable Universe Hypothesis* CUH), there is always the problem of whether those computations that are special cases of mathematical structures and in turn define them (Tegmark, 2007: 19) halt after a finite number of steps (see also Binder, 2008 for a take on the halting problem from the perspective of *computational frustrations*, which are not strictly relevant to the phenomena analyzed here). Tegmark explicitly claims (2007: 27) that '*there is a simple halting algorithm for determining whether any two finite mathematical structure definitions are equivalent*'. However, it is not clear how it would actually help in the implementation of MUH (particularly at the level of Turing-computability). One of the problems, we propose, lies in the assumption that effective computation equals function-based computation. Mathematical structures need not be interpreted (we are not saying they *are not* interpreted, just that they *need not* be), but we have shown that cognitive outputs are mutually entangled, such that the output of a process can be the input of another and that intermediate derivational steps can also be accessed by external systems, influenced by those, and in turn create a feedback loop with those external systems yielding nonlinear emergent properties (see Wegner, 1997; among others, for further argumentation, and of course our **Part II**). This means that a computation being carried out with finite inputs cannot just go on and on *ad infinitum*, because it is permanently 'monitored' by other systems, and in constant interaction with them via bidirectional 'peeking' and information flow: it is crucial to point out that increasing complexity and the construction of 'novelty' are not intrinsic properties of dynamical systems *per se* (Van Geert, 2003: 654); in the specific case of linguistic derivations, these properties are related to the interaction with the semantic and morphophonological systems *and* the externalization of a structure as a string of sounds within a speech community (which provides the *environment* for interactions among subjects to occur, motivating the appearance of new forms). The halting problem, as proposed by Church and Turing (also related to Gödel's Incompleteness results) does not arise when the informational flow is bidirectional, since any system S_x interacting with any system S_y can take sub-derivations for interpretation if those sub-derivations (or 'representations', depending on the computational perspective adopted by the reader) contain only interpretable units for S_x at a derivational point D_x. Within a strong cycle-based system like the one we argue for here, however, *interfaces* as such are dispensable, however: if cycles arise due to physical properties of the system, when manifolds reach a critical dimensionality value, the so-called *computational system* (or *syntactic workspace*) can, as in Survive Minimalism (Stroik and Putnam, 2013) exist *within* the so-called *performance systems*. This means that whatever operation we propose need not apply within a specific, separate component which then *sends* cycles to be interpreted, but

[144] The statistics-enthusiast reader might want to compare the present proposal, which follows the spirit of bidirectional Harmonic Grammar, with the 'lazy Approximate Bayesian Computation' model proposed in Prangle (2014). However, there are crucial differences: since the information flow is bidirectional in our model, there is no need to stipulate the points in which evaluation takes place, nor is it necessary to have an external controller setting up parameters for evaluation. The system self-regulates towards interface convergence, as our model for computation is both interaction-based and nonlinear.

rather, that the interference between lexical perturbations (i.e., structure building) takes place *within* the lexical field, with no need to resort to an independent syntactic component. This makes much more sense in light of the *dynamical frustration* approach, in which local domains, or *cycles*, are the optimal resolution of a tension between opposing tendencies defining each system involved (in language, and for the sake of both simplicity and concreteness, *sound* and *meaning*). All we need is the relevant fields to excite one another, as is the case with fields in their physical sense.

The *halting* rule is defined, for any interaction-based computation, in terms of fully interpretable minimal cycles. If the interface we are interested in is sound at the level of structure materialization (which entails the non-trivial problem of structure linearization), then it seems the relevant condition (or at least the set of constraints that will be higher-ranked) is along the lines of (113 a), following Uriagereka's MSO model. If the focus is set on semantics, then an approach in terms of (113 b) is perhaps the best way to go. In any case, the tension between these components (the *dynamical frustration*) remains at the center of the scene.

Once this framework has been made explicit, considerations about argument structure become more easy to state: the syntactic-semantic skeleton is finite (since the structure *halts* necessarily, due to considerations of both memory limitations and interaction and interpretability), which guarantees a finite number of structurally interpretable positions for arguments to be introduced; the specific relations between those arguments, which are defined as perturbations in the lexical field (relations thus generating interference between the wavefunctions associated with each perturbation, following Uriagereka's 2011 proposal) are determined by the nature of the relational / procedural element that triggers the interference (locative, eventive, causative…). Too many relational elements, although they do not eliminate all the information, generate too much destructive interference, so that the interpretation of structure becomes more and more costly (recall the simplicity-accuracy tension mentioned and analyzed above). Approaching phrase structure from either above or below[145], the number of relational elements, as well as the kind of information they provide to the semantic system, can be derived from the framework we have been laying out so far.

Summarizing the main claims of this section, which translate straightforwardly to the topological perspective on derivations we will work with later on, the wavefunction corresponding to a derivation should be the result of adding the waves (or base state vectors) corresponding to local domains over time, not unlike Uriagereka's proposal involving field interaction in his REM. In this case, we would have a 'slit-like' situation: waves are related to conceptual elements, whereas the slits would be the procedural elements. Why? Because they are the ones that force interference by means of relating elements: an element on its own does not produce any interference, it is just there, as a certain pattern of perturbation in the conceptual space field. So, an 'NP' (a nonterminal recognized as having the semantic and

[145] That is, we can consider the following two perspectives:

a) Starting from nothing, how many elements do we need to generate a legible unit? And, of course, which are those elements specifically? ('from below')

b) Starting with a rich array of functional structure, how much can we eliminate while maintaining the structure-to-content ratio? ('from above')

Optimally, the results should be the same, regardless of the approach.

functional potential of N) is -in a system inspired by Uriagereka (2011)- a perturbation of a lexical field defined by a certain wave function, determined by the eigenvectors associated with it. Then, the wave function of a locative domain is the result of 'adding the waves' of two NPs plus the relational P. Then, the whole locative structure is added to the eventive and causative layers, which in turn introduces another source of waves: the initiator. The rest is just more slits. The more slits, the more destructive interference: in terms of Relevance Theory, more paths to interpretation are closed (e.g., via semantic-pragmatic coercion, see, e.g., Escandell and Leonetti, 2002) if we have more procedural elements, going back to the slit experiment, there is only 'one way towards the light': few elements lead to disorder, more elements (up to a critical point), lead to chaos. But, of course, this is not enough…

3.6 Entropy variation, automata theory, and function-based computation:

In this brief section we will link the discussion of the varying entropy character of linguistic derivations (also discussed in Krivochen, 2014c; Saddy, 2018; and Uriagereka, 2011) with our considerations against the equation *effective computation = function-based computation.* The first problem to tackle is the derivation of cyclicity in formal automata theory: can it be done? Limiting ourselves to the references handled here (that is, without invoking foreign proofs), strictly computationally based definitions of cyclicity, which target finite-state sub derivations can indeed be implemented in any automaton of the kind assumed by Joshi (1985) on up in the CH (i.e., equal to or more powerful than a PDA+), if finite-state units can be encoded within the memory limits of the relevant automaton. We also reviewed a way of implementing cycles of variable computational complexity (represented by means of different Dyck trees) within Medeiros' (2015a, b; 2016) QLSR architecture. However, given the essentially unidirectional character of function-based computation, an informationally based definition of cycle is out of the question, since the automaton has access to no data other than the current symbol in the tape (the same consideration applies to allegedly 'unlimited memory-wise' automata, like the various versions of Turing Machines: recall that the infinite memory tape does not imply that at some point the read-write head has access to anything other than a single slot in the tape (a part of the relevant *m*-configuration of the TM), as Turing himself pointed out (1936: 231). Since the informationally-driven derivation of cycles depends on the computational system deriving structure while both accessing the sound-meaning systems and their legibility constraints, and being permanently monitored by them, a function-based automaton could not compute all aspects of a derivation along the lines of what we have been describing. If cycles are defined upon entropy variation and a critical value for the manifold dimensionality, the point is made stronger, because that very variation in entropy is the result of considering the interaction between different sources of information and a bidirectional flow. As we restrict the phase space to the places it is likely we will find a suitable 'interpretation', our certainty with respect to the localization of the relevant attractor grows, but certainty always comes at a cost: increasing dimensionality, in this case. So, sure, we have a restricted space to look into, but this space extends in more and more dimensions after each derivational step. This process requires whichever system one assumes is in charge of pinpointing attractors to be dynamic enough to either adjust the number of required components in a state vector (which would be quite cumbersome in formal terms) *or* extend the dimensionality of the local unit by introducing a further predicate. Either option is incompatible with a strictly serial and unidirectional procedure, for the one requires dynamical adjustment of the components of a state vector, and the other can have any placeholder replaced by a value, in a non-aprioristic manner.

The implementation of both definitions of *cycle* we have work with requires (at least) bidirectional information flow, CS feeding the syntactic operation, and receiving manifolds after the center manifold has been spammed once the relevant manifold(s) has/have reached the critical dimensional value, either through monotonic, intra-manifold structure building or through the intersection between several manifolds (the latter option is explored by Saddy, 2018; Saddy and Krivochen, 2016a). Now, in which particular sense do interpretative conditions monitor (and condition) computational operations? The CS – LF path seems to us to account for the semantic conditioning: '*whatever you derive, the process must not delete information from CS*', which amounts to the ConsP above. However, we have said nothing about how the morphophonology could condition and constrain formal operations over terminals (other than by appealing to the finite-state nature of linear externalized morphophonological representations). In a nutshell, can syntactic operations be phonologically conditioned? In past works we have proposed they can, assuming a 'generative' approach to syntax, in which the syntactic engine 'puts things together' rather than parametrize and operate over topological spaces without actually 'creating' or 'generating' anything (see also Schmerling, in press for discussion about a non-generative approach to the syntax-phonology interface). In that vein, we formulated the following constraint (which is heavily influenced by the 'deeper structures' of McCawley, 1968) using terminology from Distributed Morphology, and discussed its consequences for comparative syntax:

> <u>*Morpheme formation constraint:*</u>
>
> *We cannot group* [syntactic-semantic] *features in a terminal node (i.e., morpheme*[146]*) if there is no vocabulary item in the B List* [i.e., phonological exponent] *specified enough to be inserted in that node* (Krivochen and Kosta, 2013: 70).

Following standard practice in separationist frameworks, we assumed that a vocabulary item that materializes a certain piece of structure must somehow be specified with respect to the contexts it can be inserted into (see, e.g., Halle and Marantz, 1993). From a syntactic-semantic point of view, the relevant features we had in mind were of the kind [motion], [direction], [event], [cause], [manner], etc. So, for instance, the 'parametric' difference between verb-framed and satellite-framed languages (Talmy, 2000) can be expressed in terms of the constraint above, insofar as, for instance, Path-of-Motion constructions require a terminal to contain the relevant features [manner] and [motion], which Spanish cannot materialize in that context (Mateu Fontanals, 2002). On the contrary, English can only group [motion] and [direction] in roots borrowed from Latin (*ire* compounds, as in *in + ire, ex + ire,* etc., where the P indicates direction, *ire* being a general [motion] V), but not otherwise (and certainly not in a productive manner)[147]. We have argued in past works (including Krivochen, 2012: 70, ff. in connection to complex predications within DPs, as in [an [easy-to-read] guide] and even more complex examples, with whole propositions being incorporated into a DP) that only in a dynamic

[146] '*Although nothing hinges on this terminology in what follows, we have chosen to call the terminal elements "morphemes" both before and after Vocabulary insertion, that is, both before and after they are supplied with phonological features*' (Halle and Marantz, 1993: 114). Also Noyer (1999), who slightly modifies the definition: '*In DM* [Distributed Morphology], *the term morpheme properly refers to a syntactic (or morphological) terminal node and its content, not to the phonological expression of that terminal, which is provided as part of a Vocabulary item. Morphemes are thus the atoms of morphosyntactic representation*.'

[147] The argumentation here is necessarily oversimplified for reasons of space and scope, but see Acedo Matellán (2010) for a detailed theoretical and empirical discussion.

bidirectional model in which the morphophonological component can access the syntactic workspace can the relevant filtering take place, namely, the constraint above only makes sense in order to filter out structure that cannot receive an interpretation at the 'monitoring' system (note that we are interpreting systems more as harmonic sets of constraints than as sets of principles, which is closer to restrictivist desiderata of the kind advocated for in Harmonic Grammar and OT). In this sense, a terminal that grouped [motion] and [manner] would not be phonomorphologically interpretable in Spanish, which a system that self-regulates takes into account in order to prevent such a terminal for ever being assembled (what we could call 'learning'). However, we need not be limited to 'features', since, as we will see in **Section 4.5.1**, there are constructions which require an *atomization* of structure, that is, the realization of a *nonterminal* of arbitrary complexity assembled in a workspace W_x as a *terminal* for the purposes of further computations at W_y for $x \neq y$ (our use of the term 'atomization' is different from Uriagereka's 2015, insofar as our atomization process is based on a manifold reaching critical dimensionality and squeezing into a lower dimensionality space, not on a representational decision to group nodes). We will expand on the consequences the adoption of a model for cognitive and derivational dynamics in which structure is built in cycles of varying entropy and dimensionality has for locality conditions in language, and more in general for the development of models of neural dynamics.

The crucial point in this section is that, for different reasons, the oscillatory model of cyclicity we propose and will expand on in **Sections 3.10** and **3.11** under field-theoretic assumptions (inserted in a more general model about mental computations) cannot be wholly implemented by means of any single function-based automaton inscribed in the Chomsky hierarchy due to the inherent limitations of function-based computation. Moreover, and perhaps more specifically focused on Y-model based theories, we argue that operations in one component or level of representation are not independent or isolated from conditions applying at another level of representation. Rather, and as we have argued in previous sections, the notion of interaction and bidirectional informational flow must be invoked to account for derivations and interpretation in a dynamical non-linear system like the one we propose to model not only natural language, but cognitive processes in general. Thus, it is natural to say in this framework –using traditional terminology- that syntactic rules are phonologically and semantically conditioned (even though there are no 'syntactic rules' as such in our architecture); any phenomenon, structure building and interpretation alike, is an interface phenomenon and should be formulated as such.

3.8 Refining MaxEnt

In previous sections we introduced the concept of derivational entropy (based on Uriagereka, 2011; Saddy and Uriagereka, 2004; and expanding on discussion in Krivochen, 2014c), and how the interaction between the computational system and the relevant interfaces generates a non-linear dynamics (insofar as the output is not proportional to the input) in which the values of entropy vary cyclically. Summarizing the argument, since all elements of an array are equally likely to enter the derivational workspace at $t = 0$, we would expect the value of entropy to be maximum. Equally, once a cycle is completed we would expect entropy to go back to maximum again, since there are no *a priori* guidelines as to which element(s) to manipulate next. We will argue now that, even though this picture is essentially correct (within the context of the present work, needless to say) at a certain level of abstraction, namely, from a strictly formal point of view, things get more complicated in a scenario closer to the 'real world'.

A direct consequence of the simplified picture we presented above is that cycles are completed and wiped out of the workspace, without there being any trace of their existence: this is why the uncertainty at the moment of starting a new cycle is Max*h* (i.e., $h = 1$). This is what we would want to refine now: there are indeed remnants of previous cycles affecting the current dynamics of the system. There are both theoretical and empirical arguments against the simplified version of variable entropy starting at $t = 0$ from complete disorder ($h = 1$), which we will consider in turn:

a) Theoretical arguments:

Theoretical / conceptual issues are related to the desideratum that derivations be somehow globally 'cumulative' in a non-trivial sense. This means that we would like to maintain the essential insight of Uriagereka's REM, but at a global derivational level (i.e., not at the application of every structure building operation, as we have argued above). Uriagereka (2011: 23) claims that '*the entire point of linguistic combination is to reduce uncertainty*', with which we agree. However, the immediately following strengthening of the hypothesis is what we question:

> *As combinations proceed, we gain certainty; the ideal (final) combination between fields should reduce uncertainty to zero. We may see this procedure as another way of achieving representational stability, and we may take the REM to be the way the language faculty has to ensure that only objects whose certainty has either being achieved or stipulated* […] *can be handled by the interpretive components*. (Uriagereka, 2011: 23)

Certainty always comes at a cost, and we follow the physical insight of Heisenberg's *Uncertainty Principle*[148] in that absolute certainty is not attainable: because the commutator of x and p does not equal 0, both magnitudes cannot be measured simultaneously. While we are fully aware that the Uncertainty Principle was formulated having in mind position and momentum of a particle, a wider interpretation (not unfaithful to the original formulation) acknowledges that, as we said above, certainty always comes at a cost. Representational stability, as Uriagereka points out, is desirable; however, we do not see it as something to be 'achieved' (in any teleological sense), but only as part of the dynamics of the system, which oscillates between stability and instability. Both are essential to the non-linear dynamics of the system, and derive problematic aspects of the interaction between systems with different computational and topological properties (a clear case, which we will analyze below, is linearization of structure, which implies dimensionality reduction as well as drastic changes in the morphology –in the non-linguistic sense- of the relevant object).

A somewhat disconnected argument in favor of taking $h = 1$ as an asymptote rather than as a true state of the system comes from a perspective related to brain dynamics. Assume we have a statistical model that assigns mental states an activation probability between 0 and 1 over discrete time spans (a simplified hypothetical scenario built by Spivey, 2007: 43, ff.). Even assuming mental processes were digital, computation was function-based, and statistical models

---

[148] As a reminder, the Uncertainty Principle is enunciated formally as follows: $\Delta x \Delta p \approx \frac{\hbar}{2}$ (uncertainty in position times uncertainty in momentum is of the order of Planck's reduced constant -$h/2\pi$- over 2). In this work, h always denotes *entropy* unless explicitly indicated otherwise, as in this case; where h denotes Planck's constant.

were explanatory (none of which makes any sense in the context of the present work), the problem still remains of why the system reacts and proceeds under sub-optimal conditions (i.e., with probabilities of less than 1). In this respect, Spivey (2007: 47) claims that

> *Under realistic circumstances, changing perceptual input and continuous motor output would generally prevent the system from ever achieving 1.0 activation for any mental state.*

The reason is that response thresholds (cognitive, motor, etc.) are always lower than 1.0, that is, we make choices on the basis of incomplete information and with less than absolute certainty (a point that we made above, both in connection with interactive-based computation and the resolution of a dynamical frustration between velocity in response and accuracy in response). This allows the system to proceed swiftly 'from near-attractor to near-attractor' more quickly and efficiently, in response to new multimodal inputs and interactions (using Schöner's terms). This is consistent with the idea that 0 and 1 are asymptotes rather than legitimate attractor points the system can 'go into' (Spencer et al., 2009: 109). Note that this argument also prevents activation probabilities of 0.0, since

> *In real life, we are very rarely in situations where we have nothing at all on our minds, and then suddenly a stimulus impinges on our minds, instigating a new trajectory through mental space* (Spivey, 2007: 47)

In other words, we do not ever start processing from scratch; we never stop processing the world around us, and under interactive assumptions, outputs and inputs are entangled, which means that the output of a process is the input of another in a continuous ongoing process (see also Milner, 2004: 6-7; Goldin and Wegner, 2008: 2), something both behaviorists and cognitivists overlooked (Spivey, 2007: 48). Spivey's model will reappear below, but for the time being, we would like to borrow this idea, which also cautions us about the apparent usefulness of statistical / probabilistic models in a framework like the one presented here. This is not incompatible with the idea presented above about rest states, for those states, as we specified, do not equal 0 (i.e., the eigenvector $|n\rangle$ representing the ground state of the field is *not* $|0\rangle$), the resting state of cognitive dynamics does *not* correspond to a 0.0 activation (rather, the vector $|0\rangle$ would be no activity at all, which corresponds to being dead).

The idea that there might be 'something left' from previous cycles (a property that arises in L-grammars in the form of 'residues' and 'extensions', as pointed out in Saddy, 2018; Saddy & Krivochen, 2016b, and which was illustrated in **Figure 7** above) can also benefit from Feigenson's (2011) considerations about the ways in which the working memory represents objects (a theory we have already introduced). If it is possible to 'atomize' a chunk of structure for the purposes of further computations (in formal terms, turning an object $|\alpha|$ of arbitrary complexity into a terminal $\#\alpha\#$, with a concomitant loss of dimensionality through squeezing via the center manifold), that would be akin to taking a number of individual objects and conceptualizing them as a *set* (examples of such an operation are memorizing telephone numbers), which

> (…) *manages to evade the three-item limit of WM while still preserving access to individual representations of the sets' contents. This appears to rely on the hierarchical reorganization of items within memory* (Feigenson, 2011: 16).

Even if the 'three item' limit could be challenged empirically, the overall proposal seems plausible to us, and provides an interesting platform for studies of derivational cycles in connection to memory issues. For the time being, we will simply say that, if the human mind is capable of performing operations of the kind Feigenson proposes with representations (which in turn requires representations to have a certain flexibility in their structural format, something we have argued for in Krivochen, 2015a and Krivochen and Schmerling, 2016), additional stipulations would be required to support a claim that these operations do *not* apply to linguistic representations because of some putative specific property. If the generative (structure-building) engine is not specific to a faculty (*contra* Hauser et al., 2002 and much subsequent work within 'biolinguistics') the computational tools required to perform the structural operations over objects Feigenson puts forth are thus available all throughout the mind, a perspective we consider highly desirable.

In the scenario we have sketched, it is very unlikely that entropy keeps going back to Max*h* 'as combinations proceed' cycle after cycle (note that we are making no claim with respect to the properties of the combinatory engine in terms of the elements it manipulates or how it does that, see Krivochen, 2015a, b for details of our take on the issue): that, we would expect from a 'dumb system' with no operative memory, despite having a possibly infinite tape (an automaton, in the sense we have reviewed). A system with a finite, but nonzero, operative memory (a.k.a. working memory / episodic buffer / RAM…) would not just wipe out each and every trace of previous cycles if derivations are indeed something close to 'incremental'. Given the compelling nature of conceptual arguments in favor of keeping track of completed cycles, are there empirical arguments as well to support such a position? We believe there are, and we will now proceed to present some of them.

b) Empirical arguments:

In order to support the argument that there are traces of previous derivational cycles in the workspace, we should need evidence of phenomena involving cross-cycle dependencies. Moreover, those phenomena, if existent, should not be reducible to operations applying in a 'punctuated' fashion cycle-by-cycle (as would be the case of Wh-movement in orthodox Minimalism, see Abels and Bentzen, 2009 for a generative perspective; Krivochen, 2015b for a relativized cyclic proposal involving Wh- interrogatives, parasitic gaps, and multiple-gap constructions). Let us consider an example like (121)[149]

121) Every man and woman who went to a place wants to see it again

If the proposal about structure building made in Krivochen (2015a) is along the right lines (even in its general assumptions), then we are in the presence of a computationally mixed phrase marker, in which we can find dependencies belonging to different levels in the CH co-existing within a phrase marker. The example is complex in its own right, so we will focus on two well-defined aspects of it: quantifier scope and agreement. Assuming the semantically-based structure for coordination proposed in Krivochen (2015a: 551, ff.) and Krivochen and Schmerling (2016), the subject presents a universal quantifier [every] having scope over a finite-

[149] This example, of course, owes much to historically prior examples like Peter Geach's 'donkey sentences', but we have modified some aspects, like the coordination in the subject, to strengthen our point. However, a pure example of 'donkey anaphora' would work as well, see e.g. Heim (1982). Or any 'Bach-Peters paradox' (Bach, 1970; Karttunen, 1971).

state unit [man and woman], which is derived via *n*-ary concatenation. This unit is actually the result of taking the objects [man] and [woman] and mentally representing them as a finite-state unit, an '*object*', following Feigenson's (2011) terminology. This unit is also modified by a restrictive relative clause RRCl (in traditional descriptive terms, which will do for the purposes of the present discussion) that select a subset of the potential *denotata* for [man and woman], namely, those [who went to a place]. Apart from the quantificational ambiguity in [a place] (which is also present in [every man and woman [RRCl]], and defines two possible interpretations, roughly a collective reading and an individual reading[150]), the whole clause enters an interesting relation with the universal Q: is it the case that [every] modifies [man and woman who went to a place] or does the RRCl modify [every man and woman]? So far, phenomena seem quite local. Here is when the [it] enters the game: clearly, there is great structural distance between the pronoun and its referential antecedent. Let us graph the relevant objects using familiar square bracket notation for constituency:

122) [[every [man and woman] [who went to a place]] [wants to see it [again]]]

In (122), each bracketed object is a 'command unit', in Uriagereka's (2002a) terms, which means that c-command relations within that unit are unambiguous, and its structure can be exhaustively represented by means of finite state processes (Uriagereka, 2002c: 151; 2012: 53). Because it only displays a single kind of computational dependency, each of those is a *cycle*. Relations *among* these units go beyond finite-state computability (Uriagereka, 2002a, c; Krivochen, 2015a), and because the system's growth has changed, computation is no longer monotonic. Now consider again the relation between [it] and its antecedent [a place]: the pronoun, which belongs to a command unit, establishes a relation with an element two levels embedded. This relation is twofold: not only is there a semantic relation regarding referentiality, but also a morphological relation, which surfaces as number agreement (cf. [*every man and woman$_j$ who went to a place$_i$ wants to see them$_{i/j}$ again], subindexes indicating the endophoric relations), but it is not deterministic: the singular morphology is also acceptable, in the collective reading. If the proposal about binding we made in Krivochen (2015c), summarized in **Section 1.2** above is correct (or even plausible), then the pronoun is a variable, whose phonological exponent depends on the relation it establishes with another element in the structure: such a relation goes beyond the limits of the strict finite-state unit containing [it]. How is this possible? Consider Feigenson's concept of *set*, a grouping of entities which allows access to the relevant elements, even though these are conceptualized as a unit for WM purposes: it is plausible that an object of the kind of Feigenson's *sets* remains active in the workspace as a 'trace' of previous cycles, optimally limiting the existence of these traces by the same means by which the limits of WM (or the 'episodic buffer') are determined. We will also see that the persistence of remnants of previous cycles as proposed above is essential to account for instances of binding that seem to defy non-stipulative attempts of explanation, and will also prove useful when considering islandhood and Wh- dependencies (all of which have been

[150] These readings could be expressed in terms of 'strong' (every / each) vs. 'weak' (all) quantifiers. There, we also hypothesized that weak Q are used only if the possibility of using the strong Q is for some reason not available, thus generating the (conventional? Generalized conversational?) implicature that the speaker *cannot* use the strong Q because it either does not apply or because the speaker lacks the relevant information to make a statement using the strong Q (not unlike Grice's 1975: 49 examples of relative hierarchy among maxims).

subsumed to *operator-variable* relations in MGG). These will be topics we will cover in detail below in **Section 4**.

It seems that there are both theoretical and empirical arguments in favor of a certain amount of 'globality', which includes cross-cycle relations. How does this impact the entropy dynamics? Let us describe the behavior of entropy and dimensionality as a function of time.

If there are 'traces' or 'remnants' of previous cycles in the workspace, which can be projected back to previous derivational stages (again, following the L-grammatical formalism analyzed in Saddy, 2018; Saddy & Krivochen, 2016b), then it is not possible that entropy goes back to the maximum level after the completion of a cycle, since the next cycle does not start 'from scratch' (a point also made by Spivey, 2007). This means there is a difference between Max$h$ (the maximum entropy level) and the entropy level at the beginning of a cycle. We will call this difference Δh. If derivations proceed in a cumulative manner, not only locally (within cycles, with each application of the structure building operation to active objects {a, b, …, n}) but also globally (i.e., across cycles, which is roughly Uriagereka's 2011 idea of global entropy reduction), then it is reasonable to assume Δh is not constant, but increases as derivational cycles are completed and there are more 'traces' of what the system has already computed in the workspace. We shall say that Δh increases per cycle by a factor n, where $n\Delta h$ < Max$h$. We can now state the following hypothesis (which we will call 'Hypothesis 1' for ease of reference) about the dynamics of the derivational system:

Hypothesis 1: There cannot be a derivation for which Max$h$ – $n(\Delta h)$ = 0.

Corollary: $\frac{dh}{dt}$ cannot be 0 for $t > 0$.

Let $f(t) = h$, $f(t') = h'$ for all t, $t' = t + \Delta t$, such that $h' < h \leq$ Max$h$

In plain English now, derivations display cumulative effects, meaning we do not start from square one (or zero) at every derivational point: the phase space where attractors for our dynamical system can be found gets smaller and smaller. **Hypothesis 1** captures this intuition, the idea that with every derivational step, entropy is reduced at the cost of introducing a new element in the working area, modifying its topological properties (and consuming energy). Since derivations display cyclic properties, the same cumulative informational effects can be extended to the cycle level: no cycle within a derivation starts at the same level of maximum entropy as the previous cycle; we have won something in the meantime, and that something is *certainty* with respect to the location of interpretative attractors. If Max$h$ – $n(\Delta h)$ was equal to 0, that would mean we would be in the presence of a non-cumulative process, in which each cycle takes us back to a state of near complete uncertainty. Moreover, as we will see below, global semantic effects would not be attainable, for the workspace would have to be wiped clean of variables, operators, etc. (otherwise we could not get back to Max$h$). There are, as we will see, theoretical and empirical arguments in favor of **Hypothesis 1**, or some equivalent formulation.

The scenario is (hopefully) clearer now: each derivational cycle displays local entropy reduction at each step, but also contributes to the reduction of global entropy by having Δh increased by a factor n. The question now is: what is n? We will assume (for reasons that hopefully will be apparent below) that n is related to the dimensionality of the manifold we are dealing with (actually, building!) at a particular derivational point.

The graph of a dynamical nonlinear system like the one we have been describing so far is globally a curve asymptotically tending to zero entropy, and locally, a discontinuous cosine function non-differentiable at integer values of t (this last bit is something we stipulated, due to the notion of cycle we have defined, it follows that the function is discontinuous, but not that the points at which it is discontinuous –thus, non-differentiable- are integer values of time). Our plane is the phase space comprising all possible attractors within the ultrametric lexical field, and since there is no distance metric (i.e., no $d(x, y, \ldots n)$ function) over points or a set thereof (attractors) in the lexical field, they are all accessible, in principle, at the starting derivational point. The subsequent development of a derivation, which relates lexical attractors, is defined by means of ODE, which give us the slope of the rate of change of h over t. What does the integral do, then? The integral of a function is (informally!) the area below the curve, and if the whole plane is the complete phase space, the area below the curve defining the evolution of the system over time is the phase space to which we have access. We will see that the integral defines some very interesting dimensional properties with respect to the curve: first and foremost, we have to note the fact that the integral of a function defines a phase space that is of a dimensionality higher by 1 than that of the function. We will just leave this here, to return to it below.

In the following section we will explore the mechanisms via which the function reaches critical values that determine its discontinuity at certain points. We will attempt to derive theoretically and empirically relevant conclusions from the framework we have been sketching here, related to argument structure, limits for the dimensionality factor, and, most importantly, the role of cyclicity in materialization (in MGG terms, the operation Spell-Out) and the emergence of 'dynamic flatness' (Lasnik and Uriagereka, 2012: 21) from the oscillatory system.

3.9 Squeezing n-dimensions through Spell-Out

If we build up the dimensionality of syntactic representations by means of argument-predicate relations (as proposed in Krivochen, 2015a; partly based on Uriagereka, 2002b), linearization could proceed in a specular fashion: there are both empirical and theoretical arguments in favor of cyclic approaches to linearization, the essence of which (despite mostly superficial differences) being that a derivation is chunked and linearized not as a whole, but proceeding chunk by chunk. Although there is relative consensus with respect to the necessity of a mechanism to implement cyclicity in computations, the specifics are far from agreed upon, including the size and identity of the relevant chunks (alternatives range from *v*Ps and CPs in Chomsky's *phase theory*, to Markovian sub-derivations in Uriagereka's MSO model; including Grohmann's 2003 quasi-cartographic *prolific domains*, Marantz's 2007 phases-within-words proposal; also Bresnan, 1971, among many others). In this work, we have proposed our own version of what cycles might look like; now, we will provide a rigorous mathematical-physical framework for their implementation. The relevant thesis in Krivochen (2015a: 560) was that:

123) *If a predicate is to have scope over a referential (either sortal or eventive) variable, the number of the predicate's coordinates in the mental working area properly contains the number of the arguments' coordinates.*[151]

---

[151] The reader might find it interesting to contrast this approach with that of Mori (2005), who claims that '*each layer of inductively generated structure is dimensionally and hence qualitatively, different from the previous, less complex ones, in the Dimensional Theory. In this framework, what syntax does is just to crank up structures of varying orders of complexity, according to a certain format that involves what is customarily referred to as 'theta roles*''. Her focus is the derivation of lexical verbs, and the determination of a hierarchy of theta-roles (and *Aktionsarten*) based on dimensionality considerations. An

‘Arguments’ are perturbations in a lexical field, whereas ‘predicates’ are the elements that, in a particular derivation, make those fields interfere. For instance, let us see what a predication relation over a sortal entity defined in two dimensions would look like:

124) a) N = (x, y)
b) V = (x', y', z)
b) V(N) = A × B = (x, x') (y, y') $(Ø, z)$[152]

And so on. For instance, if a category X (X ≠ V) should take (V(N)) as an argument, it would have to be defined in (x”, y”, z’, w), introducing a further dimension represented by the w axis. As the reader may have noticed, the predication relation involving coordinates is expressed as the cross product of the coordinate sets involved, a move that makes sense if we are dealing with fields:

> *In Dirac notation (Dirac, 1958: 16, ff.), operations over vector fields are cross product vectors, such that the inner product of a bra and a ket vector* $\langle\alpha \mid \beta\rangle$ *is* $\langle\alpha \mid \times \mid \beta\rangle$.

If this globally cumulative approach to derivational dynamics turns out to be on the right track, many essential properties of the structure-building operations and conditions over its outputs will follow from the very architecture of the cognitive system: conservation, growth patterns, and computational flexibility (in the terms we have managed in Krivochen, 2015a), among the most important. What (123) gives us is a principle to conceptualize how manifolds grow and can reach a critical dimensional value, which will be essential in the discussion of the oscillatory computational mechanism we will develop below.

The approach we are advocating will allow us to define very precisely the scope of multiple modifiers over an argument, as in iteration (e.g., if n As modify an N, but none of them has scope over another, then they are all defined by the same *number* of coordinates in the cognitive workspace, the respective values for the individual components varying as these values represent the position the relevant lexical vector points to within the field, which translates to a finite-state representation of the kind we have proposed for iteration and some instances of coordination in Krivochen, 2015a; Krivochen and Schmerling, 2016). This approach thus effectively tackles the problem of uniform phrase structure grammars assigning ‘too much structure’ to natural language strings, noticed by Chomsky (1963: 298), Postal (1964: 24), and Lasnik (2011: 357-358) among others. At this point in the development of our proposal, it is not really crucial whether N is a defined as a pair, a triple, or a single coordinate, just that if N = (n), the predicate that takes N as its argument will be defined as (n+1). In other

obvious difference from the proposal outlined here (which is underspecified with respect to the nature of the symbolic objects being manipulated, unlike Mori’s) is that Mori’s take is based on the machinery assumed in the Minimalist Program, including a belief in the existence of a Faculty of Language and its initial state, UG. Features, and operations over them, are also central to his proposal. The formulation of the Dimensional Theory in her thesis is also dependent on the notion of *event* (with respect to which we have remained agnostic), and resembles the Derivational Theory of Complexity insofar as there is a connection between increasing number of participants, increasing dimensionality, and increasing complexity. We have rejected the ‘revamped’ DTC advocated for in Marantz (2005) and Ohta et al. (2013), for its essentially function-based character and aprioristic take on phrase structure and syntactic complexity.

[152] The result could have been expressed in matrix notation as well. The reader interested in linear algebra may wish to try the exercise.

words, *for n dimensions*[153] *of α, f(α)* [a predication taking α as its argument] = *n+1* (for a different perspective, see Uriagereka, 2002b: 296, who proposes that dimensions can 'warp' in order for the manifold to get to a higher-level dimensionality; similar arguments are used in string theory, with dimensions stretching and folding in specific topological configurations). Our system predicts that each structural 'generation' generated via monotonic concatenation will have a higher dimensionality than the previous one(s), provided that (123) holds, yielding a locally cumulative process working by cycles as we have predicate-argument relations established within a derivational current. In this system, it is not clear that dimensions 'warp' in the cumulative structure building process[154], even though it is possible to conceptualize geometrically the process of dimensionality increase in structure building in terms of a partial tessellation of a space of dimension *n* by means of geometrical constructs which can fold onto themselves to fill gaps and yield a complete tessellation of a space of dimension *n*+1 (as is the case, for instance, if we try to completely tessellate a 2D space with regular pentagons: there is a 36° opening that is not covered, but it can serve to fold the pentagons and create a 3D object). For the time being, we will keep developing the idea that *n* + 1 properly contains the relevant object described within *n*, such that a predication relation can be established[155]; note that the warping mechanism needs to be triggered somehow in linguistic derivations, and the present proposal has the advantage of identifying the kind of terminal nodes that make a manifold increase dimensionality (that is, *predicates*, or otherwise *relational* elements). The 'warping' mechanism does not play a role in structure linearization, in the terms we are working with here (but see Uriagereka, 2011: 49 for a proposal within the 'kite model', quite reminiscent of so-called 'cusp catastrophes' in catastrophe theory, see Zeeman, 1977), as it is not clear how the warping is actually implemented in a dynamical system in terms of timing and triggers. We are crucially assuming, with Uriagereka, that the linearization and materialization of an $\mathbb{R}^n$ manifold requires a dimensionality reduction, which we have referred elsewhere as a 'dimensional flattening', and Uriagereka interprets (correctly, in our opinion) as a 'Markovization' of the relevant structure (see also Lasnik and Uriagereka, 2012). The oscillatory system also derives an interesting property of nominal and verbal domains: only in the nominal domain the operators over state vectors may commute; in the verbal domain this is not possible. In Spanish, for instance, the order of postnominal adjectives is not fixed, such that [un libro grande, rojo, pesado] denotes the same object than [un libro pesado, grande, rojo]. In this case, the operators

---

153 We assume here that each dimension is represented by a different axis.

154 Crucially, we are not ruling the warping mechanism out, just questioning its applicability to structure building processes: it is possible that the syntactic manifold warping process plays a role in structure mapping operations, which we will explore below when dealing with locality conditions in the form of *islands*.

155 In both the 'warping' systems of Uriagereka (2002b) and Mori (2005), '*what the 'warps' system attempts is something more ambitious: it is because of the orders of complexity in the syntactic system that the particular ordering we encounter in the lexical semantics is what it is.*' (Mori, 2005: 254). Our take on the issue is the mirror image: (narrow) syntactic structure follows semantic structure, from CS to LF. In this sense, whatever ordering we encounter in lexical semantics follow from CS-LF determinations, which the narrow syntax must follow (in that sense, we are closer to Mori's characterization of Generative Semantics, which is by the way not quite precise historically), not the other way around (which is arguably the position to adopt from a Minimalist stance). The 'warp' proposal '*is the anti-generative semantics, as it attempts to have semantics follow from syntax, not the other way around*' (Mori, 2005: 257), the latter (syntax following from semantics) being our own take on the issue. If syntax is underspecified enough to apply all throughout the mind, and just blindly builds structure, our -quite personal- version of generative semantics follows. It all depends on what the reader thinks 'syntax' means.

(the adjectives) are said to *commute*. In the verbal domain this is clearly not the case: as shown in Bravo et al. (2015), different orders of auxiliaries within a chain derive different meanings which are not related: [está teniendo que trabajar] has different truth conditions and semantic interpretation from [tiene que estar trabajando]. Operators in the verbal domain *do not commute*. Interestingly, Dirac (1958: 26, ff.) points out that linear operators over *bras* and *kets* correspond to dynamical variables affecting the system defined by the vectors: these are subject to an algebra in which commutativity does not hold. The kind of relations established between auxiliaries and verbs on the one hand, and adjectives and nouns on the other (which phrase structure grammars tend to conflate) are very different from the point of view put forth in the present thesis: we have allowed finite-state relations for adjective iteration within the NP, but the computationally mixed character of auxiliary chains (which was argued for in Bravo et al., 2015; García Fernández et al., 2017) makes it impossible for commutativity to hold for the relations between extended projections of lexical auxiliaries and those projections and the lexical VP (much like in Dirac's discussion). Much research pending, it seems clear that the perspective we argue for here provides us with a richer array of tools than most current narrowly syntactic theories, particularly those based on monotonic phrase structure (more generally, 'immediate constituent') grammars.

A fundamental step in explaining the oscillatory dynamics of syntactic computation, then, is to achieve not only the explicit formulation of a cumulative structure building system, but also of a mechanism to dynamically reduce the dimensionality of the manifold for the purposes of further operations and embedding of a manifold in a higher-dimensional structure (e.g., Nash's 1954 Imbedding Theorem for continuously differentiable manifolds), a process that derives from the architecture of a system that oscillates between high-dimensional ultrametric spaces and low-dimensional metric spaces when the system is externally perturbed (Saddy, 2018; Saddy and Krivochen, 2016a). This oscillatory and continuous model, we argue, cannot rest on uniformity assumptions, be them computational (e.g., all phrase markers are derived in the same way and have the same formal properties regardless of their content and interpretation) or topological (i.e., regarding the properties of both the spaces in which operations apply and the objects thereby derived). In this respect, when contextualizing our proposal within the existing literature we have to take into consideration that recent developments of Uriagereka's theory are based on the (uniform?) ultrametricity of the syntactic space -something we have relativized above, when considering syntax as a topological transformation over completely Hausdorff spaces- as well as topological warping operations over phrase markers. In order to introduce a possible set of procedures for periodic dimensionality reduction here, we will adapt an idea from Schöner (2009) about the properties of unstable *n*-dimensional (neural field theoretic) dynamical systems, the so-called 'Center Manifold Theorem' (see also Guckenheimer and Holmes, 1997, Theorem 3.2.1):

> *When an attractor of a high-dimensional dynamical system* […] *becomes unstable there is typically one direction in the high-dimensional space along which the restoring forces begin to fade; while in other directions, the stabilization mechanism remains strong. The one direction along which forces fade spans the center manifold.* (Schöner, 2009: 9)

Haragus and Iooss (2011: 34), within a more rigorous mathematical framework, also claim that center manifold theory reduces the dimensionality of a problem, as center manifolds arise near critical points in a dynamical system (in this case, the critical value is 0, but in a spin glass configuration there can be many such critical points):

*Starting with an **infinite** dimensional problem* […], *the center manifold theorem reduces the study of small solutions, staying sufficiently close to 0, to that of a small system with **finite** solutions* (Haragus & Iooss, 2011: 34. Our highlighting)

Also Carr (2006) defines the CMT as a way of simplifying otherwise very difficult (or downright unsolvable) problems involving multiple dimensions in dynamical systems, again, approaching critical points:

Centre manifold theory [sic] *is a rigorous mathematical technique that makes this* [dimensionality] *reduction possible, at least near equilibria*.

In sum, while some properties of complex, *n*-dimensional complex systems might be difficult to solve with ordinary differential equations, solutions in a center manifold can always be described by a finite-dimensional system of such equations, which is indeed an advantage from both theoretical and empirical perspectives.

Recall that the function corresponding to a syntactic derivation is discontinuous due to the appearance of cycles which represent a temporary resolution to the dynamical frustration between semantic and materialization requirements, as we have argued above. Given a function which is discontinuous at a point *p*, the derivative $\frac{dy}{dx}(p)$ is undefined, and thus the system loses stability (causing an attractor to disappear or fade). Crucially, we will propose here that *p is the point where a cycle ends, thus motivating the assumption that a cycle is spelled-out* (i.e., linearized and materialized) *once completed, there being a discontinuity* (a 'hiatus' in Uriagereka's terms, a 'catastrophe', in Zeeman's 1977 terms) *in the dynamics of the system* (and, more graphically, in the function that describes such dynamics), *to be solved in a lower dimensionality*.

The direction (i.e., the axis) along which the attractor fades define the low(er) dimensional *center manifold*, a number of orbits whose behavior around the equilibrium point is not determined by either attractors or repellors, but is the direction along which the system evolves more slowly (as restoring forces are weaker), which means the rate of change $\frac{\Delta y}{\Delta x}$ is smaller. Moreover, we also need a way to get back into the ultrametric space to keep the system running, which we achieve by means of dimensionality increase via predication relations within a cycle (Krivochen, 2015a) or the establishment of dependencies between non-monotonic cycles (Uriagereka, 2002a; Saddy, 2018); both can be unified in terms of a dynamically developing grammar attempting to exhaustively tessellate a space and folding onto itself on specific nodes which relate monotonic structures (relational nodes, as we have said before). In either case, some aspects of the low dimensional manifold extensions, squeezed into metric space via CMT are operated over in that space via Chomsky-normal operations, while –if the interface between ultrametric and metric spaces is an orthogonal L-space, as suggested by Saddy, 2018; Saddy and Krivochen, 2016a, b- the copy of the system's previous dynamics existing in the metric space by virtue of the interface's own dynamics (see **Figure 7** above) is eventually pulled back up to the ultrametric space, to enter into further relations there. When manifolds intersect, their core properties are metricized by the CMT, in a procedure we will analyze more explicitly when considering Feynman's *Sum Over Paths* proposal and its relevance for the interaction between manifolds in vector spaces.

For this argument to work, however, we first have to make sure we are in fact dealing with (continuously) differentiable orientable manifolds: are objects that are derived by means of syntactic mechanisms indeed manifolds? Consider the argument in favor of the increasing dimensionality of derivations by means of predication relations in Krivochen (2015a), such that *for n dimensions of α, f(α) = n+1*. This system predicts each structural 'generation' containing a monotonically introduced predicate (if we quantize the process arbitrarily, as pointed out above) will have a higher dimensionality than the previous one(s), provided that our **Hypothesis 1** above and our claim (120) hold. This ensures that a manifold does not increase its dimensionality if we have only flat relations (as in iteration, which we have modelled as a Markov chain), but it does if what we have is a higher-level dependency (Types 1 / 2) involving relational elements. Our hypothesis can be seen as a reworked version of the warping one, if (120) is a principle to identify the points in which the phrase marker folds and warps thus increasing its dimensionality. A derivation would expand in a multidimensional space in real time as it is built until reaching a critical dimensional value, then, an attractor would fade, which determines the direction in which the system 'squeezes' into a lower-dimensionality space. How to determine this point? It is important to note that the determination of the critical value for cognitive manifolds is an *empirical* question, which we address in the *Intermezzo*. We said that this critical value is the point at which the function is discontinuous, but that alone would make a circular argument. The question is: why should the function be discontinuous at all (and, more specifically, discontinuous in the way we suggest here)? The answer, in our opinion, lies in the concept of *dynamical frustration*, which makes its glorious reappearance here.

Recall that we have followed Uriagereka (2011, 2012, 2014) in claiming that there is a fundamental tension in the architecture of language as a cognitive system between conceptual and morphophonological trends, which tension is one of the main features of dynamical nonlinear systems (Binder, 2008; also Holpert, 2008). More specifically, we have argued in favor or a *scale frustration*, in which global trends favor conceptual conservation principles (see also Lasnik and Uriagereka, 2005: 53; Krivochen, 2011: 53) from the Conceptual Structure[156], which is non-linguistic, through its linguistic instantiation all the way to the so-called 'Logical Form' (more in relation to the 'enriched proposition' Relevance-Theoretic LF than to the narrowly first-order logic representation that is customary in Generative Linguistics, as reviewed in Hornstein, 1995); whereas local trends favor cyclic aspects of computation, targeting minimal domains (See Ross, 1967; Chomsky and Halle, 1968; Bresnan, 1971 for some early examples of this idea). We also said that we reject the traditional Y-model in which there is an autonomous syntactic component which operates through monotonic combinatorics over discrete elements and *sends* information to the interfaces, since (a) on both empirical and conceptual grounds, we reject the autonomy of syntax and its alleged 'specificity' as there is structure (and therefore, *syntax*) all throughout cognition; and (b) we argue for a dynamical model in which the computation –which we will conceptualize as the (*para*)*metrization* of the relevant fields- takes place *within* the sound and meaning systems, and therefore any syntactically maximal / semantically minimal unit is 'isolated', as it triggers the fading of the center manifold in a particular topological space and is interpreted (instead of having the 'performance systems' Sensory-Motor and Conceptual-Intentional just passively *receiving*

[156] In turn, the conceptual structure is the result of syntactic operations applying within the conceptual space, which we have characterized as an *n*-dimensional ultrametric phase space. For ease of visualization, and using Douglas Saddy's expression (see Saddy, 2018; Saddy & Krivochen, 2016a), we are in presence of a 'conceptual sabayon' whose bubbles (i.e., empty space) configure the space of what is not thinkable. Visualization in terms of quantum foam at the Planck scale would be perhaps more serious, but undoubtedly less familiar to the reader; and besides there is no need to be so serious two hundred pages into the thesis. We're in this together now.

whatever the syntactic component sends –or, technically, 'Transfers'; Chomsky, 2008 and much related work-).

An immediate consequence of this proposal we are defending here is the following:

> *Cycles are not intrinsic to the computational system, nor are they based on purely 'formal features' (cf. Gallego, 2010: 51; Chomsky, 2001, 2008) or any other intra-theoretical stipulation, but arise from the frustrated dynamics of the system: cycles are the emergent result of the self-regulation dynamics of the nonlinear complex linguistic architecture we assume.*

Morphophonologically (in MGG terms), cycles are materializable chunks of structure, sets of solutions within the phonotactic space at the segmental level and within the ultrametric lexicon at the word level (although the concept of 'word' is far from clear, we include here both free and bound morphemes, and will not get into further details with respect to the lexico-morphophonological component). Despite global conservation trends, there are also semantic cycles, which we have referred to in past works as 'informational domains' (Krivochen and Kosta, 2013: 162-163; Krivochen, 2011: 50; 2012: 30-31). We provided an example of the dynamics of these domains above, with the derivation of the locative domain |P|, an endocentric prepositional phrase (and see also (113) for a summary of different, yet complementary, motivations for cycles). Now, that discussion can be upgraded with the inclusion of dimensionality considerations as in (123) and its analysis.

If we begin by considering the simplest case, in which the first element to enter the derivational space is defined by means of a single dimension d_x defined on the x axis (a case we will only consider for expository purposes, naturally), the derivation will proceed increasing its dimensionality arithmetically, assuming a simple monotonic structure for predication (which is limited and insufficient, as we have extensively argued here, but good for a toy model):

125) $d_x \rightarrow d_{x,y}(d_x) \rightarrow d_{x,y,z}(d_{x,y}(d_x)))\dots$

That is, in order to characterize the third derivational stage exhaustively, we need to resort to a description of the system in three dimensions. In the notation that we will continue to use in this thesis, 'Dirac notation', we have be a 1 × 3 matrix:

126) $|\psi_D\rangle = \begin{pmatrix} d_x \\ d_y \\ d_z \end{pmatrix}$

In Dirac notation, each derivational step in a monotonic derivational cascade has an associated *ket* vector (for a related, but more neurologically oriented perspective, see Beim Graben et al. 2008: 82). The *ket* can be the result of the cross product between vectors in the case of complex structures; there can be as many dimensional subindexes (x, y, z, …, n) as minimally required in order to describe exhaustively the state of the system at that particular derivational point[157]. If

[157] We have already emphasized that any quantization of time is artificial, but we feel the need to stress it further here: we take derivations to proceed incrementally over quantized time (as we said above, t adopts integer values) just for purposes of the exposition, because, frankly, it makes things easier to deal with. We do not forget, however (and nor should the reader), that we are dealing with continuous, interactive processes. We think this work is only a first step towards a better articulated theory of linguistic derivations in dynamical field terms, which should eventually deal with the continuous nature of time

there were no cycles, the dimensionality of phrase markers *qua* manifolds would quickly grow beyond polynomial time computability (even in an unrealistically simple case as (125)); particularly taking into account that monotonicity is not quite the norm in linguistic derivations: consider not only the case of subjects (which can be indifferently monotonic or not), but also coordination, both asyndetic and overt (Schmerling, 1975; Krivochen, 2015a; Krivochen and Schmerling, 2016); adjunction (Uriagereka, 2002a; Krivochen, 2015a); path-of-motion and resultatives (Krivochen, 2012: 71, ff.; Krivochen and Kosta, 2013: 115-119), parasitic gaps, and multiple gap constructions (Krivochen, 2015b), among other phenomena involving more than a single derivational space[158]. In these cases, we are relating derivational cascades (using Uriagereka's terms) each with its own derivational history, and thus, its own associated field each with its own vector components.

An interesting constraint that arises from this proposal is that, if non-monotonic relations are expressible in terms of operations over the respective *ket* vectors defining states of parallel workspaces, we can only non-monotonically relate objects with the same number of dimensions; otherwise, we cannot operate over the vectors when manifolds intersect. However, we cannot stipulate this *a priori*, since syntactic relations involve a variable number of elements defined by *n* dimensions, and as we are dealing with an open, interaction-based, nonlinear dynamical system, there are essential properties of the system that prevent us from making linearly deterministic statements like that (e.g., hypersensitivity to initial conditions, emergence…stochasticity, in general). What to do, then? One way to see this, is to say that it is precisely this incompatibility that weakens a dimensional attractor in the higher-dimensional object, thus triggering its 'dimensional squeezing' through the weakened axis. That is, the application of the center manifold theorem would follow from the oscillatory architecture of the system in a natural way, yielding cyclicity as a product of the system's dynamics: this is a highly desirable consequence of adopting the field approach, both empirically and theoretically. Naturally, the points at which there is a dimensionality incompatibility are non-differentiable, because the 'cumulative' nature of structure-building operations is halted due to the rearrangement of the system around the stable center manifold: as we have said, if *p* is the (derivational) point at which the dimensionality tension is to be resolved, and the solution involves a reorganization of the system around different attractors, $\frac{dh}{dt}$ is not defined at *p*. Interestingly, from the perspective of multi-dimensional field theory applied to neural dynamics, Schneegans (2014: 2) claims that multidimensional fields require an '*interaction kernel of the same dimensionality*', even though the mathematical possibilities for higher-dimensionality are, in this author's words '*straightforward*'. The constraints over field interaction *are not due to limitations in measuring capacity or theoretical shortcomings, but rather essential features of interactions between multidimensional physical systems.*

A direct consequence of our model is that, as in Uriagereka's, units that are non-monotonic are composed by monotonically derived derivational cascades which have intersected *qua* manifolds, because the topology of the lexical field has been disrupted.

(and other nuances we have overlooked here due to the finiteness of both the time available to complete this work and our cognitive capacities).

[158] Certain operations in TAGs (Joshi, 1985, 1989) also fall within this category, at least formally: it is not obvious –but we are looking into it- how the TAG mechanism could be implemented in a system like the one we propose here.

Crucially, the MSO characteristic of our linguistic derivational system does *not* arise because of the non-monotonic phrase markers' incompatibility with Kayne's LCA or any other monotonic linearization procedure, but because the establishment of a dependency between monotonically derived units in parallel workspaces by means of manifold intersection implies a dimensionality increase that eventually reaches a critical point given the physical and computational limitations of the system, thus triggering the reorganization of the system around the stable center manifold at a lower dimensionality.

The above proposal also makes sense in terms of quantum measurement probability: consider (127)

127) $\langle \alpha \mid \beta \rangle$

(127) makes use of so-called *Dirac-* or *bra-ket notation*, where α is the *bra vector* and β the *ket vector.* The physical interpretation of (127) in terms of measurement is relatively simple, and to exemplify it we will consider the case of electron spin (see also Dirac, 1958: 32 for a criterial *Orthogonality Theorem* applied to orthogonal eigenvectors and their corresponding eigenvalues). Assume we prepare an electron in spin β and measure it with a piece of equipment that detects spin α. The probability amplitude is given by the inner product of α and β; the actual probability, by $|\langle\alpha|\beta\rangle|^2$, that is, the multiplication of the probability amplitude by its complex conjugate (which is trivial if we are dealing with real numbers, i.e., expressions whose imaginary part is 0). Let us consider a relatively simple case, up and down spin; where β is the spin in which we prepare an electron (say, up) and α is the state our detector measures (in this case, down). Up is defined by the coordinates (0, 1)[159] whereas Down is defined by the coordinates (1, 0). Both *bra* and *ket* vectors are defined as arrays of components, such that:

128) a. Up: $\langle\alpha| = (0, 1)$
b. Down: $|\beta\rangle = \binom{1}{0}$

The inner product of the bra-ket notation, as we have said, gives us the probability amplitude, so let us perform the relevant operations over the vector components, which give us a scalar (Dirac, 1958: Chapter 1):

129) $\langle \alpha \mid \beta \rangle = 0 \times 1 + 1 \times 0 = 0$

This means, as we would expect, that the possibility of measuring a spin down electron with a spin up detector is 0 (in order to get the probability, we have to square 0, which of course equals 0). In more general terms, we say (130) (a *delta* function; following Feynman, 2013: Vol. 3, Chapter 5, equation 5.25; Chapter 16, equation 16.3; also Dirac, 1958: 39):

130) $\langle \alpha_i \mid \beta_j \rangle = 1 \textit{ iff } i = j$
$0 \textit{ iff } i \neq j$

Translated to physical terms, this means that if we prepare an electron with spin *j*, and measure it with a detector configured in spin *i*, the corresponding eigenvalue (i.e., the result of the experiment) for the vector representing the measurement will be 1 (i.e., complete certainty the electron will be detected) if and only if the spin configuration of the electron and that of the detector are equal (e.g., both up, both down). Otherwise (and limiting ourselves to a two-

[159] To be read: '*x* components down, *y* components up'

outcome system[160]), the probability of detecting an electron is 0. Crucially, in more complex systems, where the bra-ket is just one component, we *have* to assume $i = j$, otherwise, the whole term just cancels. For instance, assume that we have a state $|\psi\rangle$, and we want to know the prospects of preparing the system in state ψ (which is really a sum over the complete set of base states $i, j, \ldots$) and measuring it in state x_j. So, what we really want to know is the probability amplitude for the system described by ψ to go to a specific base state x_j That is expressed as follows (see, e.g., Littlejohn, 2007; Feynman, 2013: Vol. 3, Chapter 5, equation 5.24; also Chapter 8, equation 8.1),

131) $\langle x_j | \psi\rangle = \langle x_j | \sum_i a_i |x_i\rangle = \sum_i a_i \langle x_j | x_i\rangle$ [161]

where we act with an operator a which cycles over the values of i for the initial state of the system[162]. In a formal scenario like (131), we have to assume $i = j$ holds (a situation frequently noted as $\delta(x)$, the so-called 'Kronecker *delta*' functional), otherwise, the result of the inner product will be 0, thus cancelling the whole term (since $\sum_i a_i\, 0 = 0$). If the *bra* and *ket* vectors indicate measured and prepared states of a quantum system respectively, we cannot have $i \neq j$, for that would mean the result of measurement is 0 (i.e., we detect nothing); and if ψ is a sum over all base vectors, then clearly there is a non-zero possibility of ψ ending up in state x_j. If, following standard practice in quantum mechanics, each dimension is a component in the vector, we can only operate with vectors that have the same number of components, or set the missing components in one of the vectors to 0 (that is, say 'the system *does not have a value* for component n'). Such an operation would be tricky under present assumptions: imagine we have the inner product (or 'dot product') of α and β, such that α is defined in 3 dimensions (x, y, and z) and β is defined in 2 (x and y). The inner product would be:

132) $\langle \alpha | \beta\rangle = \alpha_x \cdot \beta_x + \alpha_y \cdot \beta_y + \alpha_z \cdot 0$

This operation, if valid, would artificially decrease the dimensionality of the operation output by 1, since the z component (i.e., the value of the z coordinate) in α would be simply neutralized. This dimensionality reduction, however, would not arise from the inner dynamics of the system (as opposed to the *center manifold* dynamics), but represents a violation of much more basic physical constraints over system interaction and measurement. In our opinion, (131) and the brief discussion that follows it are another way of saying that multidimensional field interaction '*requires interaction kernel* [sic] *of the same dimensionality*' (Schneegans, 2014: 2), but in a slightly different manner: if the 'same dimensionality' requirement is not met, the system forces a dimensionality reduction, which we have expressed in topological terms (via the center manifold theorem) and now in purely physical terms, via *bra-ket* notation. Now, is this

[160] The same procedure works for photon polarization. Assume that we prepare a photon with a polarization m which is at an angle θ with respect to the measure n. The probability of measuring m in n is defined as $(cos\frac{\theta}{2})^2$. So, for instance, if m and n are at an angle $\theta = 0°$, $(cos 0°)^2 = 1$. Notice that in this case, $m = n$, which satisfies the equality in (140).

[161] Note that we can take $\sum_i a_i$ out of the *bra-ket* notation because they are numbers –scalars-, not matrices or vectors. The result is the same, see Dirac (1958: Chapter 2) for proof.

[162] Provided that any wave can be decomposed into the sum of other waves, we sum over all components of the wave we are considering. Thus, $\psi(x) = \Sigma_k\, a(k)\, e^{i(kx-\omega t)}$, which means that our field ψ is the sum over all possible amplitudes a associated with all possible k for waves defined as the sum of a real cosine and an imaginary sine component (which, by Euler's formula, equal an exponential function) with varying wave number k.

physical dimensionality reduction relevant? We think so, as either we are forcing the rise of a cycle, or we are simply losing all the information in the neutralized dimension. If the dimensional incompatibility between a number *n* of *ket* vectors did not force the fading of an attractor and we just kept computing as if nothing had happened, we would be violating the *conservation principle* that globally rules derivations, by losing information (instead of squeezing that information through a lower dimensional complex plane).

A similar position with respect to the relation between cognitive processes and their externalization in terms of action or language is that of Spivey (2007), which we introduced above. According to Spivey,

> *Action and communication* […] *necessarily overidealize and exaggerate the discreteness of people's internal representations, typically settling on the closest response category to what was originally intended.* (2007: 168)

In terms of externalization, then, there is an essential discontinuity -imposed by means of chunking and Spell-Out- over cognitive representations and processes, which are essentially continuous over time. This view is also consonant with DFT, although the latter provides tools for describing the state of the field at an arbitrary discrete point or set thereof by means of quantizing *t* (see the differential equations in Schöner, 2009, for instance), which are most useful insofar as we keep in mind that such quantization is artificial. Interestingly, Spivey's quotation leaves open the interpretation that 'what was originally intended' is not a point, but a phase space on its own, and the 'closest response category', an attractor within that space. Note that if the *intention* was a point (or even a vector, or a tensor) rather than a set of possible solutions that satisfy the problem to different extents (in a core-periphery dynamics), then there would be a single possible externalization *per* internal representation, that whose coordinates (the corresponding *ket* vector) coincide exactly with those of the representation. In our opinion, such a deterministic model is too strong and constrained, and it is not clear how a linguistic representation (or any other cognitive representation) could encode the required specificity in terms of attractor localization. Our model is quite a bit more flexible: even if a representation whose interpretation is to be related to an intention (e.g., for the purposes of inference extraction, where the hypotheses we make about the other person's intentions when producing a certain stimulus are crucial), there is an error margin which allows for the 'closest response category' to be located and which, for all practical purposes, might be very well equivalent to the exact point. This approach, we think, makes the process of finding an attractor not only intuitively 'easier', but also computationally less costly, and formally explicit, which is a desirable result.

If the picture of variable entropy *qua* cycles we sketched above is correct, and if time expressed in positive integers (that is, *t* = 1, 2, 3…) corresponds to derivational cycles, then, $\lim_{t \to x \in \mathbb{N}} h(t) = 0$. The function is limited by two asymptotes, 1 and 0 (complete disorder and complete informativity, respectively), and it oscillates between attractors at (*t*, 1) and (*t*, 0), where *t* is a positive integer. Since our attractor points belong to an asymptote, Spencer, Perone and Johnson's (2009: 109) proposal that the neurocognitive system goes *into* stable attractors, having enough flexibility to 'escape' the stable attractor and change the system dynamics does not seem to be feasible at least at the computational level, as no value for the function can ever reach a value within the asymptote. We will see that characteristics of physiology depart from those of its emergent computational systems, in nontrivial ways. We do have to stress again, however, that we will adopt integer values for *t* for *purely expository purposes*, corresponding to derivational steps equally *artificially* quantized (otherwise, we would be falling in an internal contradiction!). The focus on *process* rather than *state* is a basic feature of interactive-based computation, and we would like to stick to it as far as we can, since we do think the theoretical and empirical payoff is worth it. Particularly, the perspective of making a step towards a model of brain dynamics based on the notion of *tension* (i.e., *dynamical frustration*), which in turn is pervasive in (other) physical systems (just like *cycle* is) is too promising to be left unexplored.

Assuming a cyclic computational system working along the lines described above, we still have to give some more precisions about the properties of the space in which we are plotting the function: what exactly does it represent? Since entropy reduces as derivational steps proceed (which in turn correspond to cumulative structure building), the certainty about *something* must increase, *something* must be easier to pinpoint…(and, of course, at some cost) We are then forced to ask, *what is this 'something' exactly*?

We assume the ***n***-dimensional space in which operations take place is the *phase space* containing all possible interpretative options, including both natural and non-natural meaning (in the sense of Grice, 1957 and related works); Fregean-componential meaning read off structural configurations (Hale and Keyser, 1997), but also lexical associations, inferences, and so on. The ground state dynamics of the system defines an ultrametric space, but imposing structure over the lexical field brings things together, which can be described by means of a metric function; this process implies a change in the topological properties of the field which we will describe in more detail below.

The relation of lexical fields by means of procedural elements can be described in geometrical terms by means of a *distance* or *metric* function *d* that 'brings elements closer' than they would normally be unrelated attractors in the ultrametric lexicon: syntax (i.e., structure, relation) disrupts the ultrametricity of the lexicon, imposing a metric over an otherwise non-metric space:

> *The idea is that when you merge, say, "men" into "like arguments" (or some such), you are* ***literally getting "men" to a proximity w.r.t. "arguments"*** *that it would not otherwise have had (as compared to, say, "men" and "boys" or "arguments" and "discussions", say).* (Juan Uriagereka, p.c. Our highlighting)

Interestingly, this argument seems to contradict the ultrametricity of syntactic representations argued for by Roberts (2015), and supported by Uriagereka himself (if we assume these authors' take on the matter is that the relevant syntactic space is *uniformly* ultrametric, which is the null hypothesis, and there is no *explicit* argument to the contrary in Roberts' or Uriagereka's papers). However, there would still be a way around it if we are dealing with two distinct ultrametric spaces (the lexical and the syntactic ones), but we see no sound argument in favour of such multiplication of non-metric spaces, as we argued above. Rather, and following Uriagereka's idea, we propose the ultrametricity of a field can be disrupted. What are the consequences of such an approach? Elements brought together via concatenation (or, more accurately, the elements whose fields interfere due to the presence of a procedural / relational element) modify the structure of the field over time, changing the topology: as we said above, syntax imposes a metric over the phase space, while at the same time restricting it in terms of the probability amplitudes of finding a relevant attractor in a particular subset of the phase space.

The interpretative phase space of a cycle (i.e., the subset of the total phase space where we are more likely to find the attractors corresponding to the perturbations we are interested in), which we have located in the interval (0, 1), is defined as the integral over a curve defining a cycle (as we are still working in 2-D), expressed as follows:

133)

$$\int_{0}^{1} (cos(x) - n\Delta h)dx$$

Recall we said that at integer units of time (in an arbitrarily quantized time scale for expository purposes only), the function is discontinuous; it is not differentiable at those points (as a cycle ends, and entropy augments to (Max$h$ – $n\Delta h$) < 1). By the fundamental theorem of calculus, if the function is not differentiable at a point, it is not integrable at that point either. The function that describes the derivational dynamics we propose is therefore only *locally integrable*. This point is essential in our argument: differential equations model the behavior of the system, and integration gives us the tools to delimit the phase space where the probability of finding an attractor corresponding to the relevant interpretation is *sufficiently* high[163]. By using the word 'sufficiently', we appeal to the cost-benefit ratio that underlies most, if not all, computational operations: surely, a system could try to look for the relevant attractor outside the space limited by the integral despite there being a very low possibility of finding anything of interest there, but such a system (which would operate without boundaries, searching within all of the phase space at every derivational step) would not be able to 'solve' any interpretative problem in a reasonable amount of time, as the phase space is arbitrarily large. Let us consider a probabilistic (or otherwise nondeterministic) TM: most of them would be inclined, when faced with competing rules, to favor the one that has a greater probability of 'success' (unless we are dealing with a 'luckiest possible guesser' kind of TM, which would be completely random despite the probability amplitude associated with different outcomes, or a completely stochastic process, in which all possible computational paths have the exact same probability amplitude).

How does this interact with our previous argument that predicate-argument relations increase the dimensionality of the output? Let us assume we start off with a single element (the number or nature of this element's coordinates being irrelevant for the time being), and the system manipulating this element evolves in time. The rate of change in entropy over time after an arbitrary lapse, (t_0 + Δt), will be given by the already familiar differential equation term $\frac{dh}{dt}$. Now, since introducing even a single element from an array into the workspace already limits the phase space, selecting a proper subset of the whole thing, we can actually calculate the portion of the phase space within which we should be looking for attractors in the time span t_0-t_1, where $t_1 = (t_0 + \Delta t)$: $\int_{t_0}^{t_1} f(t)\, dt$ -recall that $f(t) = h$-. Since the development of the system will be described by means of a (cosine) curve, the corresponding integral will delimit a *surface* below that curve, within which we are more likely to find attractors than in the rest of the phase space, as we have said above. What happens if we introduce a predicate in the derivation? According to (120), the dimensionality of the object increases stepwise; consistently, we are not dealing with a point whose evolution in time determines a curve anymore: we are now dealing with a plane dividing a 3-D phase space in two (i.e., 'places where the possibility of finding an attractor is high' vs. 'places where the possibility of finding an attractor is low'). Consequently, the integral is no longer a curve integral, but a surface integral over a closed interval for a

---

[163] In other words, the integral gives us the relevant phase space for a closed interval [a, b], and the slope of the function, defined by the derivative, tells us whether that phase spaces shrinks or expands.

surface **S** defined by a wavefunction $\psi(x)$. The function defines an oscillatory field with angular frequency ω in space and time, and is usually expressed in the form:

134) $$\psi(x) = e^{inx}e^{i\omega t}$$

What is *n* doing there, in an otherwise standard wavefunction? Recall our function is not just a cosine function (with its imaginary sine counterpart), since there is a modifying dimensionality factor *n* which depends on the predication relations established within a structure (Uriagereka, 2002b; Krivochen, 2015a; also (120) above). (134) is equivalent to

135) $$\psi(x) = \cos(nx - \omega t) + i \sin(nx - \omega t)$$

by Euler's formula; the latter expression shows the two waves in the real and imaginary planes more concretely. The curved surface defined by the field equation (134) divides the total 3-D phase space in two, and the integral of S, the surface of our field, now defines not a surface, but a *volume* under the curvilinear surface (Courant and John, 1974: 374, ff.). It is relevant to point out that the picture of curvilinear surfaces in a 3-D system is already familiar to neuroscientists, particularly those working within Dynamical Field Theory (see, e.g., the graphs for neural firing over time in Schöner, 2009: 33; also Sandamirskaya et al., 2013: 327-328; Scheegans, 2014, for an approach more centered on visual perception). How do we describe a cycle with initial 'momentum' value k_i and final momentum value k_f? (136) is a possibility:

136)
$$\langle k_f| \iint d\phi \, dt \, a^{\dagger} (\phi, t) \, a(\phi, t)|k_i\rangle$$

Which means, for a 'particle' with momentum k_i, that we apply the field annihilation operator *a* over the variation of space, since our field oscillates in space, at time *t*; then we apply the creation operator $a^{\dagger}$, which cycles over the field within an interval determined by the variation in the field $\phi(\frac{dx}{dt}, \frac{dy}{dt}, \frac{dz}{dt})$ at time *t* (we assume both operators operate at the same time, or during the same time lapse) and creates a 'particle' with momentum k_f, which equals k_i since there is no other source of momentum, and momentum is a conservative quantity.

Let us develop (136), to make the point clearer. We know that $a|k_i\rangle = 0$, and that $a^{\dagger}\langle\alpha_i| = 0$ from the definition of the creation and annihilation operators. Consider, also, that we have a field with angular frequency ω which is related to the energy of the wave (such that $E = \hbar\omega$). We see that both *k* and ω are quantized in units of $\hbar$, which is perhaps the most important take-home message of QM. So, what we really have in (136) is:

$$\langle 0| \iint d\phi \, dt \, a^{\dagger} (\phi, t) \, a(\phi, t)|0\rangle$$

Let's now expand the field terms, assuming a one-dimensional field (such that ϕ will be replaced by *x*) just for the sake of the exposition:

$$\langle 0| \iint dx \, dt \sum_k a^{\dagger} (k) e^{-i(k_f x - \omega t)} \sum_k a \, (k) e^{i(k_i x - \omega t)}|0\rangle$$

Now, if we carry out the integration and group wavenumber and angular frequency terms, we get:

$$\langle 0| \iint dx\, dt\, e^{i(k_f - k_i)(\omega_f - \omega_i)}\ |0\rangle$$

This means, basically, that $k_f = k_i$ and $\omega_f = \omega_i$, or, in other words, *momentum and energy are conserved* between initial and final states of the quantum system.

Focusing on k, the *bra-ket* gives us the probability amplitude to get k_f from k_i, and we integrate over time and space because the process of creation and annihilation can take place 'anywhere, anytime' (this integration requires a parametrized space). Now, is there anything we can do if we are not particularly interested in *momentum* as such? Well, yes; k, defined as $k = p/\hbar$, can be interpreted as an *energy* term related to the *amplitude* of the wave at a particular point (x, y, z) -or, in this case, a set thereof, $(x_2 - x_1, y_2 - y_1, z_2 - z_1)$, abbreviated $\Delta\phi$-. Crucially, as we have seen, *energy* is also a conservative quantity, so whatever we say about k is also valid for ω. The application of annihilation operators allows us to account elegantly for the discontinuity of the function (which in turn determines the presence of *cycles* computationally) within a field-theoretic perspective.

The considerations about dimensionality that we have just discussed entail that any attractor within a volume will be defined by a three-component *ket* vector, instead of just two, as we are not dealing with just (x, y) spatial coordinates anymore, but with a system of (x, y, z) coordinates; plus a *time* dimension. The search space is, on the one hand, restricted (since there is more certainty about the structure, and predictions about what could come next can be improved) *but* there is a new axis along which elements can be located, a whole new dimension in the phase space. The introduction of further structure, then, disrupts the ultrametricity of the system at 'rest state' imposing a metric function over perturbations in the lexical field (see Uriagereka's remark in this respect, above) and reduces the local entropy of the current derivational current (pun intended), but at the cost of introducing a new dimension. So far, however, there seems to be no problem: the human mind is perfectly capable of handling 3-D spaces, understanding their topological properties (e.g., assigning metric functions among objects dynamically), and predicting the behavior of entities within such spaces (even when they might not be geometrically or topologically uniform). This is something we literally do all the time, configuring the interface between our mind and information coming from the phenomenological world in the form of sensorial stimuli of various kinds. A 3-D phase space poses no problem when we have to assign a structural description or compute local probabilities for certain elements to behave in certain ways.

At this point, it is relevant to remind the reader we are describing a single workspace at a time, that is, a *single derivational cascade*. Crucially, this limit can be worked out from (123): if dimensionality increase is given by predicate-argument relations, the operations that generate these structures apply all within a single workspace, and are halted when the manifold reaches a critical dimensionality value and the CMT is triggered. Cross-workspace relations, while existent (see Krivochen, 2015b for some previous discussion about multiple-gap constructions, which are good candidates for such relations), do not increase the dimensionality of the overall manifold, since each cascade is cognitively taken as a *set* (in Feigenson's 2011 terms) for the purposes of further computations involving cross-workspace dependencies (which would explain opacity effects for the purposes of extraction *across sets* –i.e., island effects, which we will look at in depth in **Part IV**-, some of which we analyzed in Krivochen, 2015b; as well as binding-theoretic related phenomena, see Krivochen, 2015c). The considerations involving the relevance of integrals for the purposes of restricting the phase space are thus constrained to a single workspace at a time. To see this, consider that once a domain has undergone transfer after reaching a critical D value (i.e., one of its attractors has faded[164] because the system cannot look into that dimension due to its own computational limitations), the system cannot establish a dependency between an object defined by a vector in n dimensions and a member of the dimensionally flattened *set*, where the dimensionality of the flattened manifold is lower than n. Why not? Because, if the establishment of a dependency implies an interference between field perturbations defined by respective *ket* vectors, each of which is to be *found* by the interpretative system (whichever it might be) and there is a mismatch, there is a whole axis along which one of the elements could be (a whole dimension of points!), and that is inaccessible from the lower-dimensional element. Thus, there is an axis along which there is no interference, that is, no relation at all. For the sake of concreteness, imagine we have a point defined by x and y coordinates, and another point defined by y and z coordinates: is it possible to draw a line between both points? Most certainly not. A line between two points is analogous to a vector (insofar as it has a module and direction, but it is not oriented), and a vector cannot 'cross' dimensions, just like we cannot have a *bra* with more or less components than its corresponding *ket*; they are transposed complex conjugates, and they are defined in the same number of dimensions.

3.10 Some perceptual predictions involving dimensions

There is psycholinguistic and neurocognitive evidence that we can find attractors within 3-D spaces, and we do navigate our way in a *locally* Euclidean space every day with relative ease and remarkable consistency. However, problems arise when a fourth spatial dimension is considered. The limitations of human cognitive capacities in terms of the perception and manipulation of space in four dimensions has been the subject of much controversy (Ambinder et al., 2009; Afalo and Graziano, 2008; also Abbott, 1991 for a quasi-fictional perspective), and virtual reality simulations have been used in order to test the possibilities of stretching geometrical intuitions beyond 3-D (Ambinder et al., 2009). Afalo and Graziano (2008) used virtual reality mazes to test the extent to which human participants could assess distance between points and orientation in what are allegedly 4-D mazes. The authors claim that:

> *When a virtual 3-D world is displayed on a flat computer screen, a simple projective geometry is used. Virtual light rays are projected from points on objects in the virtual world to the flat screen, resulting in a display that captures the appearance of a 3-D world. This appearance includes perspective cues, motion parallax cues, and occlusion cues. Because of these cues, subjects are able to perceive a 3-D world through the window on the computer screen. In a similar way, a virtual 4-D world can be projected onto a flat computer display. The computations are mathematically the same as in the 3-D case. Light rays are projected from points on objects in the virtual world to the flat screen, resulting in a display that captures the appearance of a 4-D world. The subject sees the projection of a 4-D world onto a 2-D surface.* (Afalo and Graziano, 2008: 1069)

The authors have found increasing errors in terms of degrees as participants proceeded from 2 to 4 dimensions. They find transitions between error patterns in 4-D around Trial 140 that '*could be explained only by 4-D competence*' (Afalo and Graziano, 2008: 1073). However, even models of spatial adaptation through practice (e.g., that of Helmholtz, 1962), which the authors

[164] We are referring to the *center manifold theorem* here, see Schöner (2009: 29) for a neural field theory perspective.

mention, are not quite appropriate, because the specific properties of the space the subjects would be adapting to are still unclear. While the authors claim that

> *The 4-D world is not a misalignment of familiar spatial properties. Instead it is topologically different from the familiar 3-D world.* ***The present results suggest that the human brain is capable of learning 4-D path integration***. (Afalo and Graziano, 2008: 1074. Our highlighting)

We think that last claim is too optimistic. It is true that the topology of the 4-D space is different from that of the 3-D space, but the added dimension is another *spatial* dimension (therefore, for all physical purposes, it behaves 'the same' as the other spatial dimensions, so that it is impossible to say 'this is the first dimension', 'this is the second', and so on), and the mathematics to get from 3 to 4 is quite simple: one just adds a component to the relevant vector (or tensor). True enough, the authors focus on the topology of the 4-D space *from the perspective of a perceiving subject*, but then, it is almost otiose to point out that the stimuli were presented in a dimensionality below 4. In fact, the '4-D mazes' were presented in a 2-D screen, as a 'projection' of the 4th dimension. The extent to which that can be actually regarded as 4-dimensional is not a trivial question, just like it is not trivial to ponder whether the illusion of depth generated in, say, First Person Shooter games *actually yields* a 3-D space for all cognitive intents and purposes, although the mind 'adds' the missing *z*-axis we need to navigate our way through the game[165]. For all practical purposes, we think technological limitations make the experiments described in the aforementioned references theoretically trivial. But, what about their underlying assumptions? After all, the mathematics underlying four spatial dimensions is fairly simple; in the terms we have been managing here, just add a further component to the relevant *ket* vector, all in all it is just a number. In that respect, it is not too far away from the truth to say thay '*The computations are mathematically the same as in the 3-D case*', in Afalo and Graziano's (2008: 1069) terms (considering all due caveats, let us assume they mean $\mathbb{R}^3$ and $\mathbb{R}^4$). However, the system these authors assume for the relevant computations is not only dimensionally bound (which in and of itself is not a problem, since a computational system can theoretically generate an output that displays higher dimensionality as an emergent property, as seems to be the case of certain instances of L-grammars generating fractal structures[166]) but also bound formally: the procedures used to generate the relevant stimuli are based on functions over binary code. It is very difficult indeed to evaluate the exact extent to which the relevant stimuli are 4-D, rather than manipulations over 3-D polyhedra. Ambinder et al. (2009) presented alleged hyper-polyhedra (i.e., 4-D polyhedra) *as they would be seen while passing through the 3-D space* (in a move admittedly inspired by Abbott, 1991). That is, 'how would a 4-D polyhedron look like when passing through the third dimension?' Then, they evaluated subject sensitivity to orientation and distances on statistical grounds (Ambinder et al., 2009: 819-820). There are many problems with this methodology, from our perspective: how is a 4-D space

[165] The realism of the added dimension and its effect on the perceiving subject was greater when the first FPS games came out, and gamers were not used to the idea of moving their character in a 3-D space. The writer of these lines was terrified each time he dared play Doom (1994), because the monsters would come out of the screen and attack him (in my defence, I was 5 years old).

[166] A relevant case would be the generation of the Sierpinski triangle by means of a substitution procedure, in which *n* triangles in an area *a* are replaced by 3*n* triangles occupying *a*. Complexity increases logarithmically, and the Hausdorff dimensionality is defined as log(3)/log(2) ≈ 1.58496 (see Helmberg, 2007).

different from a 3-D space in a 'statistically significant' way? Assume we let a ball roll down a hill: the perception of velocity, acceleration, direction, depth, etc. are *undoubtedly* 3-D, and all relations can be defined in that way (e.g., acceleration is the derivative of velocity over time). Unlike geometrical intuitions *within* a dimensional phase space, perception of dimensionality involves orthogonality: while the former accepts, for instance, hyperbolic zones within a globally Eulclidean space (e.g., a saddle on the floor); we unambiguously assign a dimensionality factor to sensory stimuli (e.g., in the case of First Person Shooters, we 'suspend dimensionality disbelief', and proceed *as if* we were submerged in a 3-D space, with which we are familiar, adding a dimension that is missing in the input). Thus, either *all* judgments are *completely* compatible with a 4-D space, or *none* is. Sure, *within* a dimensional framework, subjects can differ with respect to their judgment of distance between vertices in a polyhedron, but that variation tells us nothing about the *dimensional* structure the subjects assign to the relevant stimulus (cf. Ambinder et al., 2009: 821), but to geometrical (metric) considerations *within* an *n*-dimensional vector space. Ambinder et al.'s use of regression techniques is as unjustified as it is pointless (also, see Afalo and Graziano, 2008: 1073), because the question is *polar*, not probabilistic in nature: *can humans assign a 4-D structure to an input*? In our opinion, all the articles show is that, for a certain class of stimuli, presented under 2-D conditions simulating a 3-D space through which 4-D objects pass, or a 4-D space through which 3-D subjects navigate; participants can make guesses of varying precision (and the term 'guesses' highlights the statistical nature of the experiments) with respect to 3-D compatible notions (like distance and orientation). Not too much to say, actually. Therefore, we will continue to work with 3 as the dimensional limit that triggers the fading of the center manifold in this thesis.

Crucially, the computational dynamics we propose emerge from physical constraints which are *external* to the computational system considered *in abstracto* (the latter being the object of study of MGG). If we had a non-embodied infinitely powerful system and infinite time, then we could, in principle, afford the luxury of building indefinitely long chains of predicates / arguments in a single workspace, and the corresponding integral would define higher and higher dimensional phase spaces where the relevant attractors could be looked for. We could even work in exponential time[167] to find the relevant attractors, which amounts in practical terms to saying that any structure can mean anything, since we can look for the relevant attractors anywhere within an almost unbounded phase space (a lifetime supply of sabayon, say) with infinite computational power. But things are clearly not that way: the computational system we have characterized is an emergent property of a certain physical configuration, and what could be seen as an abstract system is actually constrained by the properties of its embodiment.

Needless to say, much more work is required at the mathematical and physical levels, and also at the level of neurodynamics. For the time being and for the purposes of the present thesis, however, what we have said thus far is a sufficiently close approximation for what follows in **Part IV**, which explores the specifics of the notion of *cycle* that emerges from our theory.

---

[167] An algorithm is said to be exponential time, if T(n) (the maximum amount of time required to parse an input of length n) is upper bounded by $2^{\mathrm{poly}(n)}$, where *poly*(n) is some polynomial in n. More formally, an algorithm is exponential time if T(n) is lower bounded by $O(2^{n^k})$ for some constant k. These problems are hypothesized to be NP-hard, or, at the very least, NP-complete.

It is essential to close this part stressing the importance of *scale* issues. What the field-theoretical approach developed here entails is *not* that there is a correspondence between, say, cycles of depolarization and repolarization, and syntactic cycles. Or brain oscillations and 'phases'. The main point of **Part III** is that at different levels (which by no means have to be reduced one to another) we find oscillatory patterns, and the physical concept of *field* can give us deep insight on those patterns and what they mean at each level. Thus, the equations for the behaviour of the neurophysiology and the computation as systems that evolve in time could in principle share a common *format*, but the relevant variables, and the scales at which they matter, could be *completely different*. The field-theoretical approach focuses on the abstract similarities between the oscillatory patterns that underlie cognitive capacities (involving interactive computations) as well as neurophysiological behavior (mainly electric and electrochemical synapses). Only bearing in mind differences of scale can the physics of language approach build bridges with other disciplines and allow us to gain valuable insight on the human mind and its material substratum as a physical system.

Part IV: On an Island: Applications of a Mixed Computation Approach

A dynamical system that displays *scale frustration*, crucially, *cannot* be modelled by means of a uniform computational system: if there was a single kind of structure being derived (say, {H, XP} units at each derivational point), then local properties of phrase markers should map linearly onto global properties. In contrast, a computationally mixed system, in which phrase markers display local properties which depend on the semantic dependencies between objects, can readily capture the non-linear mapping between local and global properties of linguistic structures in *scale frustrated* systems.

Chomsky (1957: 21) argues that 'English is not a finite state language' and proposes Context-Free phrase structure rules and Context-sensitive transformations; Joshi (1985) proposes that we need mild-context-sensitivity to provide adequate structural descriptions for natural languages in the light of examples featuring crossing dependencies, and Scheer (2014) posits 'flat phonology': there are only linear relations, no hierarchical structure (these relations are finite-state). However, when analyzing different domains in the grammar, finite-state, context-free, and mildly context-sensitive dependencies appear all across the linguistic landscape if we look at it *locally* (at properties of substrings). Let us see an example of a natural language string which displays varying computational dependencies, which we introduced way above:

16') a. [[The man$_i$ who shows he$_i$ deserves it$_j$] will get [the prize$_j$ he$_i$ desires]] (Bach, 1970; Karttunen, 1971)

b. Every pilot who shot at it hit the MIG that chased him

What we propose is that there are several *local* structural layers here, each displaying varying levels of computational complexity within the Chomsky Hierarchy, and that the assignment of a strongly adequate structural description to these sentences *requires* the system to be sensitive to local changes in computational complexity. Let us illustrate this point using (16' a) as our example:

a) The – man; the – prize; he – deserves – it $\Longrightarrow$ **Finite-state**

b) [The man *S'* [will get [the prize *S'*]]] $\Longrightarrow$ **Context-Free**

c) [*the man* [who shows [he deserves **it**]]] will get [**the prize** [*he* desires ***e***]] $\Longrightarrow$ **(Mildly) Context-Sensitive**

Note that the differences arise *within local derivational units*. A strongly adequate grammar, needs to be flexible enough to accommodate for oscillatory dependencies in assigning structural descriptions to sentences, going *up and down the Chomsky Hierarchy* in local domains. In this perspective, we can define *cycles* as *emergent properties* from a computational system that does not commit to a 'one-size-fits-all' theory of phrase structure, but to one that models language as a non-linear, dynamical system (Saddy & Uriagereka, 2004; Saddy, 2018, and related work). In such a system, computational uniformity would in fact have to be *stipulated*.

In other words, any of these computational systems taken alone only partially tessellates the phase space for linguistic computation. In this context, we need to revisit and in fact challenge two basic yet often implicit assumptions of modern linguistic theory:

- Structure can only be obtained through recursive combinatorics
- A single computational engine is sufficient to generate minimally appropriate structural descriptions for all and only grammatical strings

Looking at examples like (16' a), we can see where cycles emerge in a computationally mixed model that captures the *scale frustration* that we (with Saddy and Uriagereka) think is at the very core of natural language: computationally uniform dependencies define *local domains* within which we can probe: *cycles*. In this context, we do not need to appeal to substantive elements to define cycles (including bounding nodes, as in the Standard Theory; barrier nodes, as in Chomsky, 1986a; phase heads, as in Chomsky, 2001, etc.); rather, cycles emerge from the dynamis of the computation. Simply put, when we need to change computational dependencies because a given kind of grammar is either too weak or too powerful, the current cycle is closed.

This chapter explores the dynamics of linguistic cycles and the locality conditions that arise in a derivational system in which local domains are defined by changing computational dependencies. Within generative grammar, the local domain *par excellance* are *islands* (Ross, 1967); we will briefly recap several locality conditions that have been proposed in the linguistic literature and try to find one that is strong enough to be considered a *hard condition* on linguistic computation: a constraint that cannot be violated. These conditions, we propose, help us delimit the phase space for linguistic computation, and it is precisely because such inviolable conditions exist that the linguistic system yields usable outputs (Saddy, 2018).

4.1 Deriving some empirical predictions

The main point of the discussion in **Sections 3.9** and **3.10** is that no single workspace can hold more than 3 dimensions at once due to inherent topological limitations of the human mind accessing higher dimensionalities within a derivational current[168], and we kept that

[168] Cf. Mori (2005), who proposes caused Vs to be 5-D (or, at least, to have 5 orders of dimensional complexity, which *might not* be equivalent to 5-D), see Mori (2005: 44, ff.; 250-255) for arguments in favour of this take on dimensionality, which is of course incompatible with the one proposed here. A way to reconcile both proposals would be to apply cyclic considerations over Mori's representations, and

premise alive in the formalization of the computation in terms of fields by defining a *cycle* by means of a 3-component *ket* over which operators operate. Summarizing previous discussion, 3 is the critical value that triggers the fading of a dimensional attractor, and thus the system squeezes to a lower dimensionality space. However, as we pointed out, that does not mean the object resulting from *unifying* derivational currents has 3 dimensions. The dimensional limits imposed over the objects that can be derived in a workspace and the possibilities for establishing cross-workspace dependencies would determine the limits of what is 'thinkable' (in the conceptual space) and what is linguistically representable. Non-trivially, if we have a fully explicit theory of the topology of the space where derivations take place, as well as of the possible relations between 'terminals' in terms of perturbations of an ultrametric field whose properties are disrupted by syntactic relations imposing a metric function over elements; we should be able to formalize conditions over possible dependencies. The kind of system that we have been describing yields strong cyclic conditions, and sets boundaries for operations probing into syntactic objects. Physically, these objects are squeezed manifolds, through the CMT. Computationally, each cycle displays dependencies of a single type, and it is the change in the computational complexity of semantic dependencies between terminals and nonterminals that defines the cycle; this view of cycles as non-periodic minimally interpretable – maximally generable units arises naturally in a mixed computational system. In the dynamics of syntactic cycles lies the deep connection between **Parts I**, **II**, and **III** of the present work.

This topic is not new, in fact, it has been one of the main elements of syntactic theorizing for the past 60 years[169]. Oversimplifying things for the sake of concreteness, we can find four main parts in any formal theory from the '50s onwards:

137) a. Structure building (Phrase Structure Rules, X' theory, subcategorization frames, *c*-structure, *f*-structure, etc. We have addressed these concerns in Krivochen, 2015a, 2016b, c, d; Krivochen and Schmerling, 2016; among other works)

b. Structure mapping (Transformations, displacement, slash-features, gap-filling, ellipsis, etc. We have addressed these concerns in Krivochen, 2015b, 2018b)

c. Binding and scope (relations between sortal entities in different domains, relations between eventive entities, including *consecutio temporum*, interactions between operators and variables. We have addressed these concerns in Krivochen, 2015d; Bravo et al. 2015; García Fernández et al., 2017)

d. Morphophonology: how structure gets dynamically linearized and assigned phonological exponents (the former of which has been the focus of the functions over manifolds we have been working with here, see also Uriagereka, 2012; Schmerling, in press for a completely different proposal, yet potentially compatible with the framework presented here)

chunk them in 3-D compatible sub-derivations. However, that would disrupt her thematic / argument hierarchy, probably yielding empirical consequences. We will not attempt such reconciliation here.

[169] On this point we have claimed many times, paraphrasing Alfred North Whitehead, that 'The safest general characterization of the Generative and transformational tradition is that it consists of a series of footnotes to Ross (1967)'

All four are closely related and, for us, a theory of language should provide *explanantes* at (at least) those levels and their interrelations. Note that we have been focusing on structure building as a nonlinear process, which is globally aimed at reducing entropy (following Uriagereka, 2011), with the relevant locality considerations (*as per* Krivochen, 2014c). In turn, the determination of limits on what is computable in a workspace in terms of the dimensionality of phrase markers (to use the traditional term) impacts on the possibilities for structure mapping, even though we strongly adhere to a non-transformational stance: we can, as it were, build Poincaré maps of the system, taking a 'snapshot' of the state of the system at any given point; a structure mapping is simply the function that describes the transition between any such snapshots (Krivochen, 2018b). From such a position (which we have expanded on in Krivochen and Kosta, 2013; Kosta and Krivochen, 2014a; Krivochen, 2015b, 2016b, c, 2018b), 'mapping' is simply expressed in terms of relations between objects within a workspace or between workspaces, under very specific conditions. Unlike MGG, we do not derive cycles (or 'bounding nodes', 'barriers', or 'phases') from an *a priori* Universal Grammar UG or principles thereof, but from much more basic properties of the (embodied) system, related to the physical and mathematical properties of the neurocognitive substratum from which computation emerges. In the present work, we have attempted to provide an underlying definition of hard 'cycle', or 'domain' for the application of operations, and we have seen that it is possible to derive the ubiquitous nature of locality conditions from the architecture of the cognitive system and its physical properties. Computationally, these cycles are formalized as local domains within which a single kind of computational dependency is found: a structural description combines sub-structures that vary in their computational complexity following local semantic requirements, and in building these sub-structures the computational system oscillates up and down the Chomsky Hierarchy. The ultrametric-metric/high-entropy-low-entropy dynamics we described in **Part III** correlates with the assignment of the computationally simplest structural description to substrings that captures semantic dependencies within local derivational horizons in an interactive, mixed model of computation as presented in **Part II**. As part of our physical inquiry, we argued that no workspace can host an object of dimensionality greater than 3, which equates to saying that 3-D objects are the relevant local domains; let us call them 'islands', generalizing a term originated in Ross (1967)[170]. In Ross' work, the main concern was to establish the structural boundaries for the application of transformational rules which 'moved' phrases; these rules contained variables ranging over sub-strings. The concept (and the empirical evidence), however, is easily adopted outside a narrow transformational model: physical and computational conditions (the object of our inquiry in **Parts II** and **III**) determine local domains for the establishment of relations between objects of whichever kind, and this is a crucial property in human language (i.e., we cannot have just an arbitrary amount of *structure* between any two elements we want to relate). What is more (and this is a point very frequently overlooked), Ross (1967: 494) leaves open the question whether '*islands* [can] *be shown to behave like psycholinguistic entities*', a question we have answered positively in the context of our work. It is important to stress that *'psycholinguistic entities' are not primitives of any kind, but rather emergent properties of the frustrated nonlinear dynamical system we have been studying here, and its interaction with other cognitive domains* (in the particular case of natural language, the so-called C-I and S-M interfaces, without loss of generality). Moreover, it must be noted that the considerations about dimensionality limits that we made in the previous sections

[170] In Ross (1967: 493), islands are defined as "*the maximal domains of rules of the type in question*", which is a formulation compatible with the framework presented in this thesis. We have determined what constitutes a 'maximal' domain topologically in previous sections.

are by no means exclusive of natural language; these are much more basic limits over cognitive operations, as can be hypothesized from the criticism of the allegedly '4-D mazes' experiments in **3.10**.

The present discussion is also intimately related to Uriagereka's developments within the MSO model, since in his recent work (2011, 2012, 2014) he has tried to link the necessity of cycles to 'third factor' (principles independent from the faculty of language) considerations, particularly, *architectural* principles and *efficient computation*. Less vaguely, and leaving aside unnecessary MGG jargon, the idea is that the notion of cycles must derive from physical (via the concept of *dynamical frustration*) and biological constraints over emergent cognitive structures, which arise at the interface between systems (in Uriagereka's model, MSO is motivated at the syntax-morphophonology interface); moreover, whatever model the physical view of cycles yields must also be computationally feasible. Interestingly, in a mixed computational system, there is no need to resort to LCA-like stipulations to motivate structure flattening, which in turn rest on the very problematic concepts of c-command and structural uniformity, against which many voices have risen (including ours, see Krivochen, 2015a, b; also Culicover and Jackendoff, 2005; Jackendoff, 2002; see Emonds, 2007: Chapter 6 for a proposal closer to MGG that still assumes a flat structure for so-called 'semi-lexical heads'; and Epstein, 2000 for some problems with the 'representational' notion of c-command, which led him to propose a no less problematic 'derivational' version of the concept). The physical and computational view of cycles are two sides of the same dynamical, interactive coin.

By hypothesis (although not disconnected from perceptual observations, as seen in the *Intermezzo* above), if a workspace limits its dimensionality to 3, any attempt to establish a dependency with something inside an island from the outside (i.e., probing into an island) would require a higher dimensionality, and the relevant phase space defined by the integral of the relevant function would be a hyperspace, which is not 'thinkable' on its own. We must thus somehow 'translate' it to 3-D at most, performing 'dimensional flattening' operations which imply a loss of information in a dimensional plane: the emergent computational engine we get is not far from Lasnik and Uriagereka's (2012: 21) 'dynamic flatness', and Saddy's (2017) oscillatory architecture. This critical dimensional factor translates directly to a general constraint over the dependencies that can be established between syntactic objects –manifolds- belonging to different derivational spaces. Computationally, we are dealing with the problematic interaction between monotonic and non-monotonic structure building as inputs for mapping operations. Within the MSO model, Uriagereka (2002a: 53-54) phrases the problem in the following way:

> (…) *extraction from a complement can occur within the same derivational cascade, whereas extraction from a noncomplement cannot, given my assumptions. Basically, the following paradox arises. If a noncomplement is spelled out independently from its head, any extraction from the noncomplement will involve material from something that is not even a syntactic object (or, more radically, not even there); thus, it should be as hard as extracting part of a compound (or worse). On the other hand, if the noncomplement is not spelled out in order to allow extraction from it, then it will not be possible to collapse its elements, always assuming that the only procedure for linearization is the command–precedence correspondence that economy considerations sanction.*

Similarly, the kinds of allowed relations are restricted by Epstein (2000: 154) in the following manner:

The First Law: Derivationally Construed

*T_1 can enter into c-command (perhaps, more generally, **syntactic**) relations with T_2 only if there exists NO DERIVATIONAL POINT at which:*

a. T_1 is a term of K_1 ($K_1 \neq T_1$), and

b. T_2 is a term of K_2 ($K_2 \neq T_2$), and

c. There is no K_3 such that K_1 and K_2 are both terms of K_3. (Capital letters in the original, bolds have been added by us)

The key word here, for the purposes of a dynamical system of the kind we have been exploring here, is 'syntactic': recall that 'syntax' in MGG is significantly more restricted than in our proposal, excluding semantics and morphophonology (and, more generally, any other cognitive system, unless considered somehow 'parasitic' on language[171]). In the context of the framework presented in this work, the restriction proposed by Epstein must be relativized along the lines of Feigenson's (2011) concept of '*set*' applied to derivational cascades, if we want locality conditions to have any cognitive plausibility: after all, MGG has long neglected the *implementational level* (Marr, 1982), and the incompatibilities between syntactic theory and psychological evidence (Bever, 1970; also Bresnan, 2017). Being based on more fundamental physical and computational assumptions, Uriagereka's model escapes the objection, at the relevant levels: the definition of derivational cascade can be extended to any conception of symbolic computation (with the appropriate adjustments regarding structural uniformity: opaque domains need not be defined in terms of complement-noncomplements if what matters is the local monotonicity of the structure building mechanism), and it does not preclude the existence of relations between elements *at the interface levels*. On the other hand, the strongest version of an interface-driven theory assumes there are no dependencies *but* those established at the interfaces (Krivochen, 2015a, b, c; Stroik & Putnam, 2013; also the radical MSO model in Uriagereka, 2012). Crucially, if there is no autonomous 'syntactic component', with the so-called interfaces being computational, both approaches turn out to be equivalent.

In the following sections we will go back to the golden early days of locality conditions, attempting a twofold task: on the one hand, we want to distill the 'islands' that have survived to this day, those which appear to have no counterexamples and hold empirically strong. We will analyze apparent counterexamples and challenges to test the strength of the relevant generalizations. We will also attempt to derive strong conditions on 'extraction' (dependency establishment at arbitrary distances) from the topological and computational properties of the objects involved, following the framework we have outlined in the present work.

4.2 An apparent sub-extraction paradigm

A phenomenon which seems to defy the laws of structure mapping (i.e., Move-α; transformations in general) and generally assumed constraints over long-distance dependencies

[171] Chomsky (2009: 26) claims that '*taking visual arrays as lexical items*' is an '*obvious derivative from language*'. More generally, he claims that the mathematical capacity could be parasitic on the language faculty, and that if unbounded Merge is present in other capacities, there should be a genetic instruction to apply it to language.

and movement of constituents is so-called 'sub-extraction'[172]. The phenomenon could be generally defined as the extraction of an XP from a YP which has either been moved or is otherwise opaque as a source for displacement rules. In general, non-complements are regarded as phrase markers from which no material can be extracted (Huang, 1982; Uriagereka, 2002a; Ormazábal et al. 1994), and for moved phrases –i.e., maximal projections- (which always land in a non-complement position) there is an additional constraint pertaining to extraction possibilities from within those moved phrases (Rizzi, 2007):

138) *Criterial Freezing: A phrase meeting a criterion is frozen in place*[173]

In terms of phase theory, where certain syntactic projections act as impenetrable domains for syntactic operations including Agree and Move (Chomsky, 2001, 2008), the freezing constraint is phrased as follows:

139) *Edge Condition: Syntactic Objects in phase edges become internally frozen.*

The Specifier positions of the causative eventive domain *v*P and the clause-boundary marking CP are thus freezing places, whereas the Tense projection TP's status is variable in this respect (if we accept Gallego's 2007 proposal that TP can be a phase via *phase sliding* in languages that display V-to-T movement like Spanish), being as special as IP was for late GB models (see, e.g., Chomsky, 1986a). The customarily assumed clausal skeleton in the Minimalist program goes along the following lines (see, for instance, Lasnik and Uriagereka, 2005):

140) [CP…[TP [Subj]…[*v*P [t_{Subj}] …[VP…[Obj]]]]]

It should be apparent to the reader that these constraints appeal to highly theory-internal concepts, which are not independently justified: *criterion* (which depends on features present in certain functional heads, like [+Wh-] in Complementizer heads), *phase*, *edge* (the specifier and head of a phase projection, *v*P and CP, even though additional apparent phases have appeared in recent years, including DP and PP), and even *phrase*, which here is taken to be an X-bar compatible binarily branched marker (which we have argued against both here and in Krivochen, 2015a; see also Krivochen and Schmerling, 2016 for further arguments). Conditions on structure mapping in MGG have gone more and more narrowly syntactic, leaving aside the interplay with the semantic and morpho-phonological components. The rigidity of the conditions formulated as part of UG forced MGG practitioners to devise more and more conditions to account for exceptions (e.g., Müller's 2011: 198, ff. *melting*; Den Dikken's 2007 *phase extension*; Chomsky's 2008 considerations about loosening the conditions over *Agree*, among others), to the clear detriment of 'minimalism'. It soon became quite unclear how those conditions would follow from a biologically / neurocognitively plausible model (which Ross, 1967: 494 had envisioned), and such concerns would disappear from the theory during the '70s

[172] John Lawler (p.c.) has correctly pointed out that 'sub-extraction seems to me to be a dental procedure'. Not too far away from the truth, particularly given the reactions that the relevant sentences elicit from native speakers. This is not, however, related to Emonds' development of *root transformations*…(pun due to Susan Schmerling and enthusiastically endorsed here).

[173] The freezing principle was, to the best of our knowledge, first proposed by Wexler, Culicover, and Hamburger (1975). Informally, they claimed that '*if the immediate structure of a node in a derived phrase marker is non-base* [i.e., derived by a transformation] *then the node is frozen*'. Moreover, they define 'frozen' as follows: '*if a node X of a phrase marker is frozen, then no node which X dominates may be analysed by a transformation*'. The mention of 'criteria' in Rizzi's formulation pertains, mainly, to the Wh-criterion, developed during the GB era.

and part of the ’80s, the period in which psycholinguistic and theoretical linguistic research were farthest apart. Only in the late ’80s would they return, under a new guise and with increasingly less rigor, in the form of ‘biolinguistics’.

Let us now take a look at an apparent sub-extraction paradigm (examples are taken from Kayne, 1984; and Gallego and Uriagereka, 2007):

141)

a) ?? Which words is learning the spellings of difficult?
b) ? Of which words is learning the spellings difficult?
c) * Which track is considering the length of difficult?
d) ** Of which track is considering the length difficult?

Interestingly, Kuno (1974: 379) and Kayne (1984: 227) mark (141 a) with *, but none of the speakers we have consulted does so; and there are marked differences in acceptability between (141 a) and (141 b) on the one hand, and (141 c) and (141 d) on the other, a fact that is masked if (141 a) is marked with *. Now, we should ask: why are (141 a, b) problematic for freezing theories? Because it is hypothesized that [which words] has been extracted from the phrase [learning the spellings [of [which words]]], which by virtue of being the subject of the main verb [be] is merged in either Spec-VP or Spec-TP. Since either position is a non-complement in X-bar theoretic terms, it should be impossible to extract something from inside a specifier. Things get even worse if the relevant phrase—in this case, [learning the spellings of which words]—has been moved to satisfy a feature of the head to whose specifier position the phrase is moved, this is the case if one assumes that subjects are originally merged within VP and move to TP. In general, it is assumed that the Tense head of TP has a feature called EPP –Extended Projection Principle-, which motivates the Internal / External Merge of an XP to the Spec-position[174]. Within the Minimalist Program (Chomsky, 1995 et seq.), both structure building and mapping (the latter implemented in GB through *Move*-α, latter assumed to be a special case of Merge, namely, ‘internal Merge’, that is, (re)Merge some object that is already part of the derivation) are triggered by so-called formal features, a characteristic MGG shares with HPSG and LFG (although these two do not resort to transformations but take all constructions to be

[174] To this day, the EPP itself remains unmotivated, which makes the whole proposal vacuous. In general, the assignment of features to heads (EPP, Wh-, phi-features, Edge Features, features triggering scrambling, topicalization, etc.) is one of the most controversial issues in MGG, since such features more often than not are mere artefacts to accommodate a particular analysis. As early as 1972, Paul Postal already raised an argument against the use of features like these (referring, specifically, to the assignment of a [Wh-] feature to a PP or just to its dominated NP in a [PP [NP]] structure to selectively ‘account’ for Pied Piping effects):

> (…) *the whole feature-marking proposal has no independent justification. The point is not that descriptive adequacy is unachievable in this way, but rather that it is achievable under the assumption of successive cyclicity only at the cost of having available the overly powerful device of marking arbitrarily selected nodes with arbitrary rule behavior coding features. It is strange that this powerful device should be appealed to by authors who are often at pains to stress the need for restricting the power of syntactic theory, and who have often objected to other approaches on just this ground.* […] *A theory which bans arbitrary syntactic features is stronger than one which allows them, hence to be preferred in the absence of concrete evidence showing the need for weakening the theory, following the principles which Chomsky has long stressed* (Postal, 1972: 215)

Little if any attention was paid to this sensible fragment in subsequent developments of MGG.

base-generated). In MGG, the inventory of features is given by UG, and each natural language groups them in a different way to assemble a Lexicon (see, e.g., Chomsky, 2000: 100). In our opinion, none of this provides an explanation at any level, and the subsequent patches the theory has undergone imply that not even description is easily achieved.[175]

From our point of view, *no linguistic phenomenon has a narrowly syntactic explanation*. This is an obvious derivation of the thesis that language is an emergent property of a frustrated dynamical nonlinear system; and that linguistic expressions are the optimal resolution of a dynamical frustration between semantic and morpho-phonological requirements: rejecting in the architecture the *autonomy of syntax* means that we need to take semantic and morpho-phonological factors into consideration when describing and explaining linguistic phenomena. In the cases presented above, we must consider the lexical and semantic structures and their interplay before making hypotheses about the narrowly syntactic construal (of the kind of the Phase Impenetrability Condition PIC, see Chomsky, 2001: 13-14 for a comparison between *weak* and *strong* versions of the PIC). In agreement with the judgments of Kayne (1984: 189), the native British and US speakers we have consulted find (141 b) better than (141 a) (but, *contra* Kayne's judgments, (141 a) is *not* regarded as ungrammatical). The contrast is clear with examples like (141 c) and (141 d), which are both regarded as completely ungrammatical by the same speakers. Several factors seem to play a role in the availability of extraction from subjects (or, better put, in the availability of dependencies between elements outside sub-trees immediately dominated by the root and elements inside sub-trees immediately dominated by the root; this follows from Chomsky's 1965: 71 definition of *Subject* as daughter-of-S, sister-to-VP): configuration is one of them, but configuration is, in our model, shaped by semantic requirements because *syntactic structure is semantic structure* (see McCawley, 1971: 285, points (1) and (2)). Thus, lexical semantics and compositional semantics cannot be ignored. Nor can questions of derivational timing, involving the relative ordering between adjunction and extraction, and the specific mechanisms by means of which different cycles are glued together must be made explicit. If these mechanisms include (at least) identification of common vector components, admissibility conditions over syntactic structures need to be reformulated for pure configuration is not enough to yield a descriptively adequate theory of locality (Krivochen, 2018b is an attempt to formulate such conditions in an overall model-theoretic framework based on graph theory). We will now refine our theory to account for this contrast.

Let us consider the lexical syntax ('l-syntax', see Hale and Keyser, 1993) of the relevant items (i.e., the syntactic structure of their lexical meaning): the fact that [spelling] has an NP distribution seems to be relevant, insofar as the preposition is actually a 'dummy preposition' inserted for Case reasons (or so goes the traditional MGG argument), of the kind in (142), a well-known nominalization paradigm:

142) a) John robbed the bank

b) The robbery of the bank (by John) / John's robbery of the bank

It is relevant to point out that it seems to be the case that the extracted constituent in (142 a) and (142 b) is actually an *object* –i.e., occupies a Complement position- at the relevant derivational

[175] What is more, problematic data is often violently rejected, as in the case of Everett's analysis of Piraha, Jackendoff's and Gil's analysis of Riau Indonesian, the non-scopal syntax of iteration, counterexamples to Canonical Binding Theory…

step, before the introduction of a procedural element generating a categorial interpretation over the whole structure as either a sortal or an eventive entity[176]. The phrasal structure before the introduction of a categorial-interpretation-inducing procedural element (Determiner for sortality, Tense for eventivity) would therefore be (143), using traditional bracket notation for phrase markers:

143) [$_{XP}$ spell [$_{NP}$ which words]]

where XP is categorially underspecified at both the syntactic and semantic levels: we know it is a nonterminal node, but its identity remains a mystery for the (derivational) time being. The only information we have access to at this point is strictly configurational. Under MSO assumptions, extraction from a complement is always possible, whereas (in the 'radical' version) extraction from a noncomplement is always impossible, because at the derivational point at which displacement occurs, the noncomplement (as it constitutes a separate derivational cascade) *is not even there* (Uriagereka, 2002a: 53-54; 2012: 92-93): [the track] is not so clearly a complement of [length], as the latter is a sortal entity *at all levels of representation* (conceptual as well as lexical), and takes no complements in its l-syntax (Hale and Keyser, 1993 et seq.). The relevant structure would go along the lines of (144), with [of the track] adjoined to NP as an abridged restrictive relative clause:

144) [$_{NP}$ [$_{N}$ length] [$_{NP}$ [of which track]]]]

Under a CS-related analysis, since we are dealing with *objects* (i.e., complements) at the level of CS in (143) (regardless of their specific linguistic instantiation as pseudo-genitives when XP = NP), the displacement of [which words] is possible, whereas we are dealing with a non-complement in (144)[177].

The kind of argument put forth in the previous paragraph assumes an important premise: representationally, *an object O is not inherently opaque; rather, O is opaque for the purposes of operation P at system S*. This means the *derivational timing* of operations is crucial: an operation targeting O' ∈ O might violate some constraint at system / component S, but not at system / component S', which can be a derivationally prior, derivationally posterior, or parallel system to S (this problem is not new at all, see for instance Fillmore, 1963; Koutsoudas, 1972; Ringen, 1972; also discussion in **Section 4.5.1** below). Of course, such considerations do not arise in a strictly 'syntacticocentric', function-based (thus, informationally unidirectional) model like current MGG, but any theory that disrupts the autonomy of the syntactic component in

[176] What we say is that the proposition *rob*(John, the bank) can be linguistically realized by either [T [*v* John [[*v* rob] [V *t* [NP the bank]]]] or [D the [*v* (John) [[*v* rob] [V *t* [NP the bank]]]], and there is no *a priori* way of knowing which the categorial output will be at the derivational point in (143), before the introduction of further procedural information (see also McCawley, 1982a for discussion of the idea that logical-semantic predicates can have more than one surface syntactic realization).

[177] In traditional phrase structural terms, and adopting MGG notation, we have the following situations (assuming, as in GB, that adjuncts are daughters and sisters of XPs):

```
  a. NP                              b. NP
     /   \                              /   \
  NP   [of which tracks]           spell   [which words]
     |
 length
```

favor of interaction and interference, and the (computational / biological) specificity of the structure generation-mapping system will face this 'timing' problem (for a formulation of the 'timing problem' in transformational generative grammar, see Postal, 1972: 211, ff.). In this respect, there are, in our opinion, two ways to proceed:

a) Propose specific filters / constraints for each system and determine their relative timing theorematically[178]

b) Propose *hard* conditions over dependencies that are independent of particular systems, and derive from more basic constraints related to the dynamics of these systems' substratum (given the fact that cognitive computation is embodied)

Here we argue in favor of (b), as should be at least apparent at this point: the formulation of locality conditions in these terms may appear more complicated, but if successful, such conditions tell us quite a lot about basic properties shared by other cognitive systems. Adopting the general format of (b) does not entail giving up descriptive adequacy in the ordering of specific 'rules' or constraints which are thought of as system-specific: for instance, a strong cyclic approach in which adjoined derivations are opaque for further computations derives empirical facts about extraction from parallel monotonically derived units. It is crucial to note that conditions of type (b) are necessarily *hard conditions*, which delimit the phase space for linguistic computation; it is perfectly possible (and perhaps even desirable) that the study of *soft conditions* (constraints whose violation can be repaired derivationally and/or representationally) requires a different approach since we would be dealing with *preferences* within the phase space. In this view, Fillmore's (1963) relative ordering between conjoining and embedding transformations is a specific instance of conditions of the type (b) when looked at from a different perspective: traditional transformational rules (ST transformations) make reference to specific constructions in L and even specific terminal symbols (in the case that a rule is lexically governed), which presupposes some observational adequacy. Hard conditions, defined by the properties of the substratum (as in (b)), are independent of the content of terminal nodes, but sensitive to their configuration (representationally) and derivational history. Formulating constraints over dependencies as in (b) implies adopting Uriagereka's MSO rationale with a twist, since our cycles are defined in a different manner from his. If a dependency can be established between multidimensional fields only when they have an '*interaction kernel of the same dimensionality*' (Scheegans, 2014: 2), for reasons we have outlined above in **Section 3.8 -** related to loss of information- then we predict that 'extraction' is not possible when the source and the target of the mapping operation (the domains *from* and *to which* we are extracting material) differ in their dimensionality at the derivational point at which the extraction is attempted -regardless of the nature of the object to be manipulated. To express it in terms that should be familiar(-ish) already, we can apply a structure mapping operation targeting syntactic objects α and β *iff* (145) –a *delta* function- holds:

145) $\langle \alpha_i \mid \beta_j \rangle$ and $i = j$ (see (127) above)

That is, we can map β (or some $\gamma \in \beta$) to α if and only if the dimensionality of α (the object resulting from the mapping) is the same as the dimensionality of β (the object that undergoes

[178] It is possible that Jackendoff's (2002, 2010) sets of interface rules in the Parallel Architecture can be assimilated to these system-specific filters, but since Jackendoff has not provided an explicit account of islandhood and locality in terms of his interface rules, this note is little more than speculation.

mapping). Computationally, dependencies require coexistence in a workspace: this can be achieved in either of two ways:

a) By having all relevant elements active within a single monotonically assembled (thus, computationally uniform) sub-structure
b) By having some active elements bind the remnants of previous derivational cycles after the application of *adjunction* (Joshi, 1985) as a way to relate separate cycles

Both (a) and (b) follow only if the computational system allows for more than a single kind of dependency to be generated: in this sense, the theory of syntactic-semantic relations is an integral part of the theory of mixed computation. From a theoretical point of view, some advantages of our proposal regarding the definition and inner workings of cycles over MGG's seem to be the following:

146) a. Cycles are derived from fundamental physical properties constraining the computations performed by a given system and limitations derived from the embodied character of such system

b. Conditions over 'displacement' phenomena derive from the topological properties of the objects (manifolds) involved and the computational complexity of the simplest strongly adequate structural description that can be assigned to mini-max portions of structure

c. There is no need to resort to notions like 'subject' or 'object' in defining cycles, which may not be applicable from a cross-linguistic perspective, see Haspelmath (2015) for typological discussion

d. There is no need to resort to X-bar theoretical notions ('specifier', 'complement', 'adjunct'), or to a fixed and static, *aprioristic* structural template

There is, however, a potentially problematic paradigm for a theory of strict locality that we have to pay attention to: Uriagereka (2012: Chapter 2) discusses instances of apparent 'sub-extraction' from subjects, meaning that there is an element (terminal / nonterminal) extracted from what should be a separate derivational cascade (or, representationally speaking, a phrase that is internally frozen). Such movement clearly violates the freezing principle, insofar as subjects either originate as Spec-*v*Ps (and thus, as noncomplements, are opaque to extraction) or must move to Spec-TP to check / discharge T's EPP feature *before* the extraction is performed. The argument pertaining to sub-extraction is based on Uriagereka (1988), where two observations are made with reference to possible sub-extraction instances in Spanish (which apparently extend to other *pro*-drop languages as well):

(I) Sub-extraction from subjects of unaccusative Vs (which are base-generated in a Complement position) is better than sub-extraction from unergatives and transitives

(II) Sub-extraction from post-verbal subjects is better than sub-extraction from pre-verbal ones

(II) can be called into question, particularly because the relevant example Uriagereka uses to illustrate his points (I-II) is the following (2012: 102, ex. (29), judgments are his for (a) and ours for (b)):

147) a. ¿Qué partido [te hizo gritar [(el) que hayas perdido *t*]]?
Which match CL_{ACC2Sg} made shout (the) that $have_{SUBJ2Sg}$ lost
What game did it make you scream that you should have lost?

(Cf. b. *¿Qué partido [[(el) que hayas perdido *t*] te hizo gritar]?)

Uriagereka (2012: 102) finds (147 a) '*not perfect, but* […] *also not as bad as it would be if the subject* [i.e., [(el) que hayas perdido qué partido]] *were in its canonical subject position* [i.e., pre-verbal, although this often depends on verb typology and other factors]'. This is certainly problematic (and we agree with Lappin, Levine, and Johnson, 2000 in that the distinctions in acceptability are not sharp enough to support the theoretical burden), and for us (native speakers of Spanish[179]) (147 a) is not acceptable (or is 'as bad' as its pre-verbal subject counterpart (147 b), for what it's worth), at best, it is at the very border of ungrammaticality. Sure, there are degrees of acceptability (we will deal with this below), but there is also a threshold which is to be determined if we want a theory to be empirically significant. Relevantly, an example like (148), in which the extraction targets the subject of an unaccusative V 'llegar' (these are assumed to be base-generated as Compl-V; see Perlmutter, 1978) is equally anomalous (for the same speakers we presented (147) to, see fn. 179):

148) ??/* ¿Quién te hizo enojar / gritar (el) que haya llegado (??tarde)?
Who CL_{ACC2Sg} made angry / scream (the) that $have_{SUBJ3Sg}$ arrived (late)
Who made you angry / scream (that) (he/she) should have arrived (late)?

The pattern is exactly the same as in (147), but we have substituted the transitive V [perder] 'to lose' for the unaccusative [llegar] 'to arrive', which takes no object, rather, relates a theme and a location. This verb displays a marked preference for post-verbal subjects, unless there is some further qualification to the event (as in 'late') *or* the pre-verbal subject is in fact topicalized. Thus:

148') a. Llegó Juan
b. ? Juan llegó, no Pedro (with neutral intonation)
c. *Juan* llegó, no Pedro (with *Juan* being intonationally prominent)

Under (II), the marginality or straightforward ungrammaticality of (148) is most unexpected, as extraction takes place from a post-verbal position (as shown in 148' a) –and, of course, the NP [el que haya llegado tarde] from which [Quién] is extracted is also post-verbal: we included two VP options, [hacer enojar] and [hacer gritar] because some linguists claim [hacer enojar] is a

---

[179] …and for $n = 25$ other speakers of Andalucian, Euskadi, River Plate, Central and Northern Spain, and Peruvian Spanish… Their judgments ranged from 'no!' to 'horrendous' (a native speaker from Madrid and another from Vigo did not even understand what the sentence was supposed to mean). One of our informants (native speaker of Andalucian Spanish) even suggested that (i) was a better option, though not fully acceptable either:

(i) ¿Qué partido te hizo gritar por haber perdido?
Which match CL_{2SgACC} made scream because to-have lost

Under standard MGG assumptions, (i) should be rendered completely ungrammatical since [por haber perdido] is a causal adjunct (thus subjected to the Adjunct Condition, more recently subsumed under the Condition on Extraction Domains). This, we think, speaks for the dubious nature of (147 a).

factive construction, and thus its complement would be a factive island; that is simply impossible with [hacer gritar]-. These examples, we think, call for a more careful analysis. Such an analysis should, we think, build on a reworking of the notion of *cycle*, and what it really means for a syntactic object to be 'opaque' for purposes of operations at a different cycle.

There is another issue to take into consideration: if the dimensionality-related discussion we have engaged in above actually captures the nature of cycles, the question is not only about what *can* or *cannot* be *said* (and this is not meant in a prescriptive sense, of course, rather, in an 'ability' reading), but about what *can* or *cannot* be *thought*. We are more interested in the *bubbles* (i.e., the empty space) in the 'conceptual sabayon' (see fn. 156) than in the custard itself, since the considerations about physical limitations of the cognitive system restrict the phase space of what is *thinkable*, in terms of the *topological characteristics of the objects that the mental computational system can derive*. If our field formalization is on the right track, then the double integral in (136) should define the relevant phase space, at least for a 3-manifold (since we find a volume below a complex surface).

The following section, as well as the remainder of this thesis, is an explicit argument *against* strictly syntactico-centric *explanantes*, and in favor of a dynamic computational approach that not only oscillates up and down the Chomsky Hierarchy, but also integrates multiple sources of information *during* the computation, in the line of the *interactive* models argued for in **Section 2.5.2** and **2.6**, above. We will see that adopting semantically-driven, dynamical mixed computation as a model for natural language syntax has significant empirical advantages over static, encapsulated, and strictly monotonic approaches.

4.3 Which constraints$_{(i)}$ apply to these sentences and we will try to explain (t_i)?

The following examples constitute a very interesting paradigm for extraction analyses, insofar as they stubbornly resist a strictly syntactic account (and we will see why):

149) a. *Who$_i$ does John like t_i and friends of t_i?
b. Who$_i$ does John like pictures of t_i and books about t_i?
c. *This is the man who$_i$ John saw t_i and t_i kissed Mary
d. This was a movie which$_i$ critics praised t_i but t_i was too violent for my taste
e. *This is the man who t_i killed Bill and Fred murdered t_i
f. This is a problem which$_i$ I cannot solve t_i and t_i drives me insane[180]
g. News reports also previewed Yates's testimony that she had issued an urgent warning to the White House counsel concerning Flynn's contacts with Russia, which$_i$ Flynn lied about t_i and t_i would therefore open him to blackmail.

(149 a) is a classic violation of the Coordinate Structure Constraint (CSC, which we will analyze below), and (149 b), a classic instance of Across-The-Board movement (ATB), both very well documented phenomena in linguistic theorizing. However, the sentences (149 c-g) propose some very interesting challenges to any theory of extraction phenomena, and are rarely taken into consideration in mainstream, orthodox accounts (but see, e.g., Postal, 1997). (149 f)

[180] British native speakers find (149 f) unanimously perfect. Some (southern) US native speakers prefer [but] instead of [and] (weird enough, given the meaning of the sentence), but for all syntactic intents and purposes, our point remains unharmed.

is based on an example by te Velde (2005: 347); (149 g) was taken from *The Washington Post* (May 8, 2017).

The present discussion will frequently refer to Ross (1967), which is in our opinion the best work that has ever been written on displacement phenomena and conditions over them[181] (closely followed by Postal, 1997). In that foundational work, Ross distinguished two kinds of 'transformational' (i.e., structure mapping) rules[182]:

150) a. Rules that operate only *inside* a single clause or between adjacent clauses (e.g.: Dative Shift, Subject Raising). These rules often depend on properties of lexical items involved in the construal (the presence of a ditransitive verb or a raising verb respectively).

b. Rules that operate over variables derived from transformations, establishing dependencies *across* arbitrary chunks of structure. These rules have no *lexical* exceptions.

Rules of the type (150 b) are the ones that motivated the concept of 'island'. Let us review some of the classic constraints derived from the consideration of dependencies of the kind established by rules operating over variables.

As we saw above, there seems to be a general constraint against extracting material from a noncomplement, which was expressed in the GB days in terms of X-bar theory and the Empty Category Principle. The general form of that constraint usually takes the form of something similar to Huang's (1982) Condition on Extraction Domains:

> *Condition on Extraction Domains* (Huang, 1982: 505):
> A phrase *A* may be extracted out of a domain *B* only if *B* is properly governed.

The notion of 'proper government' is a stipulation over representations that we need not review here, but, in a word, the CED as originally formulated, and its subsequent revisions (e.g., within the *Barriers* framework, Chomsky, 1986a), prevents extraction from a noncomplement (i.e., whatever is not a sister node to the head of a phrase in an X-bar theoretic tree). This condition

---

181 It can be tricky to get hold of that work nowadays, unless you know where to look. Do try http://www-personal.umich.edu/~jlawler/haj/Ross_1967.html You're welcome.

182 The interested reader should take a look at Ross' (2012) Ms. '*A preliminary, but fattened, list of transformations*', available at http://www-personal.umich.edu/~jlawler/haj/Preliminarybufattenedlistoftransformations.pdf

subsumed the earlier (Sentential) Subject Condition[183] and Adjunct Condition[184] (as well as some cases of the Wh-island Condition and the Complex NP Constraint, subsumed to the Subjacency Condition[185]), and updated them with GB terminology. However, it was soon realized that it had many problems, at both the theoretical and empirical levels: on the one hand, some 'islands' were selective, insofar as (a) they appeared under certain lexical circumstances (e.g., factive verbs; see Kiparsky and Kiparsky, 1970), and (b) they could be 'repaired' (i.e., their ungrammaticality could be ameliorated) by different means, like ellipsis and sluicing (for a relatively recent and very comprehensive account, see Culicover and Jackendoff, 2005: Chapters 7 and 8), as we can see in (151):

151) a. They want to hire someone who speaks a Balkan language, but I don't know which / *which they do / *which Balkan language$_i$ they want to hire someone who speaks t_i (Wh-Island repair under sluicing; taken from Merchant, 2008: 136)

b. *Who do you regret likes this paper? (Lexically-induced island by a factive V [regret], see Rooryck, 1992. Cf. Who do you believe likes this paper?, with a nonfactive V [believe])

On the other hand, the cross-linguistic validity of the aforementioned constraints and principles over displacement was at best shaky, and some started to take them as generalizations over certain languages (particularly, English) rather than true (representational?) universals (see, e.g., Culicover and Jackendoff, 2005: 328, ff. for some exceptions to the aforementioned constraints; also Postal, 1997 for detailed studies on extraction phenomena and a critical review of commonly assumed conditions on locality). Moreover, even within English, there are some variables a purely syntactic account in terms of 'a relation between X and Y *cannot* be established if…' cannot account for: as an example, de Vries (2017: 7) points out that the acceptability of ATB across complementizer phrases (CP) is influenced by intonational grouping as well as '*strategic*' complementizer placement, which seems to influence the

---

[183] *Sentential Subject Constraint*: (Ross, 1967: 243)
No element dominated by an S may be moved out of that S if that node S is dominated by an NP which is itself immediately dominated by S. [i.e., if the NP is the subject of the outer S].

Chomsky (1973) formulates a more general *Subject Constraint* in the following terms:

No rule can involve X, Y in the structure

…X…[$_\alpha$…Y…]…

Where α is a subject phrase properly containing the minimal major category containing Y.

[184] *Adjunct Condition*: (*apud* Müller, 2011: 56)
Movement must not take place from an XP that has been merged without subcategorization being involved.

[185] *Subjacency Condition*: (Chomsky, 1977: 73)
A cyclic rule cannot move a phrase from position Y to position X (or conversely)
in:

i) …X…[$_\alpha$... [$_\beta$...Y...]…X…

where α and β are cyclic [a.k.a. 'bounding'] nodes [i.e., S/S' or NP] definition of prosodic units and impact on parsing. Verb typology and diathesis also seem to play a role, particularly when the island effects arise not as a matter of narrow syntax, but influenced by so-called 'performance issues'[186] in online parsing, including limits on working memory (de Vries, 1992). Interaction seems key. Thus, for instance, a narrowly syntactic MGG-like approach cannot in principle account for the paradigm in (152) (see also Williams, 1978 for more examples like 152 a):

152) a. */?? This is the man who$_i$ John saw __$_i$ and __$_i$ kissed Mary
b. This is a problem which$_i$ I cannot solve __$_i$ and __$_i$ drives me insane (see also 149 d, although the second V [be] is unaccusative in that example. (149 g) features a transitive V)

The problem here is that the two sentences have an identical syntactic structure, including a relative clause in which the relative operator is the object of a verb and the subject of another, coordinated with the first. Both verbs in the first conjuncts are transitive and take their respective NP complements which are marked with Accusative / Objective Case (thus, no Case theoretic arguments seem to be relevant here), all finite verbs have their subject position satisfied (thus, no EPP / feature valuation arguments are applicable either) in both examples. Yet, (152 a) is ungrammatical (or, more conservatively, highly unacceptable), whereas (152 b) is perfect for British speakers, and 'kinda weird' (but crucially not unacceptable!) for some US speakers (southern varieties specifically). How can this be? De Vries (1992) proposes that in examples like (152 a) the unacceptability arises as a *parsing* problem: a Garden Path effect (i.e., a temporary misparsing) arises because the structure [the man who$_i$ John saw __$_i$ and kissed __$_i$], in which the gap is the object position of a head coordination, and [John] is the subject of both [saw] and [kissed] is available in real-time processing (as an ATB construction), until the disambiguating element [Mary] is introduced (and dooms the initial interpretative hypothesis, yielding an unacceptable if not straightforwardly ungrammatical sentence, 152 a). The ATB parsing is indeed available in real processing time before the disambiguating element is introduced, but '*clashes with the context*' (de Vries, 2017: 6), i.e., it is incompatible with the introduction of [Mary] later in the string. On the contrary, no such Garden Path arises in (152 b), since [a problem which$_i$ I cannot solve __$_i$ and drives __$_i$] is *not* a possible parsing, for it corresponds to *no* interpretation (in other words, the interpretation, whatever that is, is outside the phase space: the corresponding vector points somewhere we cannot go). The violable character of some islands (the fact that some islands can be repaired under deletion, for example) was problematic for a theory of UG based on structural uniformity and function-based mapping between different components (syntax-semantics-morphophonology). Thus, for instance, Merchant (2001) appeals to 'PF deletion' of offending material for the purposes of extraction constraints, then generalizing his proposal to Bare Argument Ellipsis and sluicing. In this respect, Culicover and Jackendoff (2005: 254) explain the origin of the MGG rationale behind the assumption of Movement being the underlying mechanism behind different displacement / deletion phenomena:

> *The view that the mechanism* [underlying BAE, *wh*-questions, sluicing, clefts, and pseudoclefts] *is movement is largely a historical artifact of the order in which the*

[186] In our opinion, the distinction between 'competence' and 'performance' is completely artificial and harmful to linguistics as both a theoretical and an empirical science. This harmful effect is most noticeable within MGG, where 'performance' is a black hole where 'peripheral' cases which principles of UG cannot explain are thrown into without further ado.

constructions were approached: A' constructions were the first to be analyzed, then sluicing, which looks like the remnant of a wh-question (... but I don't know what ~~Harriet's been drinking~~), then BAE [Bare Argument Ellipsis].

A PF-deletion approach like the one adopted by Merchant (and related ones, e.g. Chung et al. 1995) requires the multiplication of empty nodes, and the assumption of 'hidden levels' of interpretation, linked via ordered transformational rules. This kind of solution is acceptable in a transformational theory, but we have defended here (following much tradition in non-transformational grammars, as early as Bresnan, 1978) that it has no cognitive reality and achieve descriptive adequacy only at the cost of multiplying the theoretical primitives. In our opinion, the fundamental error of syntacticocentric approaches to extraction phenomena (without loss of generality) is that they ignore the subtle, but crucial, difference between 'simple' and 'procrustean'[187]: simplicity is desirable as a methodological desideratum, just like the *minimal description length* is the goal to achieve in the encoding of formal language strings. However, just as it is not acceptable for a formal encoding to leave out elements of a string simply for the sake of a shorter description, we will show as this section proceeds that it is not acceptable to try to account for a particular phenomenon in language by appealing to strictly syntactic principles (or, for that matter, strictly semantic, or strictly phonological…): more often than not, there are many aspects of the phenomenon that will be lost, and simplicity will be replaced by 'procrusteanness'. Any linguistic phenomenon is *in essence* an interface phenomenon (we will insist on this point several times when discussing constraints). And interface phenomena are, if Uriagereka and Saddy are correct in their architectural claims, dynamically frustrated.

Of all constraints that have been proposed since the seminal work of Ross, there is one in particular that stands over the rest, both in theoretical elegance and empirical adequacy: the so-called Coordinate Structure Constraint, whose original formulation we reproduce here:

> *The Coordinate Structure Constraint* (Ross, 1967: 89):
> *In a coordinate structure, no conjunct may be* ***moved****, nor may any element contained in a conjunct be* ***moved*** *out of that conjunct.* (emphasis ours. It will be relevant further below)

The empirical strength of the CSC, even after all these years, is remarkable. Postal (1997: 52) calls it 'the most problem-free syntactic constraint ever discovered', and its applicability goes well beyond English (unlike many newer constraints, including most *phase-theoretical* constraints, which need extra-stipulations to accommodate cross-linguistic empirical facts), holding strong even for non-Indoeuropean languages (see, e.g., Georgopoulos, 1985: 87-88). Very much aware of some shortcomings of the general formulation, Ross formulated some qualifications and further specifications over the CSC, which are very often overlooked:

> The main verb of the second conjunct cannot be non-stative (1967: 168)
> e.g. **The tall nurse who Tony has a Fiat and yearns for is cruel to him.*

> The second conjunct cannot be negative (1967: 168)
> e.g. **The shirts which I went to the movies and didn't pick up will cost us a lot of money.*

> The tenses in coordinated clauses must follow a *consecutio temporum* (1967: 169)

[187] Kudos to Susan Schmerling for making us aware of the existence of this word in English.

e.g. **The excellent whisky which I went to the store and have bought was very costly.*

Also, the kinds of variable-affecting transformational rules that are subjected to the CSC are partially specified:

Chopping rules are subject to the constraints (...) *copying rules are not* (1967: 428)

Chopping rules involve a dependency between the displaced constituent and either 'nothing' (a null Ø element in the structural change configuration, see also Postal, 1997: 59), or a syntactically and semantically specified but phonologically null element (represented as *e* for 'empty' back then, later, *t* for 'trace') in the extraction site. Copying rules involve a dependency between the displaced constituent and a resumptive pronoun in the extraction site (either overt or covert), copying the index of the displaced constituent. Technically,

> *If the structural index of a transformation has n terms, a_1, a_2, a_n, it is a* ***reordering*** *transformation if its structural change has any a_i as its k^{th} term, or if a_i is adjoined to its k^{th} term, where $i \neq k$*
>
> *If a transformation reorders a_i, and its structural change substitutes the identity element or some a_k, $i \neq k$, for the i^{th} term of the structural index, the transformation is a* ***chopping*** *transformation. Other reordering transformations are called* ***copying*** *transformations*. (Ross, 1967: 427. Emphasis ours)

Constraints are second-order principles, configuring well-formedness conditions over transformational rules (or their outputs): they quantify over rules rather than over expressions (see Pullum & Scholz, 2001 for a comparison between these two). Let us see some examples involving the transformation of a sentence via a chopping and a copying rule to make the contrast clear (Ross, 1967: 432, ex. 6.153. (c) has been added by us to complete the paradigm. Judgments are Ross'. Indexes have been added for expository purposes only):

153) a. * I spoke to $\_\__i$ about the war yesterday [that guy who's always following us]$_i$ (Heavy NP Shift)
b. I spoke to him$_i$ about the war yesterday, [that guy who's always following us]$_i$ (Right Dislocation)
c. [That guy who's always following us]$_i$, I spoke to him$_i$ about the war yesterday. (Left Dislocation)

(153 a) involves the dependency between a rightwards displaced constituent and null position, whereas (153 b) and (153 c) fill the empty position with a resumptive pronoun (a *copy* of the displaced NP), which makes the displacement rule immune to island constraints thus yielding a 'weak island', or 'selective island', that is, an island that arises under certain contextual conditions (like the presence or absence of an overt resumptive pronoun). Importantly, we will not be concerned with weak/selective islands here because their acceptability depends heavily on dialect, idiolect, specific lexical choices, and other non-systematic variables that fall outside the formal scope of the present thesis. Selective islands are phenomena that appear to be located around points of the phase space with very low probability distributions, and it is thus not usual for base vectors to refer to those states. Yet, *they do belong to the phase space*[188]. The program proposed here strives to *delimit* the phase space (find the places where the system 'breaks',

[188] In aphorism form: *where there's a vector, there's a probability amplitude*. Deep, huh?

which in the long run enables us to make a ‘map’ of the phase space for human language), and therefore we will also be concerned with those elements that are *outside* the realm of what is (and *can be*) human language.

The essential distinction between chopping and copying rules (in terms of the objects they operate over and also the domains they can target) led Ross to propose a reformulation of the CSC (and the rest of constraints) in terms of ‘*no conjunct must be* ***chopped****...*’ (1967: 428. Emphasis ours, confront with the emphasis on the first formulation of the CSC above), which is crucially different from the usual naïve interpretation of the CSC (in the words of Postal, 1997: 59):

> *Extraction from any island (hence from a coordinate island) is absolutely banned.*

This formulation is not quite compatible with Ross’ original argument, and is also arguably *false* on empirical bases: the CSC applies only to *chopped* constituents, not to *copied* ones (and with respect to the former, only under the right conditions, as we specified above). We have to bear this in mind, because some more recent arguments involving the CSC overlook the finer-graded qualifications above, resulting in either *ad hoc* stipulations invoked to defend a false version of the CSC, or triumphalist rhetoric claiming to have disproven it. Both extremes are equally harmful. Postal (1997: 83) further elaborates on the formulation of the CSC in Ross (1967):

> *It seems correct to divide Ross's original formulation of the CSC into separate principles. The one I called the Conjunct Constraint* […] *forbids the extraction of coordinate conjuncts themselves. The other, the CSC, bans (non-ATB) extraction from true conjuncts.*

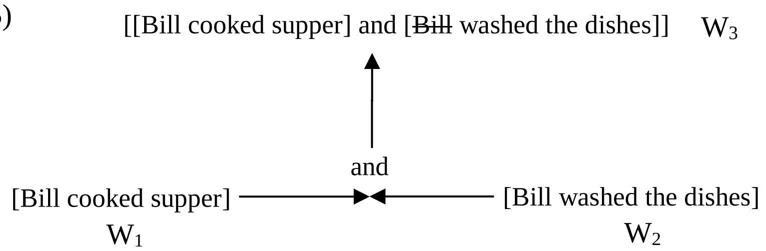
Here we will be mostly concerned with the CSC strictly understood, but we will see that Postal’s distinction becomes relevant when analyzing apparent counterexamples to the constraint.

Relevant examples of the CSC in action include the following:

154) a. Bill cooked [supper]$_i$ and __ washed [the dishes]$_j$.
b. * What$_i$ did Bill cook t_i and wash [the dishes]$_j$?
c. * What$_j$ did Bill cook [supper]$_i$ and wash t_j?

In Krivochen (2015a) we proposed that general extraction constraints follow straightforwardly if a monotonic vs. non-monotonic structure building approach to structure generation is adopted. If each sub-derivation (or ‘cascade’) is generated in separate workspaces, and unified in a third workspace, then constraints on extracting material out of non-monotonic units can be subsumed to their derivational history (Krivochen, 2015b). Graphically, we assumed a derivation like (155) for (154 a) (using similar notation to that of Uriagereka, 1998: 180; 2012: 86):

155)



In the same way we have derived constraints on displacement (focusing on apparent A/A' asymmetries) in Kosta & Krivochen (2014a), the alternation between monotonically and non-monotonically derived syntactic objects can simplify the system of constraints over extractions and gaps in general (including gaps which are not usually considered to derive from movement, like 'sluicing', see Ross, 1969a for discussion and examples; also gaps in existential and unaccusative constructions in Spanish of the kind analyzed in Krivochen and Kosta, 2013: 167, ff.; Krivochen, 2015b). In (155), each sub-derivation is monotonically built, and being exhaustively describable in terms of binary branching tree graphs, they are expressible as finite-state sequences without losing structural information (Uriagereka, 2012: 53; Greibach, 1965). And if FS is enough, by the requirement of strong adequacy in Joshi (1985: 208), we need not go further up. Each sub-structure can thus be represented as in (155'):

(155') a. Bill → cooked → supper
b. Bill → washed → the → dishes

However, the union of those cascades by means of a conjoining transformation (Fillmore, 1963) goes beyond the FS limit, up to CFG. The general format of a *conjoining generalized transformation* is as follows:

$$\left.\begin{matrix} P \\ P' \end{matrix}\right\} \rightarrow P''$$

Where P and P' are *pre-sentences* (structures to which embedding transformations and preliminary singulary transformations have already applied, in Fillmore's cyclic architecture – see below-).

If we *unify* the FS units, the two occurrences of [Bill] are identified (by virtue of their identical base vector components; in the terms we used in Krivochen, 2015b, they are *two tokens of the same type*[189]), and strictly speaking what we have is a multidominated node [Bill], with mother nodes (say, the root S, if S → NP, VP) in both FS cascades (see Krivochen, 2018b for a graph-theoretic approach). The derivation of (154 a) seems to *require* mixed computation. But this is not all: we still need to account for the cases in which a dependency *cannot* be established; the cases in which the CSC does apply.

The exceptions to the CSC are *not* lexically governed (i.e., do not depend on the choice of verb or NP), but depend on syntactic structure (see, e.g., Goodall, 1984; Williams, 1978):

156) a. Bill washed [the dishes]$_i$ and Fred dried [them]$_i$.
b. What$_i$ did Bill wash t_i and Fred dry t_i?
c. John$_i$ ran the race and [John$_i$] won the prize
d. Who$_i$ ran the race and t_i won the prize?
e. Peter likes t_i, and Susan hates t_i, [this book John was talking about]$_i$
(Rightwards ATB, a.k.a. Right Node Raising; see Postal, 1974)

[189] In the *type-token* account assumed here and based on Krivochen (2015b) syntactic terminals are addresses to points in the lexical phase space. This has several advantages over atomic terminals (where the terminal is a lexical item and does not stand for a conceptual-lexical address): as pointed out in Karttunen & Kay (1985: 136), '*relative addresses remain valid even when trees are embedded in other trees; absolute indices would have to be recalculated*'. In a strongly cyclic model where substitution and adjunction play a fundamental role in structure building, this is not a minor advantage.

When dealing with this kind of displacement, which Williams (1978) called 'Across-The-Board' movement (ATB; see De Vries, 2017 for a very comprehensive review), the crucial condition seems to be, descriptively, that gaps are coindexed, as we can see in (156 b); also (and we will return to this below, when considering conditions over dependencies), that the displacement should occur out of 'structurally parallel positions' (Goodall, 1984: 85). In this sense, and as we will see below, even in cases of asymmetric coordination, it is not clear that we can say 'extraction has been performed from the first / second conjunct' (in linear terms), and it is also not a principle (although it is a valid generalization) that ATB rescues extraction from the second conjunct in an asymmetric coordination construction (cf. the *Condition on Asymmetric Coordination* in Na and Huck, 1992: 125[190]). The condition for $i = j$ in the definition of each of the elements involved in the dependency, introduced in (130) in the form of the *Kronecker delta*, should not be alien to the reader at this point: we have seen it is required that there is an identity between *bra* and *ket* vectors in order to have a positive outcome in a polarization or spin experiment[191]. Does that reasoning apply here? Yes, and it is not as far-fetched as the skeptical reader might think: in fact, we are always dealing with mutually orthogonal states. That means, in this case, that a 'gap' (call it *trace*, *copy*, *slash-category*, it is immaterial to the argument) must be assigned one of *n* interpretations ($n > 1$), these interpretations being *mutually exclusive* (as we saw more globally in the case of GPS) and determining the position of the relevant interpretation in the phase space. In other words,

156 b') * What$_i$ did Bill wash t_i and Fred dry t_j? (notice the distinct indexes for the two traces)

is correctly predicted to be out. Out of what? Well, of the relevant phase space, of course[192]. Meaning, (156') is not a possible solution for the system dynamics, in other words, it is not a possible state of the system at any point in time. What are we saying? We are saying that, since language comprehension is heavily based on anticipation (see, e.g., Kamide, 2008; Foucart et al., 2015), and we go from 'left to right' (i.e., we base our expectations on what we have already heard / read), the Wh-element (to use a term familiar to linguists[193], it is really a *ket* vector under the present assumptions) is the way 'we prepare the experiment', and the indexes, 'what we measure'. That is, of course, a metaphor (one of the very few in this work!), but the concept behind it should be clear: as we build structure for an interpretation, we restrict the phase space

[190] Formulated as follows:

> Condition on Asymmetric Conjunction (CAC): *In any asymmetrical conjunction, if extraction is performed on a secondary conjunct, it must be performed across-the-board*.

[191] By the same reasoning, examples like (i), from Goldsmith (1985: 3) are irrelevant for CSC purposes:

i) How many lakes can we destroy and not arouse public antipathy?

Even though it is presented as a counterexample to the CSC, we are unable to see why: we have an intra-clausal movement (in traditional terms), from Compl-[destroy] to the left periphery; the second conjunct presents no extraction whatsoever. At most (even though it can be discussed) we would have chopping of [we] across the board (that is: [we [*t* destroy and *t* not arouse…]]), but that is contemplated by the CSC. See Schmerling (1972) for a careful analysis of apparent counter-examples to the CSC to that date.

[192] …and this is what 'ungrammatical' means here.

[193] …is there any linguist left in the audience at this point? Oh, you, sir. And you, ma'am. Thank you.

where we can find variables, given an operator. 'Indexes' are thus just notations for different locations for the relevant attractor. There is no possibility of finding a solution in the phase space defined by the index *j*, which leaves us with two options: either we force the system towards the attractor that has proven useful (thus yielding the coindexation situation), or we abandon the enterprise altogether, since there is just not enough information to find the relevant attractor, and the function characterizing the behavior of the system is undefined at that point (a situation roughly paraphraseable as 'what on Earth does this mean?', see the comments on fn. 179 above).

Recall that we said, in line with Uriagereka (2011), that the lexical field is topologically ultrametric in its ground state, and so is the workspace in which derivations are performed prior to the introduction of relational elements, which –precisely- disrupt this ultrametricity, yielding a different topological space as a result. Following Uriagereka's insight, we claimed that syntactic relations impose a metric over the lexical attractors involved in a syntactic configuration (i.e., 'metrize' the space). Relevantly, an ultrametric space is Hausdorff if, for X a topological space and *x* and *y* distinct, topologically distinguishable points in X, there exists a neighbourhood[194] *U* of *x* and a neighbourhood *V* of *y*, and $(U \cap V) = \{\emptyset\}$. This is called the 'separation axiom'. The separation axiom holds for ultrametric spaces, as we will see shortly below. Since this aspect of ultrametric spaces is crucial, we will develop this idea a bit. Recall the defining properties of ultrametric spaces:

157) An ultrametric space is a set of points with an associated distance function *d* mapped onto the set of real numbers ℝ such that:

a. $d(x, y) \geq 0$
b. $d(x, y) = 0$ *iff* $x = y$
c. $d(x, y) = d(y, x)$
d. $d(x, z) \leq max\{d(x, y), d(y, z)\}$ (Ultrametric Inequality)

Ultrametric spaces are metrizable, insofar as (158) holds

158) d: X × X = [0, ∞)

see (157 a, b), which amount to the separation axiom. Metrizable spaces inherit topological properties from metric spaces, but might not be metric themselves: pseudometric and ultrametric spaces are *metrizable* (i.e., can have a variable distance function imposed on them), but not *metric* themselves[195]. We argued against a pseudometric approach to the conceptual and phonotactic spaces in **Section 3.2.1** above, based (among other things) on the notion of *distinguishability* (i.e., avoid zero distances between distinct points), which left us with the desirable scenario of an ultrametric topology for the conceptual sabayon and the phonotactic space before the application of syntactic operations. In the ultrametric spaces, different points have disjoint neighborhoods, each at a fixed distance $d > 0$. It might be good to remind the reader at this point that by giving an explicit topological characterization of the spaces, we are

[194] More definitions: for *X* a topological space and *p* a point in X, the neighbourhood of *p* is a subset *U* of *X* that includes an open set *V* containing *p*. In simpler terms, a neighbourhood of *p* is a set containing the point where one can move that point some amount without leaving the set.

[195] More generally, all normal spaces are metrizable. Non-normal (thus, non-metrizable) spaces include some specific kinds of spaces used in algebraic geometry. They are not relevant to our theory.

defining the limits of what is possible and what is not (which has *nothing* to do with a prescriptivist stance over what should or should not be said). That is, the phase space of what is thinkable should be derived from physical constraints involving a dynamical frustration at their core, if the ideas we have been developing here are on the right track.

What does it mean to say that ultrametric spaces are *metrizable*? Well, we kinda derived that above when claiming that syntax imposes a metric over objects, disrupting the ultrametricity of the space. The ground state of cognitive dynamics is defied, we contend, as an ultrametric L-space, and external perturbations change this; 'syntax' is the parametrization of this space which arises from the dynamical frustration existing between the orthogonal planes of *sound* and *meaning* ('expression' and 'content' in Hjelmslevian terms; see also Tesnière, 1959). Now, *if ultrametric spaces are completely Hausdorff* –that is, distinct points have closed disjoint neighborhoods- *then disrupting the ultrametricity of a field also impacts on its Hausdorff characteristic, creating intersections between the neighborhoods of points* (or the set thereof). *A derivation*, we claim, *is the transformation of a completely Hausdorff space with a strong separation condition such that disjoint points have disjoint closed neighbourhoods into a Hausdorff space via syntactic operations over points and their neighborhoods that bring them together*, with topological distinguishability as well as metricity as one of its main characteristics (for a topological introduction, see Willard, 2004). Saddy (2018) explains the process as follows (the quote is kinda long, but it's worth every word):

> *The extension of the manifolds into metric space comes about due to the fact that the distance function between any point x in manifold X and any point y in manifold Y varies as the manifolds get closer and finally intersect. This is a state of affairs that is impossible in ultrametric space* […], *for $x \in X$, $y \in Y$, $d(x, y) = k$ (some constant) in the ultrametric space* […]*; but $k \rightarrow 0$ as the space in which the manifolds exist is deformed. This is a process of metrization. Crucially, the metrization of the initial space is a consequence of the intersection of manifolds.*
>
> *We have thus two operations: (a) the transition from a non-metric space to a metric space, and (b) the transition from high-dimensionality to low-dimensionality. Each is underpinned by different, and potentially independent, mathematical and physical principles, and could, in principle, be analysed in isolation: the CMT is in charge of dimensionality reduction, but it remains agnostic with respect to the metric properties of the space in which the manifolds exist; in contrast, the deformation of the manifolds in ultrametric space brings them closer together and modifies the distance function between them (i.e., between any two points belonging to each of them). Both processes are intimately related and, moreover, they are both necessary to account for cognitive computation and their neurobiological underpinnings.*

It is important to note that the process of metrization and dimensionality reduction is only part of an ongoing dynamical process; manifold X has its own internal dynamics, which carry on after its encounter with manifold Y and *vice versa*. The intersection (or embedding) between the two (or more) manifolds yields a shared extension of the manifolds into a lower dimensional metric space, which gets operated on via normal-grammatical operations, which build structure in the locally cumulative way we have explained above, essentially following (133) until a critical value is reached, then the derivation halts and the resulting object leaves the workspace and gets interpreted. The internal dynamics driving the overall manifolds (X and Y) carry on – time does not stop. Recall we said that the intersection between manifolds is triggered by a perturbation of the ground state of the system, in the form of an external input: this energy input disrupts the ultrametricity of the ground state, and causes manifolds to intersect, triggering the CMT. This intersection yields the lower-dimensional extension we are working with here. Over time, once the external perturbation has been removed or has ended *sponte sua*, the lower dimensional extension is drawn back into the ground state dynamics: any physical system tends towards energy minima and the metric extension of the collided manifolds is an excited state.

There are various advantages in such a topological view: not the least of which is the possibility of defining the notion of 'edge' (or 'periphery') in cycles in a fully explicit way, which in turn pertains to problems of accessibility and intervenience. Note that in an ultrametric space, in which distances do not sum, either everything is an edge or nothing is: the notion of 'edge' becomes trivial. However, as the spaces get parametrized and manifolds are drawn closer, the distance function imposed over the initially ultrametric space (which in turn bought us topological distinguishability plus no inherent selectional bias) ceases to be constant. This means that in a metric manifold we can consider any point *x*, and there will be points *y*, *z*, … in the neighbourhood of *x*, and others which do not belong to the closed neighbourhood of *x* but are *close* to it:

> *If S is a subset of topological space X then a neighbourhood of S is a set V that includes an open set U containing S. It follows that a set V is a neighbourhood of S if and only if it is a neighbourhood of all the points in S. Furthermore, it follows that V is a neighbourhood of S iff S is a subset of the interior of V. The neighbourhood of a point is just a special case of this definition*.
> (https://en.wikipedia.org/wiki/Neighbourhood_(mathematics))

The boundary set of *U* constitutes the *edge* of *U*. Let's see. The *boundary* (we use the topological terminology at this point because *edge* belongs to graph theory) of a subset *S* of a topological space *X* is the set of points which *can be approached both from S and from the outside of S*: this captures the syntactic idea that the *edge* of an object *S* is indeed a part of *S*, but also accessible from outside *S* (by operations targeting that edge from another object). More precisely, it is the set of points in the closure of *S*, not belonging to the interior of *S*. Now, since we are dealing with manifolds, things change, albeit slightly: the *boundary* (which, again, for all syntactic purposes is equivalent to what we would call the *edge* or *periphery* of a syntactic object) of an *n*-manifold with boundary is an $(n - 1)$-manifold. The notion is intuitive enough: the boundary of a 2-manifold (e.g., a sheet of paper) is a 1-manifold (a line). A manifold *M* with boundary is a space containing both *interior* points, which constitute the set of points which are inaccessible for operations outside *M*; and *boundary* points, which can be targeted from outside *M* (see Willard, 2004 for a much more technical discussion of the topological details; also Joshi, 1985 to see the importance of the boundary or edge in defining the operation *adjunction* in TAGs; Schmerling, in press for an argumentation of the crucial place of peripheries in the formulation of formal rules for phonological processes in a modified Montagovian framework). Crucially, every topological manifold is a manifold with a boundary, so the notion of *edge* does not require any *ad hoc* stipulation. We do, however, require the relevant space to be *metric* in order to define the notion, which casts further doubts about the plausilibity of an ultrametric approach to phrase structure as in Roberts (2015) and Uriagereka (2011) (as opposed to ultrametricity as the mathematical model of the ground dynamics of the derivational space, which is what we argue for here). A phrase structure grammar can only define 'core' and 'periphery' if additional assumptions are added: in X-bar theory, *endocentricity* was defined axiomatically, and the early Minimalist Program (Chomsky, 1995: 178) defined the notions of *internal domain* and *checking domain* (basically, the complement and specifier(s)) assuming the notion of head. On the one hand, we need not restrict the discussion to 'heads': we do, however, need to define what we are identifying the edge (boundary set) of. On the other, the set of operations we can apply to edges and local neighbourhoods is very much restricted, which is in principle desirable if we can account for empirical data with these conditions. The linguistic discussion in this **Part IV** will be devoted to an analysis of CSC island conditions under the assumption that strong islands, which define the boundaries of the linguistic phase space, are homomorphic *closed* manifolds: impenetrability follows from independently defined topological notions. Each of these manifolds displays locally uniform computational dependencies, but globally, the whole structure after conjoining and embedding is computationally *mixed*. Moreover, the present framework can accommodate and in fact explain effects of both *locality-as-impenetrability* and *locality-as-intervenience*, in terms of closed manifolds and token-to-token relations –where tokens are defined as eigenvectors-; this is a highly desirable scenario, which deserves further development.

Now, going back to our derivational cascades, each of them has performed this 'completely-Hausdorff-to-Hausdorff / ultrametric-to-metric' topological operation. So, considering the sketch in (155) above, each cascade is a space which, in turn, shares a number of points with the others, call it the *intersection* of the open neighborhoods of points X and Y belonging to each independent cascade. Let us be a bit more explicit here: the derivation of [Bill cooked supper] in W_1 involved the topological transformation of an ultrametric space with a strong separation condition into a metric space by means of imposing a variable distance function between the syntactic objects involved in the derivation, which, crucially, is different from the distance function over the rest of the elements of the lexicon (those left unused). The same happened with [Bill washed the dishes] at W_2. They are then related at another workspace W_3, having been assembled in parallel[196], and, crucially, bearing the same dimensionality (going back to Schneegas, 2014 and our own discussion above). But these cascades share a *type*, that is, they instantiate the same element from the lexicon (Krivochen, 2015b, d for a development of the theory): [Bill]. Of course, there can be a reading in which the Bill that washed the dishes is *not the same* as the Bill who cooked supper (thus the different indexes):

159) a. Bill$_i$ cooked supper and Bill$_j$ washed the dishes
b. * Bill$_i$ cooked supper and __$_j$ washed the dishes
c. # Bill$_i$ cooked supper and Bill$_i$ washed the dishes

In that case, *both* instances of [Bill] would be expected to be materialized (and we start to see a deep connection between materialization and semantics here, of the kind a Y-model does not allow, since both systems –Sensory-Motor and Conceptual-Intentional, in MGG terms- are completely independent), a disjoint reference reading like (159 b) for (159 a) being utterly impossible, and (159 c) being markedly awkward in the coindexed reading (*unless* the relevant interpretation involved some kind of topic contrast, as in 'it was Bill who cooked supper and washed the dishes'; or 'Bill cooked supper and washed the dishes, the rest of the guests did nothing'). Therefore, and nontrivially, both spaces share a point, which is characterized as a perturbation in the lexical field and is defined, topologically, as the *intersection* of the two sets (recall that spaces are sets of points mapped onto $\mathbb{R}$ -the set of real numbers- or $\mathbb{C}$ -the set of complex numbers- with some distance function *d* ranging over *n* points). The dynamical

[196] This is, of course, a move that bears resemblances to Goodall (1984) and his relation of 'parallel structures' in coordination.

frustration between 'form' and 'meaning' can be solved optimally with economy of materialization: the intersection point [Bill] between the two cascades is Spelled-Out *once* (i.e., [Bill cooked supper and __ washed the dishes]), the minimal-maximal number of morphophonological instantiations of the perturbation we need to build a representation at the level of explicature (or construction of a representation of the explicit meaning of a string). This model of multiple derivational cascades with varying degrees of computational complexity might seem more complicated than a uniform, function-based MGG architecture to some readers, but recall we are attempting to have a model that is simple, yet primarily empirically adequate. Simplicity over procrusteanness. Let us now go in a bit more depth into the problem of the phrase structure and semantics of coordination, for they are areas in which the theoretical and empirical advantages of a mixed computation, dynamic approach to phrase markers over MGG's static theory are clear. Once we have analyzed the different kinds of computational dependencies that we find in coordinate structures, we will be better equipped to evaluate the applicability of the CSC to our dynamical model of phrase structure.

4.4 Two kinds of 'and'[197]

Aprioristically assuming that *all* coordinations are structurally uniform and subjected to the *same* kind of constraints (both in terms of the kinds of mapping rules that can apply to them and semantic interpretation and morpho-phonological features -like agreement patterns-) would be falling into a procrustean theory of the same kind we are criticizing[198]. Following Ross (1967), Schmerling (1975: 211-212), among many others, we will broadly distinguish between *symmetric* and *asymmetric* coordination, but add a difference at the level of syntactic structure: the phrase marker (or set thereof) corresponding to symmetric coordination is not the same as the one corresponding to asymmetric coordination, a difference that, naturally, extends to semantic interpretation and morphological agreement patterns (for there are no 'narrowly syntactic' *explanantes* in the context of our framework). We will refer to this difference as *et-coordination* vs. *que-coordination* (adopting Latin terms due to their descriptive resemblance, as we will see shortly). The empirical specifics of this distinction are currently under research (see Krivochen and Schmerling, 2016 for extensive discussion and examples), but we can summarize the main characteristics of each:

Que-coordination:

- Finite-state / regular (Type 3) dependencies between terms
- *In general*, only 2 arguments are related (more on this below)

---

[197] This section owes much to discussions and collaborative work with Susan Schmerling. The usual disclaimers apply.

[198] Postal (1964: 24), while commenting on Wells' (1947) *Immediate Constituent Approach* points out that

> *PSG* [Phrase Structure Grammars] *necessarily assign the wrong P-marker* [phrase marker] *to coordinate structures through arbitrary assignment of excess structure.*

As we see, Postal's warning is not far from Lasnik's more recent insight. Interestingly, MGG has attempted to provide a transformational expansion to PSG to account for different kinds of coordination, instead of entertaining a position like ours, that natural language grammars (in the sense of 'generative devices') may comprise more than a single kind of computational dependency, and furthermore making the generative system sensitive to semantic requirements.

- The arguments are interpreted as a single entity (*perfective*); thus no probing into a conjunct is allowed
- Triggers singular agreement when it is NPs being coordinated due to internal opacity

Et-coordination:

- Phrase-structural (Type 2, perhaps even Type 1) dependencies between terms
- *n*-ary
- Each argument is a separate entity (*infective*), allowing for inner probing into a conjunct
- Triggers plural agreement when it is NPs being coordinated due to accessibility

In Krivochen (2015a: 551) and Krivochen and Schmerling (2016) we analyzed the contrast between (160 a-b) below in terms of different phrase markers corresponding to each NP coordination type (but this also extends to other categories –like VP- as well, as we will see):

160) a. La subida **y** la bajada de la bolsa [preocupan al Gobierno] / [me tienen preocupado] (Spanish)
*The rise **and** the fall of the stock Exchange worry*$_{\textbf{[Pl]}}$ *the government / have me worried.*

b. La subida **y** bajada de la bolsa [preocupa al Gobierno] / [me tiene preocupado]
*The rise **and** fall of the stock Exchange worry*$_{\textbf{[Sg]}}$ *the government / has me worried.*

The argument goes as follows: on the one hand, (160 a) presents a coordination of two separate entities, both semantically and syntactically (each one having its own NP as separate arguments of the V). This has its morphological correlation in the plural verb inflection, which is sensitive to the referential domains delimited by D. (160 a) is an example of *et-coordination*. (160 b), on the other hand, features two entities [subida] and [bajada] which are presented as one, conceptualized as a complex process, roughly paraphraseable as 'fluctuation'. This is an example of *que-coordination*. However, the possibility of such a paraphrasis is by no means a necessary condition, nor does it always hold:

161) a. Red beans and rice is good for you[199]
b. Fish and chips and a smoothie makes for a great meal

Singular V agreement might look weird (or at least unexpected) *prima facie*, but consider that the respective subjects are actually a single conceptual entity (and in these cases, refer to a single dish). In the terms we have been using here, [red beans and rice] and [fish and chips and a smoothie] are *cycles*, constituting a single cascade. 'Hold on!', the reader may say. 'You said *que*-coordination generally admitted 2 terms, and [fish and chips and a smoothie] contains 3!' Does it really? One thing we did not mention above was that que-coordination is a recursive process (in the technical sense), within the limits of *head-tail* recursion imposed over FSL (see Uriagereka, 2008: Chapter 7 for a recent review). Thus, what we have in (161 b) is actually:

[199] We owe Susan Schmerling for most of the English examples of this section, as well as fruitful discussion about their syntax and semantics. All examples have been checked with native speakers (non-linguists) of both American and British varieties of English.

161’) [[fish and chips] and [a smoothie]]

That is, the first member of the coordination in (161 b) has been assembled by means of *que-coordination*, and taken as a single unit for the purposes of the following derivational step (the same operation we saw when considering L-fractals above; also the considerations pertaining to TAGs and the status of placeholders in QLSR), which has yielded the *que-coordination* of two entities ‘fused’ into one, as a perfective whole whose internal structure cannot be probed into. We might add at this point that even though there are semantic analyses of this kind of apparently anomalous agreement pattern (conjoined NPs plus singular V agreement) from a mereological perspective (e.g., Zamparelli, 2011), there is *not always* a part-whole relation between the que-conjuncts, or between the coordinated elements and a ‘whole’, rather, it is the *que-coordination* that triggers the interpretation of the NP as a single entity. An example of syntax-semantics interface at its finest. An eloquent Spanish example is (161”):

161”) a. Se le echa sal y pimienta
SE$_{\text{IMPERS}}$ CL$_{\text{DAT}}$ pour$_{\text{Sg}}$ salt and pepper
b. ??Se le echan sal y pimienta
SE$_{\text{IMPERS}}$ CL$_{\text{DAT}}$ pour$_{\text{Pl}}$ salt and pepper
One pours salt and pepper (on it)

Notice that in (a) the subject-verb agreement morphology is singular, with the plural version (b) being definitely weirder if not completely unacceptable.

A curious feature that characterizes *some* instances of *que-coordination* is their quasi-collocational character (which is, in our opinion, the responsible for the usual limit of 2 elements *per* application of the operation): the order of elements is not exactly fixed and modifiers can be added (as in (162 a), below), but changes in such order often yield constructions with degraded acceptability if we want to maintain V agreement untouched (as in (162 b)):

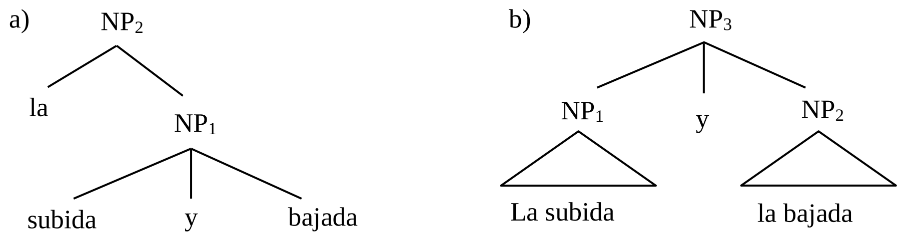
162) a. The abrupt rise and equally abrupt fall of the stock market worries…
b. Chips and fish and a smoothie (all) make/$^{\#/??}$makes…

Interestingly, (162 b) is understood as *et*-coordination in the fully acceptable reading, with each member of the coordination (that is, [chips], [fish], and [a smoothie]) constituting a unit in its own right. The same happens with fixed expressions like [nuts and bolts], and it is clearly seen in other languages as well (e.g., Spanish [papa y huevo], *potato and egg*, the name of a popular side to Argentinian *asado*), for speakers that have those coordinations as *constructions* (in the sense of Goldberg, 2006). *Que*-coordinated expressions of this sort also configure single prosodic domains, such that we get /fiʃənʧips/ (where underline marks primary accent) with an overall falling intonation, but /ʧips/ənfiʃ/ (the *et*-coordinated version), with each term configuring a prosodic domain by itself, and receiving falling intonation independently[200]. These aspects are overlooked in formal semantic analyses, which blame interpretative differences on NP denoting sets of properties, and NP conjunction acting either as set union or set intersection (see, e.g., Heycock and Zamparelli, 2005). These analyses tend to ignore prosodic aspects of the coordination, and assume structural uniformity to hold for the NPs under consideration (interpreted in the X-bar theoretic sense), such that extra formal operations over a cartographical functional hierarchy within the NP (or DP) and the features that compose each functional head

[200] Actually, if we are dealing with an enumeration, all terms but the last one may carry rising tone.

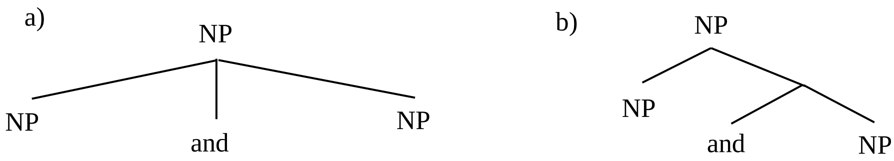
have to be specifically called for (since the basic generative toolbox only has Merge and Agree). These strictly '(generative) formal semantics' analyses are also heavily dependent on strict Fregean compositionality holding for semantic interpretation, which is, in our opinion, too narrow as an approach to meaning in natural languages (see Schmerling, 1975 for an analysis of the interplay between coordination and the 'rules of conversation' proposed in Gricean pragmatics).

How does the *et*- vs. *que*- distinction affect phrase markers, if at all? Following the line of Krivochen (2015a) and the proposal we made there about mixed dependencies in phrase markers (but correcting some technical details), we assume the following classical tree-like representations for the narrow coordinations in (160), without inserting them in a wider phrasal context for the sake of comparison:

163)


Que-coordination *Et-coordination*

There is a clarification point we have to make here: the phrase markers in (163) differ from those we initially proposed for *et*- and *que*-coordination in Krivochen and Schmerling (2016), in which the tree representations were the following (using just NP labels):

163')


The difference in the representation of *et*-coordination makes it evident that trees have enormous limitations as models even at the most basic levels: in Krivochen and Schmerling (2016) we argued that *et*-coordination creates hypotactical relations between conjuncts, in the form of *figure-ground* dynamics. Thus, following a localistic theory, we adopted a Prepositional Phrase as a model, in which Spec-PP is interpreted as a *figure* and Compl-PP is the *ground* (Mateu Fontanals, 2002; Hale and Keyser, 2002, *inter alios*). However, even though (163' b) does capture the localist dynamics and the hypotactic relation between both terms of the *et*-coordination, it forces us to commit to the claim that [and NP] is a constituent, which does not make much sense. Moreover, if we take X-bar theory in all seriousness, we are forced to either say that [and NP] is an &' ('ampersand-bar; or the intermediate projection of the coordinating head; see Progovac, 1999; Kayne, 1994; Chomsky, 2013, for proposals of this sort) constituent, *or* that [and] is the Spec- of the lowest NP, yielding a structure of the kind [NP and [N' [N]]]… can a head occupy a Spec- position at the 'base component' (i.e., after the application of pure structure-building operations, with no mapping involved)? Also, how is that structure supposed to be interpreted semantically? Is NP an argument of [and] (which corresponds to NP as the complement of an [and] head) or is [and] an adjunct of NP (which corresponds to [and] being the Spec- of NP)? The questions do not even make sense outside the narrow margins of MGG.

We have said that *et*-coordination yields a CF kind of dependency, being thus computationally 'higher' than *que*-coordination (which is strictly finite-state), and this is a claim that holds independently of any syntactic formalism: it is a *computational* claim, and as shown in Krivochen and Schmerling (2016), Bravo et al. (2015), and other works, it is not without its empirical consequences.

Topologically, it is interesting to analyze *et-coordination* in terms of *imbedding manifolds*: if a distinction between preliminary simple transformations (including chopping and copying), embedding transformations, and conjoining transformations is drawn (following Fillmore, 1963: 209), conditions on extraction from coordinate structures should optimally be derived from the order in which transformations (i.e., topological mappings and the establishment of dependencies between points) apply. Both embedding and conjoining transformations are generalized transformations: Fillmore recognizes…

> *two types of generalized transformations* […] *those which embed one sentence into another-the embedding transformations, and those which conjoin one sentence to another-the conjoining transformations.* (Fillmore, 1963: 209)

The structural pattern created by embedding transformations is *hypotactic*, whereas *conjoining transformations* create *paratactic p*-markers. Moreover, in the original formulation, *embedding transformations* are context-sensitive (as we see in the general formulation below, minimally adapted from Fillmore, 1963: 212), whereas *conjoining transformations* are context-free:

Embedding transformation:

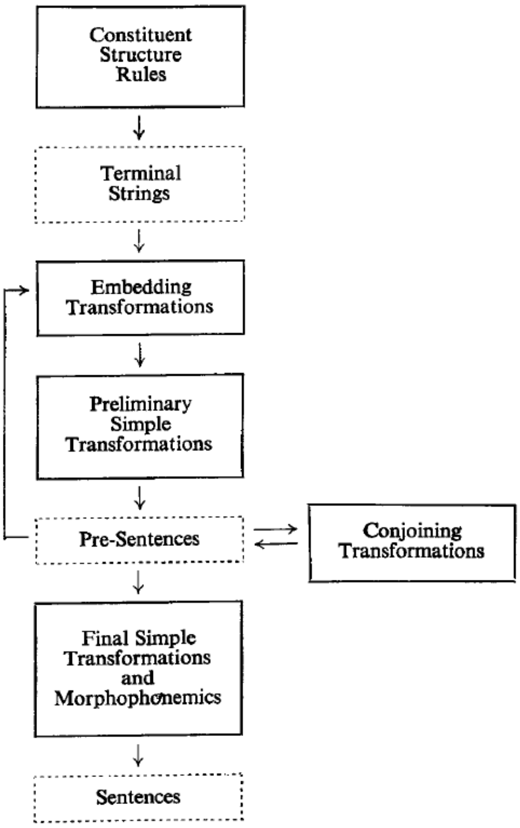
Given P a pre-sentence, A a constant, and a WAY a terminal string,

A → P' in context W…Y

Conjoining transformation (repeated from above):

$$\left.\begin{matrix} P \\ P' \end{matrix}\right\} \rightarrow P''$$

This requires some further argument if we want to take it at face value: Fillmore's argument was made in the light of Chomsky's (1957) complete rejection of Markov processes, and of course didn't consider the possibility of a mixed computational system like the one we propose here and in previous works. Fillmore's ordering implies that the result of an embedding transformation can constitute the input for conjoining transformations within a transformational cycle, but not the other way around (unless we resort to a further cycle): since we know the properties of imbedding mappings applied to *n*-manifolds (thanks to Nash's 1954 imbedding manifold theorem), we have at our disposition a set of explicit mathematical tools to analyze cyclic conditions over syntactic manifolds; and Fillmore's ordering gives us an empirical hypothesis to work with. Let us see Fillmore's architecture (1963: 209), which makes the ordering of transformations explicit like it had not been done before (and very rarely ever since):



Note that it is possible to get a *pre-sentence* to which an *embedding transformation* has already applied, then apply a *conjoining transformation*, thus getting another *pre-sentence*, and make it the input for a second cycle of *embedding transformations*; strictly speaking, we could proceed like that *ad infinitum*. However, if we attempt to minimize the number of mappings, that means we need to minimize the number of transformational cycles in Fillmore's architecture (where a 'transformational cycle' is the path from *embedding transformations* to *pre-sentences*): we want the procedure to *halt* as soon as possible in order to minimize the amount of active structure. Much work pending, provisionally we propose that the CSC amounts, in this model, to a constraint over applying a preliminary simple transformation to a structure derived by means of embedding transformations: if the model is implemented in a system with finite memory, strong cyclic conditions arise from the inner dynamics of the system (a reasoning that is not very far from Uriagereka's MSO at all). The possibility of expressing the *et- / -que* dinstinction in terms of Fillmore's mappings is currently under research (see, e.g., Krivochen & Saddy, 2017): even though we reject the psychological reality and explanatory value of transformations as conceived of in MGG (while recognizing their *descriptive* power; in this we agree with Rogers, 1974: 556 when he says that '*Transderivational constraints, global and interpretive rules, and transformations, it seems to me, don't* <u>*explain*</u> *anything: they describe.*' (underlined in the original)), we do have mappings between spaces and operations which affect manifolds by changing the landscape of the workspace in which derivations take place. The conditions derived from the *et- / -que* distinction arise from these dynamics, and since the coordination types can be recursive procedures (given the possibility of atomization), an empirical approach to 'rule ordering' is a welcomed addition to the formal apparatus. Thus, for instance, (161') requires an ordering between the successive que-coordinations: [fish and chips] are conjoined, and then they are subject as a unit to a further conjoining rule yielding [[fish and chips] and [a smoothie]], a paratactic structure triggering singular V agreement. The segmentation in (161') implies an ordering between both rules: we don't get [[fish] and [chips and a smoothie]], which requires an explicit formulation of the order in which operations apply in a derivation and whether the application of a rule in a cycle either feeds or bleeds the application of another rule in a future cycle (using Kiparsky's 1968: 197-198 terminology): assuming that the procedures for generating *et*- and *que*-coordinated structures do interact, does *et-coordination* feed or bleed *que-coordination*? Does *que-coordination* feed or bleed *et-coordination*? And, how do they interact with chopping and copying? Note that in current MGG these questions –which are eminently *empirical*- cannot even be formulated, even though we are convinced that this kind of questioning is essential in developing an empirically testable model of grammar (almost independently of topological or physical considerations).

Coming back to the internal structure of the sortal entities involved in coordination, if the introduction of the procedural element D comprising sortality is what delimits a nominal cycle (a point that has been argued for in MGG from a completely different perspective; for the original argument concerning the s-projection of N, see Abney, 1987), then it makes sense to have a single D/Q procedural element in (163 a) and as many as we have entities in (163 b) above. Of course, we need to independently justify the presence of such procedural instructions. In our opinion, we are dealing with different *modes of presentation* of entities (be they sortal or eventive, we shall provide more examples below), thus not being determined in a narrow-syntactic way, but depending on the speaker's intention and the information conveyed by CS. In this respect, let us consider two Latin examples, which illustrate the difference between the two kinds of coordination eloquently:

164) Perdiderint cum me duo crimina, carmen et error (Ov. *Tristia* II, 207)
Ruin$_{\text{3PlPastPerf}}$ *with me two crime*$_{\text{Pl}}$, *poem and error*
'Two crimes ruined me, a poem and an error'

165) effodiuntur opes, inritamenta malorum.
Arise$_{\text{3PlPastPerfImpers}}$ *wealth*$_{\text{Pl}}$, *incitements of the-bad*
iamque nocens ferrum ferroque nocentius aurum
and now harmful iron, and iron$_{\text{Dat}}$ *more-harmful gold*
prodierat […] (Ov. *Met.* I, 140-142)
come forth$_{\text{3SgPastPerf}}$
'There arose wealth, incitement of bad things. And now came forth the harmful iron, and gold, (which is) more harmful than iron'

In (164) we see that there is a plural morphological mark in the V ([perdiderint] is a poetic form for the usual prosaic [perdiderunt], such a change of vowel being common in verse), which corresponds to a plural NP subject [crimina]. The following coordination [carmen et error] is an *epexegesis* (a further clarification) of the plural NP [crimina], they both select the same *denotatum* (that is, they both point to the same perturbation in the lexical field[201]). The presence of the coordinate conjunction [et] is revealing in this respect, since we argue that each N, [carmen] and [error] is presented here as a separate entity, which has correlates in a plural N

[201] This implies, of course, that we do not adhere to a theory of reference in which the meaning of a description is an object (as in so-called 'direct reference' theories). As a matter of fact, we do not really *need* a *denotatum* to exist outside the mind, after all, we are dealing with (and operating over) vectors within a cognitive space. Whether the perturbations in the lexical field those vectors point to have some kind of relation to a non-mental entity or not is an independent question, which is more philosophical than anything else and thus will not be addressed here.

[crimina] and plural V agreement. Notice that it would have been possible to use enclitic [que], yielding [carmen errorque], if the author had so desired: verse meter would not have been affected (the hexameter would have remained untouched, and the stress would even have fallen in the same element, [error]). We can therefore assume there is a semantic-pragmatic reason for the choice of [et] over [que], and since phrase structure building follows semantic requirements, the phrase marker we propose corresponds to [carmen et error] will have to make the independence of both sortal entities readable from the structure, *plus co(n)textual factors*.

Interestingly, (165) presents us with a verb in singular form [prodierat] and a coordinated subject, [nocens ferrum ferroque nocentius aurum]. Again, the choice of coordinating conjunction is not derived from strict metrics: [nocens ferrum et ferro nocentius aurum] also yields a well-formed hexameter. Syntactic interpretation, in terms of the structural description we assign to that string, must pay attention to the whole construal and the kind of relations constituents establish among themselves. An 'autonomous syntax' approach like MGG, which is also committed to structural uniformity (Culicover and Jackendoff, 2005: 7), is insensitive to semantics. This entails both an underlying static universal phrase structure template *and* the existence of 'hidden levels' of structure (Op. Cit., 6-7) to accommodate variations of the universal template via transformations (structure mapping operations). Semantically, it is interesting to note that [nocens…] can be grouped under [inritamenta malorum], that is, we can consider the coordination as an *epexegesis* of [opes], which consist of [inritamenta malorum]. The case is not different from that in (164) in this respect, the crucial difference being the way in which entities are presented: as a single complex whole (*que*-coordination, in (164)) or as multiple independent entities (*et*-coordination, in (165)). It is essential to point out that the morphological exponent of the conjunction (in the case of Latin, [et] or [que]) is *not univocally related* to the aspectual, semantic, and computational characteristics of its output: English or Spanish only displays a single morphological element ([and] or [y], respectively), but *both modes of presentation*[202].

The computational and semantic characteristics of the output impact profoundly on extraction possibilities and thus on the applicability domain for constraints over variables: only *et*-coordination yields an infective element which can be probed into. The CSC, which is the center of our attention in this part of the present work, is *only relevant* for *et*-coordinated structures. Why is this so? *Que*-coordination, as argued here and in Krivochen and Schmerling (2016), yields a *perfective* entity, which is opaque for purposes of further computations, including agreement and extraction. Thus, a *que*-coordinated structure [X and Y] computationally yields a terminal node #Z#; thus, we cannot probe into it and extract anything, just as we cannot extract part of a noun or a verb. Let us see an example, involving the *slash* allomorph of que-and, which we have discussed and motivated in Krivochen and Schmerling (2016):

166) British/American law relies heavily on precedent. (to be read, "British slash American law relies heavily on precedent")

As predicted, the examples in (166'), which attempt extraction from the *slash* version of *que*-coordination, are ungrammatical:

[202] The expression *modes of presentation* is crucial, since it inspired the use of aspectual labels to refer to each kind of coordination in Krivochen and Schmerling (2016). After all, aspect is the mode of presentation of an event (see, e.g., Comrie, 1976: 4).

166') *This$_i$ is the law which$_i$/American relies heavily on precedent
*Which law$_i$ does t_i/American rely heavily on precedent?

The argument we want to put forth is *not* that que-coordination violates the CSC; rather, *the CSC is irrelevant for que-coordinated structures due to their perfective character, which yields a single entity that is opaque for further computations* (including extraction, as well as agreement). This opaqueness is consistent with local cycles being computationally uniform and derivations being globally mixed: we can have a que-coordinated (FS) local domain embedded in a bigger CF structure; elements within the FS unit will be inaccessible for operations outside that unit (including extraction targeting a position in the CF structure). Et-coordinated structures, on the other hand, are subjected to the CSC in the cases specified by Ross (1967), and 'repair' strategies (e.g., ATB) are available *only* for et-coordinated structures. Crucially, as we show in Krivochen and Schmerling (2016), the *et-*/*que-* character of a given coordination can be determined independently of extraction paradigms (e.g., by means of anaphora, pronominalization, ellipsis, agreement), which reinforces our view of a non-uniform and non-autonomous (or even 'autistic', as in MGG) computational component.

The *slash*-allomorph of que-coordination seems to be the one that better accepts *n*-ary branching, for instance:

167) Bruce Wayne/Batman/The Dark Knight/The World's Greatest Detective is Gotham's protector

In this case, [Bruce Wayne/Batman/The Dark Knight/The World's Greatest Detective] is taken as a terminal, and also configures a single unit for the purposes of referent assignment. The operation that produces (167) as an output, however, is the same that yielded [fish and chips and a smoothie], with the proviso that in the latter case, as we argued above, we have [fish and chips] as one entity derived by que-coordination, and [a smoothie] as another entity, which are recursively combined via que-coordination themselves. Given the fact that we are dealing with *head-tail* recursion, and not with *center-embedding* (a.k.a. *true recusion*) or anything of the sort, the resulting structure is still within the limits of finite-state languages, and there is no incongruency with our earlier claims. What is more, we predict that no slash-coordinated structure can feature crossing dependencies or center embedding, for those are available only when we go up to CF or (mildly) CS languages[203].

The considerations about *et-* and *que-*coordination we have been making here (and in Krivochen and Schmerling, 2016), lead us to the following provisional formulation of a descriptive constraint over dependencies between variables and operators *across et-coordinated structures* along the following lines, combining our earlier computational and physical discussion:

168) In an *et*-coordinated structure of the form [$_K$...SO$_i$ [and [$_L$...SO$_j$]]], where
- SO$_{(i,\,j)}$ are in parallel structures (where 'parallelism' is defined over semantic-syntactic construal, see McCawley, 1968 and much related work; also Krivochen, under review),

[203] i.e., any and all examples of the forms [*Bruce Wayne$_i$/Batman$_j$/Gotham's millionaire playboy$_i$/The Dark Knight$_j$] (crossing dependency) and [*Bruce Wayne$_i$/Batman$_j$/The Dark Knight$_j$/Gotham's millionaire playboy$_i$] (center-embedding) are ill-formed.

- K and L are terms, and
- L ⊂ K;

a mapped phrase marker [$SO_{(i/j)}$...[$_K$...SO_i [and [$_L$...SO_j]]]] is legitimate *iff* $i = j$[204]

This condition straightforwardly yields the well-formedness of ATB constructions (Williams, 1978), but it also predicts that canonical examples of the applicability of the CSC (see (154) above) are indeed outside the phase space for a specific language. Note that this formulation makes *no reference* to situations of the following kind,

169) [SO_i...[$_K$...SO_i [and [$_L$...]]]]

in which the term [$_L$...], of arbitrary complexity, is left untouched. Let us give an example of such a construction, taken from Goldsmith (1985) –traces and indexes added for expository purposes only-

170) [[$_K$ [How many lakes]$_i$ can [$_K$ we destroy t_i]] and [$_L$ not arouse public antipathy]]?

It is crucial to point out that the nature of the term L is not relevant: Schmerling's (1983: 17-18) argument that [and not] is actually a constituent (a terminal, in the terms we have been managing here, the composition of et-coordination + negation), and thus the possibility that [not arouse…] is not itself a constituent, makes no difference for the purposes of (168). The key factor here is that there is *no variable* (in the sense of Ross, 1967) within term L. Extraction only targets K, and the extended phrase marker after chopping is still K (calling it K' would make no difference: the dependency is still established *within* a single cycle).

Another important consideration pertains to the parallelism between structures (in the terms of Goodall, 1984, 2009). That is, if $i = j$ but the terms are not structurally parallel, or if terms are structurally parallel but $i \neq j$ or we are in the presence of *que*-coordination, (168) does *not apply*. Mind you, dear reader, (168) is a *descriptive* statement, not an explanatory one. *Why* things are that way is something the field-theoretic approach can explain by formalizing the properties of the terms in question, and how their respective eigenvectors are defined. For example, we may require that for some syntactic objects to be 'parallel' in the relevant sense they must be defined in the same dimensions (i.e., they have to have not only the same number of components in their vectors, but also to be defined along the same axes), a requirement that follows from the considerations about Dirac notation and operations over state vectors made in **Section 3.9** above. We may refine this requirement and establish constraints on V typology, or argument vs. adjuncts, but these considerations should optimally be expressible in terms of field interactions. For example, a VP with an adjunct will not have the same dimensionality as a transitive VP, because the adjunct is a separate derivational cascade (thus, a separate manifold) whereas an NP complement in a transitive VP is derived in the same derivational cascade as the V (see Uriagereka, 2002a for discussion).

[204] For the physical justification of this requirement see (130) above and the discussion that follows it. (168) brings together the Address Issue and the Kronecker delta, establishing the $i = j$ condition as a requirement for the Address Issue to be solvable given a structure mapping operation over an *et*-coordinated structure. Crucially, (168) does *not* apply to *que*-coordination because there is no possibility to 'extract' a term, for we have something analogous to a 'lexical item' (Uriagereka, 2002a: 52), an internally unanalyzable object.

In order to further illustrate the requirements in (168), let us look at the following examples (from Na and Huck, 1992: 126; judgments are theirs), which present extraction from either conjunct but whose structures are not parallel:

171) a. Which knee$_i$ did Terry run in these shoes and hurt *t$_i$*?
b. Which shoes$_i$ did Terry run in *t$_i$* and hurt his knee?

Verb typology is relevant for the determination of parallelism: this is thus a notion that cannot be defined by reference to configuration alone, but requires semantic input. The first conjunct has a [V$_{unerg}$ + adjunct] structure, and the NP [these shoes] is within the adjunct. That is, [these shoes] is not an affected object, whereas [his knee] *is*, since the second conjunct has a [V$_{trans}$ + NP] structure. (171 b) presents P stranding, but the non-stranded version (171') is perfect for most speakers as well (and, irrelevantly, preferred by prescriptive grammars):

171') In which shoes$_i$ did Terry run *t$_i$* and hurt his knee?

However, going back to the CSC, the possibility of stranding suggests more *copying* than *chopping* in these examples, because of which the CSC might not apply. Why is this so? Well, if there is the possibility of leaving the P behind, or, rather, of materializing the lower copy of P, it means that *there is something left there*, a copy or a trace. In this sense, a stranded P is no different from a resumptive pronoun: they both require a licensed structural position. Incidentally, (168 a) is less than acceptable for some native speakers, who report understanding what the sentence is *intended* to mean, '*but it's not what it says*'. For a more detailed discussion of the apparent counterexamples to the CSC presented by Lakoff (1968) and Goldsmith (1985), and a careful analysis of the data, see Postal (1997: Chapter 3). In this work, we would like to stress some further points:

- A 'unified' analysis of coordination (be it syntactic or semantic) is not only not possible, but also not desirable from the viewpoints of both theory and empirical analysis
- Only *et*-coordination generates figure-ground (hypotactic) dynamics, corresponding to phrase-structural dependencies
- *Que*-coordination's outputs are limited to strictly paratactic relations, modeled by means of simple Markov chains

The consequences of figure-ground dynamics for *consecutio temporum* and implicatures drawn from Grice's (1975) Mode maxim (such that linearly ordered events are interpreted as ordered in time as well) have been addressed in Krivochen and Schmerling (2016) –building on the seminal discussion about *asymmetric conjunction* and temporal implicatures in Schmerling (1975: 213-215)-, but let us provide a final example, which we have partially analyzed in **Section 3.3**:

172) Bill bought the cookies and John bought the milk, and we all had a wonderful afternoon tea

Here, we have the following propositions (in predicate-first notation):

e_1 = buy(Bill, cookies)

e_2 = buy(John, milk)

e_3 = have(we, afternoon tea)

But these three events do not all stand in a symmetric relation to each other, crucially. Above (I mean, waaay above, in **3.3**) we have claimed that e_1 and e_2 are que-coordinated, and their relation is in this respect symmetric in the simplest (and more accessible) reading: there is no particular inferred order in which the purchase of milk and cookies took place. In relation to the figure-ground dynamics, the situation regarding e_3 is quite different: only *after* (and perhaps even *because of*) the purchase of milk and cookies could we have a wonderful afternoon tea. Thus,

172') $(e_1 \sim e_2) < e_3$ (where ~ is 'not ordered with respect to', and < is 'previous to')

The temporal structure in (172') should betray the syntactic structure we assume: a computationally mixed phrase marker, in which e_1 and e_2 are related by means of certain computational procedures and display FS syntax (strict parataxis), and are then conjoined as a unit with e_3 via CF means (thus yielding a phrase marker which displays global hypotactic relations). Neither e_1 nor e_2 constitute the figure or the ground with respect to the other: as we argued in Krivochen and Schmerling (2016), this seems to be a decisive factor for extraction purposes, because only when there is a figure-ground dynamics (i.e., hypotaxis) can we apply structure mapping operations like *chopping* transformations in Ross' (1967) terms or *gapping* (Ross, 1969a).

Crucially for purposes of the present work, the *et-* / *-que* distinction is not merely a descriptive device, but has correlations in computational and, furthermore, cognitive terms. We claim that the distinction between *et-* and *que-* coordinated elements is analogous to that between *sets* and individual *objects* in Feigenson's (2011) sense. Recall that '*representing sets requires maintaining access to the individual components of the set*' (Feigenson, 2011: 16), which means we can still interpret each member of the coordination as a part of a bigger whole despite the chunking. This is, we claim, the situation with et-coordination, for there is the possibility of inner probing (as we exemplified with morphological agreement). On the contrary, *que*-coordination yields an object that is opaque for the purposes of further operations: an *object* in Feigenson's terms. For a theory that aims at implementational adequacy in the sense of Marr (1982), finding a correlation between different phrase markers displaying varying computational dependencies, semantic differences in natural language paradigms, and categories devised in cognitive research is a very welcomed result.

4.5 Addressing the address issue

The considerations we have been discussing with respect to gap coindexation directly pertain to the address issue, which in more general (and more neutral) terms than those used by Uriagereka (2012: 75), can be thought of as the way to relate elements belonging to different sub-derivations, with no specific reference to any model of phrase structure (and, thus, divorcing the address issue from the LCA)[205]. The issue takes an interesting form under the

[205] It is essential to note that we are trying to *derive* the conditions under which different derivational cascades are related from the dynamics of the system. In strictly formal terms, within TAGs (Joshi, 1985: 209; Joshi and Kroch, 1985: 9) the operation *adjunction* targets an auxiliary tree β and inserts a tree whose top node is labeled X if β contains, *by definition*, one and only one X node. Adjunction thus inserts the tree labeled X at X in β based on label identity. As proven in Joshi (1985), dependencies (*links*) are

present framework: *how does the relevant interpretative system know where to locate the attractor corresponding to a certain perturbation of the lexical field given a gap-like dependency?* If we think about it from the topological perspective we have been putting forth here, the problem is actually a problem of localization, and how *n* elements (say, occurrences of syntactic objects like NPs or VPs) can point to the same location within the phase space. So, for instance, in (173 a-c), all coindexed elements, *qua* vectors, point to the same perturbation in the field[206]:

173) a. [Which picture of himself$_{\mathrm{j}}$]$_{\mathrm{i}}$ did John$_{\mathrm{j}}$ say Mary likes __$_{\mathrm{i}}$ (Uriagereka, 2011: 5, ex. 11)

b. John$_{\mathrm{i}}$ wondered [which picture of himself$_{\mathrm{i/j}}$]$_{\mathrm{k}}$ Bill$_{\mathrm{j}}$ saw __$_{\mathrm{k}}$ (Chomsky, 1995: 205, ex. 36 a)

c. John$_{\mathrm{i}}$ took a picture of himself$_{\mathrm{i}}$/him$_{\mathrm{i}}$ along to the party. (Culicover and Jackendoff, 2005: 221, ex. 57 a)

Let us focus on what seems to be the most complex example, involving both optionality [him]/[himself] and Wh-movement, (173 a). In (173 a), there is a gap in the complement position of [like], coindexed with a complex syntactic object (a nonterminal), [which picture of himself]. This nonterminal, in turn, contains an anaphoric variable [himself], bound by [John]. Each of those elements is an 'address' for a field perturbation (each of which can in turn be the result of operating over several vectors). The problem is: how can we make sure the variables are bound by the right expressions? In other words, why don't we get a structure like (174) for (173 a), in which the indexes have been changed (among other possible combinations)?

174) [Which picture of himself$_{i}$]$_{j}$ did John$_{j}$ say Mary$_{i}$ likes __$_{i}$

If our argument starts from a linguistic structure, there are few options, most of them involving stipulations over phrase structure and the relevant derivational timing for Binding principles to apply (e.g., Chomsky's, 1995: 209 'Preference Principle' stipulating that restrictive predicates

---

preserved in all trees after adjunction, which is crucial for cross-cycle relations. In TAG, the address issue does not arise as a problem, as it is solved in the very definition of the operation *adjunction*.

[206] We must note that, even though examples of this sort are common in the literature on binding (part of which points out the special character of 'picture-Ns' for binding purposes, see e.g. Culicover and Jackendoff, 2005: 466, ff.), they are not accepted as natural by a significant number of speakers. Some Southerner US speakers, for instance, accept the declarative [John$_{i}$ said Mary likes a picture of him$_{i}$/himself$_{i}$], but reject the interrogative version (173 a). Facts in Spanish are also complicated: some native speakers (including myself) accept (i) (for others, it is ?), but (ii) is unanimously rejected (?* / *) by the Peninsular and Latin American speakers we consulted:

i) ¿Qué foto de él$_{i}$ dice Juan$_{i}$ que le gusta a María?
Which picture of him says John that CL like to Mary

ii) ¿Qué foto de sí mismo$_{i}$ dice Juan$_{i}$ que le gusta a María?
Which picture of himself says John that CL like to Mary

A variant of (i) in which the synthetic genitive [suya] is used instead of [de él] is unanimously accepted. It is worth pointing out that [suya] is a pronominal expression, just like [él], but it displays Case morphology. However, our framework can accommodate these facts, as well as those cases in which pronoun/anaphora optionality is available.

must be interpreted in their thematic position at LF). A different approach is necessary, we think. There is thus no single answer to the question raised by Lebeaux (2009), '*Where does binding theory apply*?', an answer that could apply uniformly to all kinds of referential expressions (sortal and eventive), a problem that has been identified in peripheral generative grammar already. Moreover, it is not even obvious how a theory of referential dependencies is related to a theory of binding as formulated by MGG, which we will refer to here (as we have done in previous works) as Canonical Binding Theory (CBT). CBT is based on principles of the kind '*if X is P, then…*[distributional constraint]', where X is a syntactic object and P is a binding-theoretic primitive (anaphora, pronoun, R-expression). As we have already pointed out, our take on the matter is quite different from that in MGG, insofar as *the morphophonological exponent of X as P depends on its local or nonlocal relation to another token of the same type*. Consider for instance the following configuration, using the notation for eventive (Γ) and sortal (Δ) entities we introduced in Krivochen (2015d) -and in **Section 1.2** above-:

175) $[\Delta_1 [\Gamma [\Delta_2]]]$

In (175) we have two sortal variables, with an eventive variable acting as a relational element, bringing the sortal entities together, creating interference between the field perturbations corresponding to each Δ element. Notice that both sortal variables Δ are *tokens* of the same *type* (otherwise, we would have indicated that they were *tokens* of different *types* with *prime notation*, as we did in Krivochen, 2015d), moreover, they both appear within the same cycle: if Δ_1 is a terminal, it increases the dimensionality of the $[\Gamma [\Delta_2]]$ structure by 1, yielding a 3-D manifold (and incidentally for purposes of the present discussion –but crucially for purposes of previous arguments-, closing the cycle and triggering the fading of the relevant dimensional attractor). But let us focus on the dependency between $\{\Delta_1, \Delta_2\}$: how is that supposed to be Spelled-Out (i.e., what kind of morphophonological exponent do we give it for externalization purposes)? In Krivochen (2015b, 2015d) we made a proposal along the following lines, undoubtedly inspired in the Harrisian tradition (also, Lees & Klima, 1964; Ross, 1969b; Langacker, 1969, among many others). Let D be a local derivational cycle, and Σ a structural description:

176) $\forall(\Delta) \mid \Delta \in D$, Spell-Out($\Delta_1$) =
a) Pronoun *iff* $\nexists(\Delta_2) \mid \Delta_2 \in D \;\&\; (\Delta_1 \in D'), D' \subset D$
b) Anaphor *iff* $\exists(\Delta_2) \mid \Delta_2 \in D \;\&\; (\Delta_1 \in D'), D' \subset D$
c) Lexical NP[207] *iff* $\nexists(\Delta_2) \mid \Delta_2 \in \Sigma \;\&\; (\Delta_1 \in D'), D' \subset \Sigma$

Note that we have replaced the concept of 'dominance' between nodes (as in 'XP dominates X', in turn related to the notion of c-command) by the less specific set-theoretic notion of 'belonging': bear in mind that sets may be *ordered* but not *linearized* (and this allows us to formulate structural conditions over ordered sets without the interference of linearity). This should come as no surprise if the reader takes into account the incremental approach to phrase structure we proposed in (123) above, repeated here:

[207] We also defined 'Lexical NPs' as nonterminals of any of the following forms:

i) {Q, N} for Q = *some, any, all, every, two, few, no…*
ii) {D, N} for D = *a, the, my*
iii) {NP, N} for NP = *John's, my mother's, The World's Greatest Detective's…* (note that (iii) is, unlike (i) and (ii), recursively defined)

123) *If a predicate is to have scope over a referential variable (either sortal or eventive), the number of the predicate's coordinates in the mental working area properly contains the number of the arguments' coordinates.*

The approach we put forth is 'locally incremental' insofar as further layers of syntactic structure within a local neighbourhood increase the dimensionality of the manifold until a critical value is reached (and the CMT is spanned), while decreasing entropy (see **Section 3.5.2** above). Moreover, there is also a cross-cycle incremental effect since there are 'traces' or 'remnants' of previous cycles active in a workspace (which as we suggested above take the form of cognitive 'sets', see Feigenson, 2011) for as long as there are unbound variables / operators (in the Koopman / Sportiche sense), as argued in **Section 3.8** in relation to generalized quantification and other global interpretative effects.

In any case, what we claimed in Krivochen (2015d) -and would like to reinforce here- is that the materialization of a variable is sensitive to (but *not* completely determined by) its structural position in relation to other tokens of the same type. Notice that, in grammatical terms, the oversimplified structure (175) could receive the materialization (177):

177) John shaved himself

The question is, why should (176) –or a similar domain-based approach to the syntax-morphophonology interface- hold? Different *types* (e.g., Δ, Δ', Δ''…) are defined by different *ket* vectors over the lexical field, in terms of the location of the perturbations they point to (their respective amplitudes are a different matter, which pertains to lexical semantics in a different way which falls outside the scope of the present work), whereas different *tokens* of the same type (e.g., Δ_1, Δ_2, …Δ_n) are defined by the same *ket* vector, to which different operators apply. In terms of the location of the entity in the conceptual field, both vectors (corresponding to [John] and [himself]) point to the same 'place', but this does not have to map transparently onto the materialization of the relevant variables, meaning that they need not receive the same morpho-phonological exponent, as would be the case in [John shaved John]. The set of possible morpho-phonological exponents for terminal nodes for a language L includes tools to generate more optimal competitors to materializations like [John$_i$ shaved John$_i$] -akin to (159 c), [# Bill$_i$ cooked supper and Bill$_i$ washed the dishes]- which are misleading at best if interpreted without building any implicature, be it a generalized or a particularized conversational inference (Grice, 1975), both of which depend to no small extent on prosody. These tools are so-called 'anaphoric elements', and within Canonical Binding Theory the category includes reflexives (-*self* elements) and reciprocals (like 'each other' / 'one another'), even though it must be noted that these do *not always* have the same distribution. For us, the relevant condition for a variable to be Spelled-Out as one of these guys is that there be *another token of the same type* within a cycle.

Let us go back to (174). Consider first a somewhat simpler, declarative version. Replacing NPs by variable notation, we would get (178):

178) Δ_1 said Δ'_1 liked $\begin{Bmatrix} which \\ a \\ the \\ some \\ \dots \end{Bmatrix}$ picture of $\begin{Bmatrix} \Delta_2 \\ \Delta''_1 \end{Bmatrix}$

Where Δ = John; Δ' = Mary; Δ'' = an exophoric variable (since, before Spell-Out, there is no reason to assume the most embedded sortal variable has to co-refer at all, see (181) below). Let us enrich the representation (202), with locality considerations, shall we? (note, incidentally that we are already out of the computational limits of FSAs, as we need to introduce links in our representation, not unlike Joshi's 1985 annotated TAG) In terms of cycles, we have (179):

179) $$[\Delta_1 \text{ said } \Delta'_1 \text{ liked} \left[\left[\left\{\begin{matrix} which \\ a \\ the \\ some \\ \dots \end{matrix}\right\} \text{ picture of } \left\{\begin{matrix} \Delta_2 \\ '' \\ \Delta_1 \end{matrix}\right\}\right]\right]$$

Cycle II

Cycle I

with each pair of brackets delimiting a cycle (such that *Cycle II* is, in variable notation, [Δ_1 said Δ'_1 liked Δ], with Δ being replaced by *Cycle I* by a mechanism we will expand on in **Section 4.5.1**). As a sortal entity, expanding (recursively) on the basic LF-legible form {D, √}, the [Q picture [of Δ]] structure is a *minimal interpretative unit*, as well as a *maximal generative unit* (i.e., a cycle). If the nontransformational *type-token* theory of displacement explained in Krivochen (2015b) is adopted, then (174) has roughly the structure in (180)[208], in which the cross-cycle dependencies are more evident (and so is, hopefully, the requirement of higher-level computational power for the establishment of such dependencies, see below):

180) $$\left[\left[\underbrace{\left[\left\{\begin{matrix} which \\ a \\ the \\ some \\ \dots \end{matrix}\right\} \text{ picture of } \left\{\begin{matrix} \Delta_2 \\ '' \\ \Delta_1 \end{matrix}\right\}\right]} [\Delta_1 \text{ said } \Delta'_1 \text{ liked} \underbrace{\left[\left[\left\{\begin{matrix} which \\ a \\ the \\ some \\ \dots \end{matrix}\right\} \text{ picture of } \left\{\begin{matrix} \Delta_2 \\ '' \\ \Delta_1 \end{matrix}\right\}\right]\right]}\right]\right]$$

Different structural tokens, same structural type

It should be apparent that there is a problem here. The variable optionality in Cycle I yields the two possible outputs in (181) (see Wasow, 1979):

181) John said Mary liked a picture of $\left\{\begin{matrix} him \\ himself \end{matrix}\right\}$

[208] Notice that the notion of *token* is not limited to terminals. As a matter of fact, a theory with just terminal-tokens would require extra stipulations at the level of generation. Besides, what counts as a terminal might vary: in cases like complex path-of-motion incorporation, for instance, a whole sub-derivation is considered a terminal for the purposes of global interpretation, such that in [I climbed the hill out of town] (in its Path of Motion reading), we have [I #climbed the hill# out of town] for *semantic* purposes (not unlike a Generalized Transformation of sorts, but a TAG *adjunction* perspective would be more accurate). Examples of this kind, in which there is a nonterminal-to-terminal conversion, are in fact quite frequent:

i) The boat [crossed the Atlantic] to Dover
ii) The gizmo comes with an [easy-to-read] guide

Distributionally, also, consider that only the complex terminal can be replaced by #went# or some such verb, no subpart of it can, as we can see in: [*I went the hill out of town], in which #went# only replaces #climbed#. This follows straightforwardly if we bear in mind that terminal nodes cannot be tampered with. We will return to these examples in **Section 4.5.1** below.

with the anaphoric form [himself] unambiguously referring to [John] and the pronominal form [him] referring to either [John] endophorically, or exophorically to a third participant (the Δ''_1 variable in (180)). If the latter, pronominal option is chosen, there is no difficulty. However, if the former, anaphoric option is intended, how do we get to a configuration in which the requirements in (176) hold? Particularly, taking into account that we are dealing with a Multiple Spell-Out model here: is there any point in which Δ_2 and Δ_1 both coexist in a derivational space, such that (in traditional terms) Δ_1 can 'bind' Δ_2? If [which…] is a cycle, its completion will trigger the fading of a dimensional attractor via the center manifold mechanism we explained above, and the whole object will be dimensionally squeezed, to be subsequently subject to interpretation. Can we wait until both cycles have been 'transferred' to establish a dependency that gives us the morphophonological output? No, at least not if the cycles we are handling are defined as in (113 b) physically as sub-domains of locally decreasing entropy in a discontinuous oscillatory function: we need to have some kind of instruction to relate units belonging to different cycles[209] to yield a global interpretation. We face a tension again…

Recall the arguments made in **Section 3.8** above in favor of the existence of low-dimensional remnants of previous derivational steps in a workspace, based on generalized quantification and long distance agreement effects: could something like that apply here? We think it can. Consider the following possibility: Δ_1 does not *directly* bind Δ_2 (for when Δ_1 enters the workspace, Δ_2 is *not there anymore*); rather, Δ_1 binds *a remnant* of Δ_2, which is still active in the workspace. Why would it still be active (or 'survive', in Tom Stroik's words)? Let us make the following assumption: if there is a trace (or a remnant, a residue of some kind) of a previous cycle in a workspace, and that previous cycle (having been transferred as a lower-dimensional manifold) contains an unbound variable, it is a sensible requirement that this variable be bound for all semantic purposes (as well as for determining which morphophological exponent will correspond to that terminal, including Ø). Otherwise, the vector defining the relevant variable just points nowhere within the phase space: then, why select it from the lexicon and introduce it in the construal in the first place (recall our lexical selection and syntactic systems are semantically driven)? The presence of an unbound variable is a superfluous element that implies a departure from the global CS-LF conservation principle ConsP, and in fact has no reason to exist if the type-array used to derive a sentence is selected according to a global conservation *principle* relating CS and LF (or global conservation *tendencies*, the distinction is now immaterial to the argument). Now, let us introduce a further condition: if an *n*-ary relation $\mathfrak{R}$ *can* apply to objects {O, O', O", …O^n} at a derivational point *p*, then it *must* apply. In other words, relations are established *locally* and *as soon as possible*[210]. Why? A possible justification for this position comes from the notion of 'cycle' and its relation to the finiteness of our memory capacities: we want to wipe workspaces clean as soon as we can in order to let new computations take place, and prevent the old ones from consuming resources. As usual (at least in the context of this work), conditions are not absolute: let us assume that we have a local domain D, within which objects O and O' *can* be linked by $\mathfrak{R}$ (coreference, *consecutio temporum*, agreement / concord, you name it). According to the principle outlined above (which

[209] Actually, we need that mechanism anyway: it is basically another way of formulating Uriagereka's *Address Issue*.

[210] Of course, as we will see soon enough, this does *not* mean that once a relation has been established it cannot be changed. Thus, there is no contradiction between saying that relations are established as soon as possible and claiming that there is optionality, because derivations are *not* deterministic or function-based.

we have formulated elsewhere more pompously as 'theoretical possibility equals deontic necessity'), then $\mathfrak{R}$ *must* apply. Now, imagine that O" is introduced in a further derivational step, and it disrupts $\mathfrak{R}$(O, O'), yielding either $\mathfrak{R}$'(O, O") or $\mathfrak{R}$'(O', O"), one of which (choose arbitrarily here, it's just for the sake of the exposition) is preferred, for a given speaker, at a given time. Can we tamper with $\mathfrak{R}$ and replace it by $\mathfrak{R}$' in real time? Of course we can! Otherwise, we would imply the procedure is subject to the conditions over function-based computation we have spent so many pages criticizing, and which has proven empirically inadequate for natural language. Interactive computation allows for this kind of irruption of other systems in the ongoing processes, if there is relevant information that can change the course of the computation. In this context, 'relevance' is, as usual, defined as a ratio between the cost of including more information (or information of different sources or natures) and the cognitive benefits of such inclusion. The kind of computational processes we have argue that underlie natural language are open, non-linear, and bring together different sources of information in real time, apart from what is contained in the 'narrow' linguistic derivation (and what strict Fregean/Chomskyan componentiality can give us, cf. Chomsky, 1965: 136).

Let us assume the paragraph above is somehow on the right track. In our non-transformational *type-token* theory of displacement (but the idea is also valid for the model-theoretic approach in Krivochen, 2018b which dispenses with the MGG remnants of types and tokens from a lexicon), the whole unit [which picture of Δ_1] is 'tokenized' –or 'atomized'- once the cycle is completed (see note 208), and another token of the same structural unit can be introduced in the derivational space 'further up'[211] in the structure; the reasons have been reviewed in Krivochen and Kosta, 2013; Kosta and Krivochen, 2014a; Krivochen, 2015b. To briefly summarize these reasons, we proposed that tokens of a given type are inserted in structural positions in which they generate specific semantic effects required by CS, including presuppositions, topics, focus, and the like), which just means 'later on'. But there is a crucial timing to all this: if the token introduced later in the derivation is merged to satisfy semantic requirements related to the generation of a presupposition-comment dynamics (so that [Which picture of himself did John say Mary likes?] presupposes [Mary likes some picture of John], and more generally, [Mary likes something]), among other things[212], it does so only after lower cycles have been completed. In fact, the representational directionality that is so central a topic in MGG, *top-down* vs. *bottom-up*[213], can be expressed in terms of cycle completion: lower

[211] 'further up', 'in the left / right periphery', are just metaphors to give the reader the feeling we are still dealing with trees (and thus minimize his horror at the ideas presented here). We are actually (and most emphatically) not taking them in any seriousness. The use of 'up' and 'down' terminology makes absolutely no sense in a framework like the one developed here, with a strong focus on topology, with syntactic objects viewed as multidimensional manifolds, and particularly after the development of the idea that completely Hausdorff spaces are transformed into Hausdorff spaces with a weaker separation condition over the neighbourhoods of distinct points via syntax. We beg the reader to bear this in mind.

[212] Meaning, we are concerned here with presuppositions generated by the structure, not by terminals. Thus, the presupposition [there exists an *x* such that Mary(*x*)] and things like those will not be taken into consideration here.

[213] See Zwart (2009), Zeijlstra (2012) for two examples of how the directionality of operations –Merge and Agree, respectively- is still a major issue in Minimalism. Interestingly, the overall historical development of Generative Grammar has shifted from 'top-down' –from nonterminals to terminals- PSR in the early days (with the caveat that Generalized Transformations could give us bottom-up computations as well, but they were soon exiled from the Standard Theory, and even in *Aspects* most of the apparatus is conceived of as top-down) to *strict* 'bottom-up' –from terminals to nonterminals-

objects are cycles that have already been completed by the time an object O enters the workspace, which is the derivational *hic et nunc*. The consequences for the derivation of (174) are direct: in the variable version (180), which we repeat here:

$$180)\ \left[\left[\begin{Bmatrix} which \\ a \\ the \\ some \\ \ldots \end{Bmatrix} \text{ picture of } \begin{Bmatrix} \Delta_2 \\ '' \\ \Delta_1 \end{Bmatrix}\right] [\Delta_1 \text{ said } \Delta'_1 \text{ liked } \left[\begin{Bmatrix} which \\ a \\ the \\ some \\ \ldots \end{Bmatrix} \text{ picture of } \begin{Bmatrix} \Delta_2 \\ '' \\ \Delta_1 \end{Bmatrix}\right]\right]\right]$$

Δ_2 establishes a relation with Δ_1 *as soon as Δ_1 enters the workspace*. But how can it do that, since Δ_2 is in a different cycle than Δ_1? Because, as we said above, $\mathfrak{R}(\Delta_1, \Delta_2)$ involves Δ_1 and *a remnant* of Δ_2, which was left unbound -and, if it is in the workspace, it *must* receive an interpretation, since global conservation *cannot* be violated[214]-. As soon as Δ_1 enters the workspace, being a token of the same type as Δ_2, they are related, being defined by the same vector, as they point to the same attractor in the lexical field. Once the cycle containing Δ_1 is completed (i.e., once it reaches the critical dimensionality value 3), the Spell-Out generalization (176) takes care of the rest as far as the morphophonological component is concerned. Derivationally, this happens *before* the later token of [which…] is introduced. It is also plausible to say that this later token is enriched with information from previous derivational stages that have been already taken by one or more of the interpretative systems: after all, the system has memory, and we would require some sort of stipulation to claim that we have to start over for the interpretation of variables within this derivationally later token. Our system, as we pointed out in **Section 3.10** is essentially cumulative, both in informational and topological terms: each predication relation increases the dimensionality of the syntactic manifold by 1, until a critical point is reached and the system squeezes through a faded dimensional attractor. The oscillatory dynamics that we model (cycles of high dimensionality-ultrametricity to low dimensionality-metricity and back to high-dimensionality through compositionality) crucially depend on the underlying computation being capable of keeping elements active while they are being accessed (e.g., in order to establish a dependency with a member of a term) or while they are undergoing some kind of formal / topological operation (e.g., the modification of the distance function between some number of perturbations in the lexical field). True, the system's tape (probing) memory is not unlimited (thus, we are not in presence of a TM), but it is not null either (thus, we are not in presence of a uniform finite-state automaton). In terms of *memory,* were we constrained by the CH, this situation would seem to require some sort of LBA, for which memory is '*a linear function of the input*' (Uriagereka, 2008: 228); although perhaps we would need something more in specific situations dealing with cross-cycle dependencies. The point is,

---

conditions over Merge in the Minimalist Program (the so-called *Extension Condition*, see Chomsky, 1995: 190; Kitahara, 1997; and Uriagereka, 1998: 258, ff. for a clear presentation).

[214] …and we mean 'cannot' in the strongest possible sense: conservation principles are among the most stable principles in the physical world. Local applications include conservation of angular momentum, conservation of energy, conservation of nucleons after particle decay…the list goes on and on. In language, Harmonic Grammars and Optimality Theory implement so-called 'faithfulness constraints' (McCarthy and Prince, 1995; McCarthy, 2010), which require the GEN(erator) output to be 'faithful' to the input in a sense related to global conservation. See also Lasnik and Uriagereka (2005: 53) and Krivochen (2011: 53) for more linguistic perspectives, focused on lexicon-syntax and syntax-semantics relations respectively.

a single-unit stack does not seem to be enough; we need to be able to probe deeper into active structure.

The system described above (and which we have presented in terms of *strings*), crucially, is equivalent to a phrase marker that folds / warps in real time and which has [which picture of NP] in its boundary (see Martin & Uriagereka, 2014: 175-176; the kind of structure-preserving discontinuity that this system yields owes quite a lot to McCawley, 1982b, 1998 and goes at least as far back as Wells, 1947). What we add to the essential claims of the warping proposal is the determination of the structural positions which act as pivots for the phrase marker to fold. The 'discontinuity-as-type-token' hypothesis, as presented in Krivochen (2015b) and further developed here, provides us with tools to unify different tokens of a structural type as well as determining available structural positions for tokens and a global semantic criterion for structure-building. The warping approach lacks in these departments, but does make predictions with respect to where we would expect to find cross-cycle dependencies: *at the periphery of the relevant cycles* (in TAG terms, at the 'frontier' of the relevant elementary tree). It also dispenses with the need to resort to 'derivational remnants' (since we have only one element and the space has folded in order to have in a local relation with more than one node; this is equivalent also to having a single eigenvector being operated over by several Hermitian operators under different structural conditions, in different neighbourhoods), even though the warping mechanism does need to preserve relations within selected structure (i.e., it needs to be an isomorphism) in order to capture both thematic and criterial semantic effects (and thus the empirical predictions of type-token and warping are in this respect weakly equivalent even though their theoretical foundations are quite different). Some aspects of the combination between these theories are presented in Krivochen (2018b) and Krivochen & Saddy (2016). Discontinuous phrase markers (see the references above) offer a representational perspective on the matter of local and non-local dependencies; the vision presented here incorporates the insights derived from that perspective (involving Right Node Raising, the derivation of parentheticals, etc.) but implements them in a system that is dynamically sensitive to the emergence of cycles (which in a representational theory must be identified by means of categorial labels, like NP and S'). We will not get here into the issue of whether different tokens of the same structural type (and thus defining the same eigenvector) introduced at different derivational points are equivalent to a snapshot in which a single node has more than one mother, but will just assume that they are two sides of the same coin: derivational and representational perspectives on the same phenomenon.

A further qualification is necessary, though, in terms of implementation. Since we are dealing with interactive computation, the input is not exhaustively specified at the beginning of the computation: if external sources of multi-modal information can tamper with the process *in medias res*, we want to leave the door open for extra-tape information to influence the computation. It is possible that such information can be encoded in more parallel tapes, but (assuming that such a solution is actually implementable) there is still the problem of intersecting tapes, which we have noted when analyzing Medeiros' QLSR architecture. Of course, changing the rules in the middle of the game (i.e., adding information of various modalities which was not specified in the input of the process) is not possible for an *a*-machine (the kind analyzed in Turing, 1936 and much subsequent work), and we either enter the realm of operator-controlled *c*-machines, or we abandon the idea that we are working with an automaton performing function-based computations altogether and embrace a dynamical, interactive approach which is orthogonal to the requirements of traditional automata theory. This thesis is a clear plea for the latter, after having analyzed some of the limitations of the former.

Since the system in charge of interpreting the cycles has memory (notice we say nothing about whether those cycles are monotonically built, whether they display one or many kinds of computational dependencies, etc… because those options are in principle all available in our framework), once a variable Δ_x[215] has been affected by an operation, or more generally entered in a relation $\mathfrak{R}$, this information is remembered by the system ($\mathfrak{R}$ might be unary, by the way…it all depends on what we want to do with Δ_x. If we are focusing on language, as we are doing here, it depends on what CS wants Δ_x to do). In the particular case of (180), this means the 'binding' relation between Δ_1 and the remnant Δ_2 is remembered (held active) by the system, and Δ_2's 'binding' status in terms of the configuration in which it appears applies to the derivationally later token of Δ_2 within the [which…] structure merged in the left periphery of the clause.

Summarizing the topological argument, syntactic relations weaken the separation condition over the points in the ground-state ultrametric space, such that from having disjoint closed neighbourhoods for distinct points we have *open* neighbourhoods, and we can thus have connected neighborhoods of points in manifolds, the (set-theoretic) *intersection* of which is the so-called 'gap'. This intersection is the point (or set thereof) toward which the *ket* that defines each element involved in the gap-filler dependency points. In the disjoint reference example, we maintain the Hausdorff property of the space when there is a simpler way to analyze the topology of the object. That, however, is not possible in the case of multiple Wh- questions (like 'Who bought what?' in English), either with or without fronting (depending on the language, that is). If dimensionality increases as we build structure, then two objects at different structural positions *within a derivational cascade* cannot be defined by the same coordinates (or by *ket* vectors with the same dimensions), as this would mean that two different elements are occupying the same position in the phase space, along the lines of [What$_{i/j}$ did Bill wash t_{i+j} and Fred dry t_{i+j}] with a gap or trace corresponding to two mutually orthogonal states of the system *at once* for the purposes of interpretation. But, as we have seen, it can be the case that elements in *different* workspaces or derivational cascades (in our terms and Uriagereka's respectively) are defined by the same coordinates –the same vector-, which means they will be taken as one and the same element when the derivational cascades are unified, because despite having been introduced in the derivation at different points, they *are one and the same element*. Using the same notation as in (155) for derivational cascades, we get the following:

182)

[[Bill washed ~~what~~] and [Fred dried ~~what~~]] W_3

↑

and

[Bill washed what] ⟶ ⟵ [Fred dried what]

W_1 W_2

[215] This should be obvious, but we will say it all the same: we mention Δ variables just for the sake of exposition, but things apply to Γ variables as well. After all, we are always dealing with perturbations of a field, the Δ – Γ distinction being relevant only in terms of linguistic description.

The rule inserting the auxiliary [do] expressing tense and, more generally, auxiliary insertion and inversion is not our concern now, nor is why a token of [what] ends up in the so-called 'left periphery'[216], as we are trying to determine how inter-workspace dependencies might work from a topological perspective under mixed computation assumptions.

Let us now consider a different scenario, which appears to be simpler, as there is no 'Wh-movement' (keyword here: *appears*):

183) John$_i$'s friend$_j$ shaved himself$_{*i/j}$

In (183), we have to dig a bit into the structure of the first NP to find the antecedent for [himself], just like we did with (174). *Prima facie*, it would seem that the antecedent for the anaphora is the closest element in the linear order. A simple tweak to (183) can easily disprove that:

184) A friend$_j$ of John$_i$'s shaved himself$_{*i/j}$

Here, [John] is linearly closer to the anaphor [himself] than [friend], yet the dependency between [friend] and [himself] remains untouched. Can we account for this? How does this happen? If the way in which we accounted for dependencies above is on the right track, then the explanation goes along the same lines: both [friend] and [himself] point to the same location in the lexical field at the moment a dependency is established. Now, why is [John] not an 'intervenient' element for that dependency (in the sense of Rizzi, 1990, for instance)? Saying that it is just defined by a different vector would not be a complete or satisfactory answer (and yes, it also seems circular). The fundamental question remains: why does an element defined by a different vector not count as a possible 'target' for the purposes of establishing a dependency? We have to take at least two factors into consideration:

a) Whether the 'intervening' object is in the same workspace as the elements involved in the relevant dependency

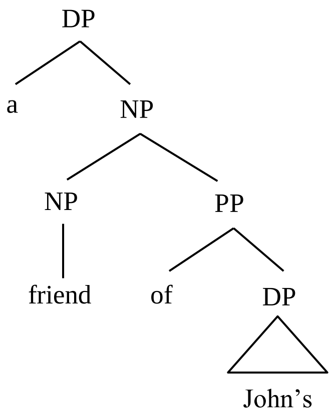
b) Whether the components of the vector are defined in the same number of dimensions and indeed the same dimensions

Factor (a) is the main locality condition in a MSO model, cross-cascade dependencies being directly dependent on the solution to the Address Issue and whether the system can relate separate cascades, whereas (b) refers to our Token Collapse principle in Krivochen (2015b: 279). We have proposed in this work a solution to the Issue when dealing with the ATB and *Wh*-movement examples in (156) ('What$_i$ did Bill wash t_i and Fred dry t_i?' and so on). The two sets of examples, however, display similar syntactic properties: there is CF 'displacement' of a constituent of arbitrary complexity, and this displacement takes place across what a strict version of MSO would define as a boundary for the establishment of a dependency, because we are dealing with globally non-monotonic structures. (183-184) are classic examples of binding that are straightforwardly characterized by Principle A of Binding Theory (notice that we have not said 'explained', since Binding Theory is ultimately said to derive from UG, and has thus no real explanatory value unless UG is itself proven to exist outside the theory that assumes its existence as an axiom). The noun phrase [John] does not c-command the anaphoric element

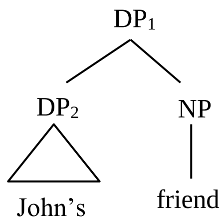
[216] We have discussed the latter in detail in Krivochen (2015b); Kosta and Krivochen (2014); Krivochen and Kosta (2013: Chapter 4); we assume the motivation to be related to the generation of semantic effects, including entailments, presuppositions, and the like.

[himself], nor does the prepositional phrase [of John's]. In both cases, however, the NP projection of [friend] c-commands [John] (and, thus, is more accessible as an antecedent for [himself] than the more embedded, c-commanded [John]), as we can see when comparing the X-bar syntactic trees for the relevant DPs[217,218]:

185) a)



b)



Both DP structures (185 a) and (185 b) would be located in Spec-TP, thus c-commanding the VP projection containing [himself]. The chain of arguments leading to the relevant binding dependencies in MGG goes as follows (Chomsky, 1981a, 1986, 1995; Reinhart, 1976; among many others):

I) α *c-commands* β *iff*:
a) α does not dominate β, and
b) the first branching node that dominates α also dominates β.

I) α *binds* β *iff* :
a) α *c-commands* β, and
b) α and β are *coindexed*.

III) An NP (β) is *A(rgument)-bound iff* it is coindexed with a higher NP (α) in either a subject or object position.

A further refinement of (I) is necessary in this case, as [John] c-commands [friend] in (185 b), and thus is more accessible for the purpose of further operations (Aoun and Sportiche, 1983; Chomsky, 1986a):

I') α *m-commands* β *iff*:

a) α does not dominate β,

b) β does not dominate α, and

c) the maximal projection of α dominates β

---

[217] We will not deal with the nuances of the syntax of the Saxon genitive ['s] here, since they are immaterial to the argument insofar as the c-command relations remain untouched. See Abney (1987) for some discussion.

[218] We have assumed that [of John] is an abridged restrictive relative clause, thus adjoined to the maximal projection of N (call it N" or NP). The reason is that [friend] has no [_ + NP] subcategorization rule, thus, takes no complement. Any constituent that depends on it therefore has thus to be an adjunct.

As we can deduce from these definitions, [friend] always either c-commands or m-commands [John] (note that DP_1 is the maximal projection of [friend], which takes DP_2 as its specifier), which is sufficient for Principle A to hold if we replace 'c-commands' in (II) by 'm-commands'; thus 'explaining' the dependency between [friend] and [himself]. However, a MSO model cannot omit the fact that in the MGG representation (185 b) [friend] is in turn in a different cascade from [himself], being contained in a Specifier, these being command units in the sense of Uriagereka (2002b): if this was so, then the antecedent would be inaccessible, even if the processor in charge of establishing a dependency had unlimited tape memory (i.e., if we were modelling the dependency in a Turing Machine), because under the more radical assumptions, the two phrase markers *never co-exist* in the same workspace. Back to square one of the Address Issue, then? Not quite. Let us cite the Issue again:

> <u>Address Issue:</u>
>
> *Whenever a phrase marker K is divided into complex sub-components L and M, for K to meet LCA conditions of Multiple Spell-Out, the daughter phrase-marker M that spells-out must correspond to an identical term M within K* (Uriagereka, 2012: 75)

Note that if the mechanism that we used for (174) applies here as well (and there is no reason to assume otherwise), we actually have that 'identical term' Uriagereka speaks about (and which, as we have seen, is there by definition in TAGs): two vectors point to identical perturbations in the lexical field. As a matter of fact, we would need an additional stipulation to prevent the interpretative procedures from relating two constituents defined by the same vector, as they are basically *the same thing*, all that differs is the syntactic context in which they appear. The non-interveniency of [John] for binding purposes is thus only *partially* due to structural reasons, more so for Uriagereka's MSO model than for ours, as only the former defines cycles exclusively with reference to conditions over phrase markers and their possibilities for linearization following the LCA. In both cases (185), [John] is dominated by a projection that establishes a point of symmetry (i.e., mutual c-command) with the cascade containing [friend] ({NP, PP} in (a) and {DP_2, NP} in (b)), which in Uriagereka's (2002a) model *triggers* the Spell-Out of the LCA-offending phrase marker and eliminates the symmetry point.

Our proposal has two aspects: on the one hand, we have a critical value for phrase marker (*qua manifolds*) dimensionality, which is cumulatively built via *predicate-argument* relations in a metric space (our claim (123)). On the other, the sound and meaning systems *have access to the workspace where derivations take place*, and once a fully interpretable unit for either of them has been assembled, that component takes it, leaving a remnant –a derivational crumb- behind (see **Section 3.8** for discussion and consequences of this approach). It is also critical that, *by definition, identical points have identical neighborhoods*; if we define the points in the lexical field(s) towards which the vectors corresponding to [friend] and [himself] point in terms of their neighborhoods, those must be identical, because *the points themselves are identical*, with the distance function *d*(friend, himself) = 0 *at every point in the derivation*. This means that, while syntactic operations modify the distance function between any two distinct points making their neighborhoods intersect by changing the topology of the workspace by weakening the separation condition, *this only happens for distinct points*. What happens in cases like (186), though?

186) a. John and Mary like each other
b. John and Mary like themselves

In the present proposal, [John] and [Mary] are *et*-coordinated (notice that the coordination [John and Mary] triggers plural agreement, and can be probed into by anaphoric variables), which means they interfere. If [John] and [Mary] are each defined by a *ket* with specific values for each of the dimensional components, the result of *et*-coordinating them must be expressed as the combination between those vectors. Obviously, as none is the *bra* of the other, the *inner product* operation does not apply for defining the vector corresponding to their combination. The structure [John and Mary] points to a region of the phase space that is 'somehow' related to the inner components of the *et*-coordination: 'somehow' is here *the cross product of the ket vectors defining the terms of the et-coordination*. Thus, if [John] = $|\alpha\rangle$ and [Mary] = $|\beta\rangle$, [John and Mary] = $|\alpha\rangle \times |\beta\rangle$; let us call the result of this cross product $|\gamma\rangle$. Now, it is crucial to remember that this kind of operations is only possible if the vectors have '*an interaction kernel of the same dimensionality*' (Schneegans, 2014: 2), which is precisely factor (b) above. In general, we assume that the cross product, which results in a vector rather than a scalar, is the way to represent compositional interactions among vectors, for *n* vectors defined by the same *number* of components and the same dimensions, but of course not necessarily the same *value* for each component. This is intended to capture the conjecture that the result of field interaction is compositional (a point we share with Uriagereka, 2011).

Now, we have to look at what happens with (186 a), in which the following relations hold:

186') a. like(John, Mary)

b. like(Mary, John)

Following the sequence of the Greek alphabet, let us call the vector defining (186' a) $|\delta\rangle$, and the one defining (186' b), $|\varepsilon\rangle$. It must be noted at this point that $|\delta\rangle$ and $|\varepsilon\rangle$ need not be defined via the same number of dimensional components as $|\alpha\rangle$ and $|\beta\rangle$ (recall that *ket* vectors are column vectors with as many components as dimensions we are working with), as a matter of fact, if our cumulative structure building proposal is correct, we predict $|\delta\rangle$ and $|\varepsilon\rangle$ to be defined in a higher-dimensional space.

Similarly, for (186 b), we have the following sub-predication structures:

186") a. like(John, John)

b. like(Mary, Mary)

And we will also give each of these a vector: $|\zeta\rangle$ for (186" a) and $|\eta\rangle$ for (186" b).

We would like to propose that there is no difference with respect to the vector operations involved between reflexives and reciprocals (which are always *cross products*), at least in general, but the differences arise *with respect to what each vector represents*. In variable notation, we have the following:

$|\delta\rangle$ corresponds to the predication structure $\Gamma(\Delta, \Delta')$

$|\varepsilon\rangle$ corresponds to the predication structure $\Gamma(\Delta', \Delta)$

However, $|\zeta\rangle$ and $|\eta\rangle$ present us with a different scenario:

$|\zeta\rangle$ corresponds to the predication structure $\Gamma(\Delta_1, \Delta_2]$

$|\eta\rangle$ corresponds to the predication structure $\Gamma(\Delta'_1, \Delta'_2)$

The componential result of each structure points toward a different attractor within the phase space, because the vectors involved in the cross product are not equivalent. In one case (186 b), there is a relation between two tokens of the same type for each sub-predication structure, in the other, (186 a), each sub-predication structure involves a token of different types (indicated, as above, via *prime* notation). Importantly, should we have the two examples of (186) as observables, only (186 b) could be captured by means of a *delta* function (we have worked here with the *Kronecker delta*, as we have modelled discrete states), for there is an identity requirement (such that $John_i$ must like $John_i$, and $Mary_j$ must like $Mary_j$) for the sentence to make sense. If we cross the indexes, or add tokens of other types (e.g., if $John_i$ likes $John_k$ and $Mary_j$ likes $Mary_l$), (186 b) ceases to be a suitable linguistic instantiation of such a structure.

Going back to the Address Issue and the problem of *how to connect antecedents and consequents* (or *gaps* and *fillers*), within our theory there is simply no cause for combinatory operations to relate (i.e., 'make interfere') *n* perturbations of the lexical field that are identical because, by definition, they are 'already related'; thus, the topological transformation that weakens the separation condition over the neighbourhoods of distinct points does not affect the definition of those perturbations in terms of their dimensional components. We would require some extra *ad hoc* stipulation to prevent the establishment of a dependency between tokens of the same type defined in such a way. However, as we have seen when referring to De Vries' work and the paradigm in (149), purely topological or formal considerations (more widely speaking) are not enough to account for all relevant extraction cases under the CSC. An interface approach is, in our opinion, necessary and ultimately unavoidable.

4.5.1 Derivational units from another dimension

In this subsection we will address an issue that we mentioned only in passing in a footnote (fn. 208): how internally complex structures derived in a workspace get inserted into a structure derived in a parallel workspace as internally opaque units, interpreted as lexical terminals. We refer to this process as *atomization* of a structure: a phrase marker of arbitrary complexity derived in a workspace is adjoined to a parallel structure via *embedding* or *conjoining* (in the senses of Fillmore, 1963) as an opaque unit for purposes of operations at the target of adjunction (including processes as heterogeneous as extraction and inflection). Note that the atomization of a complex derivational unit rendering that unit internally opaque is weakly equivalent to a rule ordering hypothesis in the sense of Fillmore and also Koutsoudas (1972), Ringen (1972), among others: singulary transformations targeting elements within a syntactic object O *must* be ordered *before* atomization and adjunction of O, otherwise the resulting object will be illegitimate. In this section we will analyze in some more detail cases in which derivational units can no longer be probed into after their insertion in a wider phrasal context. Let us repeat the relevant examples and add some new ones:

187) a. Bruce was sitting in his studio, when, suddenly, a bat broke the window into the room

b. Yusei also broke the window into the room and quickly set up his duel disk. (from www.janime.biz/5DS/series054.html)

c. The boat crossed the Atlantic to Dover

d. As I climbed the hill out of town, I noticed a crowd of pupils standing in front of their school at the side of the road. (from www.tag.wordaligned.org/posts/tour-ofbritain-2009)

e. The audience laughed Bob off the stage (Hilpert, 2014: 35; see also Goldberg, 2006)

f. This might be a hard-to-do idea, but I would like to know… (from https://forum.paradoxplaza.com/forum/index.php?threads/the-ai-and-aircraft-carriers-in-ftm.555282/)

g. An Easy-To-Find Guide of All the Best (from http://www.things-to-do-in-charleston.com/charleston-attractions.html)

h. (I know) How likely to win John is (Joshi and Kroch, 1985. Ex. 31)

In Krivochen (2012: 70, ff.), Krivochen and Kosta (2013: 115, ff.), and in passing, in fn. 208 here, we have argued that in the cases of the Path-of-Motion constructions (a-e) and complex adjectival predication (f-h), we are in the presence of a nonterminal derived in a workspace, call it W_1, that gets inserted in the place of a terminal node in a structure derived in a separate workspace, call it W_2[219]. The whole process is not very different from what a Generalized Transformation is supposed to be, in the words of Chomsky (1995: 189):

Generalized Transformation:

i. Target a category α
ii. Add an external Ø to α, thus yielding {Ø, α}
iii. Select category β
iv. Substitute Ø by β, creating {γ, {α, β}}

The internal complexity of α and β is arbitrary, such that either can be a *p*-marker itself. If so, that is, if either α or β (or both) are phrase markers (thus, nonterminals), then we are basically taking 3-manifolds at most (because of reasons we discussed in **Section 3.9** and the *Intermezzo* to **Part III**) and performing a dimensionality reduction so that they become a *terminal* for all subsequent structural intents and purposes (as such, it cannot be probed into, a property that will become relevant below). The resulting manifolds will, by virtue of being derived in a vector space, express the core vector components of the original high-dimensional structures, as a single manifold or as more than one, depending on whether dimensionality reduction was triggered by the intersection of different manifolds (as described in Saddy, 2018) or by a single manifold reaching the critical dimensionality value via locally cumulative structure building (the option we have mainly described here). We then take this lower-dimensionality version and adjoin it to another structure derived in parallel, call it β, thus increasing the dimensionality of

[219] This approach is similar to what we can find in Construction Grammar, see e.g. Goldberg (2006) or Hilpert (2014). CG assumes that the Path of Motion ('Caused Motion', in their terms) is a 'construction', a form-meaning pair lexically stored as a unit. We focus on the fact that these constructions are *cycles* that cannot be tampered with (i.e., targeted by transformations of any kind), which is compatible with the CG approach. Our theory, however, is strongly derivational, and since we operate with parallel workspaces and derive cyclicity from the oscillatory nature of the syntactic engine (rather than from properties of lexically stored units), our approach is more dynamic and flexible than that of CG. In terms of descriptive adequacy, they should be equally powerful.

the overall structure now containing β as a proper subpart. The operations applying *inside* α and β (i.e., trigger and target are both contained in the same derivational unit) are usually referred to as 'singulary' transformations, the operation that puts α and β together is a 'generalized' transformation (see Fillmore, 1963). The combination of both defines a *mixed computational system*: singulary transformations over FS sub-units are CF; generalized transformations, if implemented through a TAG with links -as we have seen already- take us up half a notch in the CH, to *mild context sensitivity*. This whole mechanism implies that there are two possible fates for a cycle which has reached the critical dimensionality value that triggers the center manifold:

188)
  a) It gets Spelled-Out –thus abandoning the syntactic workspace altogether-
  b) It gets realized as an atomized unit within a wider phrasal context –thus remaining within the syntactic workspace and being subjected *as a whole* to further syntactic operations-

Both options are freely available within local derivational horizons, and correspond to a monotonic and an oscillatory computational engine respectively, if there is nothing left in the workspace after the cycle has been Spelled-Out: since we have claimed computation need not be uniform, it would be foolish of us to restrict the possibilities for a 3-manifold to either (188 a) or (188 b) exclusively and uniformly. In fact, (188 b) can be more generally phrased as '*a lower-dimensionality version of the cycle serves as* (part of) *the input for a set of processes P at a parallel or subsequent system*'[220], which is basically Saddy's idea of normal operations over lower-dimensional condensed manifolds in a metric space. Importantly, it seems that Spell-Out serves, in this context, as a halting operation, for what gets Spelled-Out cannot be operated on anymore as it is no longer accessible from the syntactic workspace (see also Uriagereka, 2002b, 2012 for extensive discussion of this idea). This wider formulation seems to be relevant for the study of syntactic manifolds in relation to neural networks as irreducible graphs, which connect with other such graphs via paths of a lower dimensionality than those that make up the network itself (see Saddy and Grindrod, 2016; Grindrod et al., 2014; Grindrod, 2015 for a presentation of the problem). In this sense, and as a consequence of the mechanisms presented thus far, 'islands' (local syntactic domains) are, *qua* phrase markers, irreducible subgraphs within a derivation as seen from a topological perspective (it is useful at this point to recall our definition of a cycle as a maximally generable unit: this is just another way –albeit more technical- to

[220] A possible entailment of the present formulation is that any terminal in a language implemented in a physical system of the kind we have been describing here can be considered to be an atomized higher-level structure. While trivial for formal purposes (because basic and constructional meanings are not referential), the proposal has far-reaching consequences for natural languages, where correference is a major issue (in fact, it is the main focus of Binding Theory). One of the main consequences of this proposal in the most radical interpretation is that there is no *primitive* distinction between terminals and nonterminals in natural languages, they are *defined contextually* when inserted in a wider syntactic structure: any referential 'terminal' in a workspace W_x can be thought of as the atomization of an arbitrarily complex 'nonterminal' coming from W_y for $x \neq y$, or –equivalently-, as being defined by the same *ket* as an arbitrarily complex nonterminal within the same or other workspace (as seems to be the case of dislocation examples, as in (153 b-c), with the resumptive pronoun being an atomized version of the 'dislocated' structure). If this approach was somewhat correct, it would situate natural language grammars between L-grammars (derivationally) and normal-grammars (representationally), which is a most welcomed result under the present assumptions. The structural disambiguation required to yield compositionality needs access to the semantics of local domains. The theoretical and empirical implications of this reasoning are currently under research (see, e.g., Saddy, 2018; Krivochen & Saddy, 2017).

present the same concept). Part of the problem we are faced with when proposing a system in which the dimensionality and (computational/topological) complexity of an object gets dynamically reduced (e.g., to that of a simple Markov chain) is how and when to achieve this dimensionality reduction (i.e., when the CMT comes into play); but another, equally crucial part is how to avoid creating a highly entropic system in which information gets lost without control. Say, for instance, that we have irreducible graph networks[221] A and B in a neural network, which is the focus of Saddy and Grindrod's work (see also Grindrod and Lee, 2017). In such graphs each node is connected to every other node of the network (such that, in the case of a neural network, the weight of the excitatory / inhibitory connection at each point cannot be tracked to a single other node, but represents the result of nonlinear interactions between stimuli from the other nodes), and the network is itself connected to other graph(s). This interaction, however, requires that whatever computation is carried out within a network be 'flattened', possibly to finite-state level (or even instantiated in non-normal languages), as the nonlinear processes taking place within the network must be communicated to another network in a simple way, exploiting a tradeoff between dimensionality reduction and entropy. We want the interaction to take place in low dimensionality in order to make it computationally less costly, but we don't want that dimensionality reduction (which takes place by virtue of Center Manifold dynamics) to make us lose core properties of the representations being manipulated. The specifics of this process, and the structure and properties of neural networks *qua* irreducible graphs are explored in current research (e.g., Grindrod et al., 2014). Here, we want to make a point of the fact that the kind of oscillatory computation we argue for is *not* a stipulative and ultimately arbitrary process; or one that captures the putative behavior of a subsystem with utter loss of generality. Quite on the contrary, oscillatory patterns are ubiquitous in cognition and neural dynamics (not to mention more basic physical systems, including electric and magnetic fields, very much present at the core of neurocognitive research in the form, e.g., of imaging techniques).

The crucial point we want the reader to bear in mind is that *the output of a computational process* (which is, in the terms we have worked with in this thesis, a manifold) *can be topologically manipulated to yield non-homeomorphic structures* in dimensionality terms. If the argument we made in **Sections 3.9** and **3.10** is correct, then any cognitive manifold will have an upper dimensionality boundary of 3: recall *we do not predict that every single manifold will necessarily be a 3-manifold*, we do say that 3 is the critical dimensionality value that spans the center manifold[222]. Without loss of generality, the considerations we made in

[221] As a reminder,

> *A connected graph on three or more vertices is irreducible if it has no leaves, and if each vertex has a unique neighbor* [sic] *set. A connected graph on one or two vertices is also said to be irreducible, and a disconnected graph is irreducible if each of its connected components is irreducible* (Koyama et al., 2007: 35)

[222] Avery Andrews (p.c.) has pointed out that '*Current generative grammar clearly does not have a good theory about the interface of highly productive and not very productive*.' This comment pertains to Krivochen and Schmerling's (2016) theory of imperatives having a finite-state syntax (in turn based on earlier work by Schmerling, 1982), and the example Andrews has in mind is the following '*Somebody with a shotgun that isn't a civil war heirloom that's been in their family for five generations please shoot that dog that's making so much noise in the back yard of our horrible neighbors who I am afraid to confront at their front door because ....*'. The possibility of atomizing structure in the manner explained in the text, taking into consideration the idea that phrase markers display a computationally mixed character, not only solves the *apparent* problem posed by the rigidity of the generative system (since it is more of an intra-theoretical limitation than a real problem), but it also captures the fact that '*the imperative in (i)*

previous sections are compatible with derivational cascades being 1-, 2-, or 3-manifolds. Now, what we expand on here, is *how a manifold can grow and unfold by means of interference or intersections* (depending on whether the perspective is field-theoretic or topological) *with other n-manifolds derived in parallel*.

In Krivochen and Kosta (2013: 116) we argued that things like [NP climbed the hill out of town] must be analyzed as [NP #climbed the hill# out of town], with #climbed the hill# occupying a terminal position that is usually 'reserved' for a verb. The same case holds for [NP #broke the window# into the room], [NP #crossed the Atlantic# to Dover], and so on. That is, in these particular cases, structures of the form {V, NP} derived in a workspace W_1 *can* be realized as terminals for the purposes of further computations involving a marker being derived in a parallel workspace W_2 if required. More generally, the two options presented in (185 a-b) are available for any manifold that has reached the critical dimensionality value *or* that has exhausted the set of types it could work with. Crucially, if a structure is dimensionally 'flattened' (and computationally simplified), and then realized in a different, parallel workspace for further structure building; that means that from the local perspective of this structure, the syntactic computation proceeds by building a manifold cumulatively. The manifold reaches a critical dimensionality point, which spams the *center manifold* and the core propertoes of the manifold squeeze to a lower dimensionality. Then, this dimensionally flattened, concentrated version of the original structure (which we will refer to as an 'atomized' version, because the dimensionality reduction renders it a terminal node) becomes involved in further structure building processes in a different workspace, which in turn build increasingly complex manifolds in the form of predication structures, following (123). This processes define a back-and-forth, oscillatory dimensional dynamics that applies to interaction between networks as irreducible graphs as well as to complex derivational cascades being atomized and adjoined as terminals in further computations that expand parallel manifolds.

It is relevant at this point to clarify that this oscillatory dynamics is not exclusive to the verbal / eventive domain (i.e., VP) in natural language: the nominal / sortal domain (i.e., NP) displays such processes as well. Examples (187 e-f) are eloquent in this respect: in both cases, we have object-raising like structures, in which the "object" appears in a post-predicate position. Note that the entire complex predicate has the functional potential of an adjective, and is a constituent in and of itself:

187 g') *A to find guide

But

187 g'') An easy guide

Also, while it is possible to modify the atomized structure as a whole (e.g., [very easy to find]), it cannot be probed into (cf. *[easy to perhaps find]). This even works for monotonically

---

[Andrews' example] *is such a computationally mixed example, involving both finite-state and phrase-structural substrings. Note that the embedding clause in (i) remains an imperative: everybody in (i) cannot have scope over the subject. At the level of the imperative as a whole, the indicative substring is interpreted perfectively, computed as if it were a point with no internal structure: in truth, it is a domain that is opaque for the purposes of further operations. This is by no means incompatible with a finite state syntax for imperative and other non-embeddable structures, as the internal complexity of such embedded clauses (however this complexity is measured) is not relevant to computational modelling in terms of formal languages*' (Krivochen and Schmerling, 2016).

assembled complex structures like [very difficult to attempt to solve], in which what is [very difficult] is the *attempt*, not the *solution*. This is consistent with a cumulative model in which 'higher' (later) cycles properly contain 'lower' or previous cycles (or remnants thereof, as we saw in **Section 3.8** above), without this meaning that they have all been derived monotonically: even though the structure grows, it does *not* do so uniformly as we introduce cycles derived in parallel.

These cases (187 e-g) are analogous to classical instances of *tough-movement*, such as (189):

189) a. This problem is difficult to solve
b. This problem is difficult (cf. 184 f'')
c. *This problem is to solve (cf. 184 f')

Whatever the reader's theory about tough-movement is (for our own perspective, see Krivochen and Kosta, 2013: 112, ff.), in (187 e-g) there is a *tough*-(like)structure (such that [easy to find] ~ [difficult to solve]) that gets *atomized* after reaching critical dimensionality or exhausting the resources in a certain workspace -a process that is available for standard *tough*-movement as well, such that |difficult to solve| becomes #difficult to solve#- and instantiated as a lower dimensional manifold in a parallel workspace, where it is used for further computations. Thus, and as predicted by the theory, while (189') below is not as usual a structure as (189 a), it *is* attested (*circa* 23,000 Google results for the exact phrase "a difficult to solve problem" as of 15/12/2015, including academic papers):

189') […] When faced with **a difficult-to-solve problem**, it is standard practice in the optimization community […] (from Andrew Blake, Pushmeet Kohli, and Carsten Rother (2011) *Markov Random Fields for Vision and Image Processing*. Cambridge, Mass.: MIT Press, p. 249)

The notion of atomization must not be overgeneralized, though. In this respect, Susan Schmerling (p.c.) correctly points out that '*not all 'tough-' structures have corresponding 'atomized' versions*'. Relevant examples are of the form:

190) This problem$_i$ was hard to convince John to expect Mary to persuade him to work on t_i

Schmerling correctly points out that '*Such atomization is impossible here*'. This poses an interesting conundrum, because in principle it is not obvious why this should be the case. In other words, it is not clear why we would not be able to atomize such that we can get things like

191) **A [hard to convince John to expect Mary to persuade him to work on] problem*

but, as seen above, we can indeed have *an easy to solve problem*.

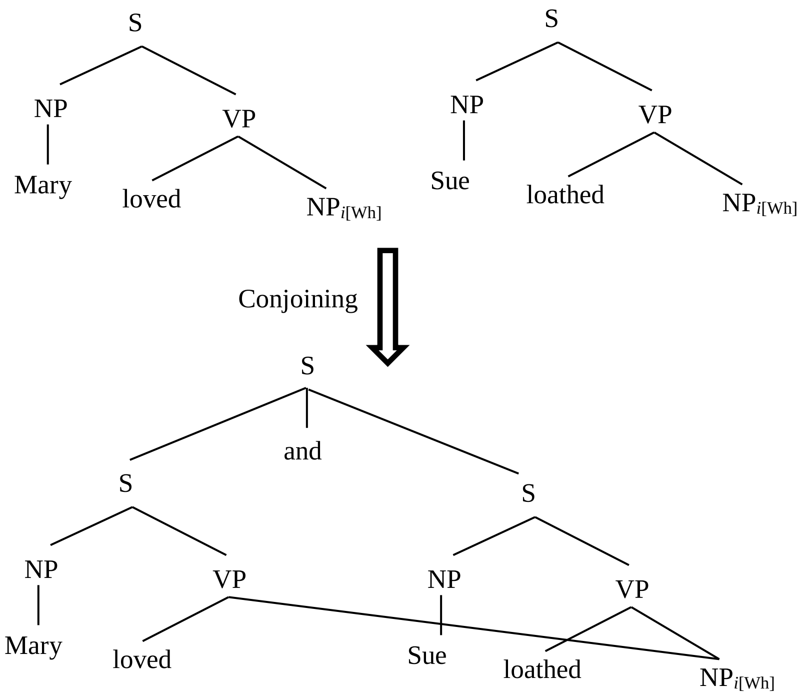
The reasons for this are not entirely clear, although it seems to be related to the semantic nature of syntactic computation. Note that in order to interpret (190) we need to have access to the monotonically recursive structure of the embedded ECM clauses, and after atomization that access is *not* possible. We need to probe into the sequence of embedded clauses in order to have an adequate semantic representation, which requires the layered structure (or, more specifically, the distinctness of clausal domains) to be preserved. Thus it is not possible to build a semantic representation for (191) if the sequence of embedded clauses is atomized. Note that this problem does not arise in the simple case in which a single eventive domain is atomized, as in *hard to find* or indeed *climb the hill*.

Our theory predicts that, if adjoined domains are atomized units, whose internal structure has been flattened through *center manifold* dynamics, opacity effects (i.e., blocking of reordering transformation) should appear. Recall our adoption of the distinction between *embedding* and *conjoining* transformations: Fillmore distinguishes between two kinds of *generalized* transformations: (a) *embedding transformations*, which insert a sequence into another thus generating *hypotactic dependencies*, and (b) *conjoining transformations*, which take A and B and form C containing A and B, generating a *paratactic* dependency between them. For each symbol in a sequence, an *embedding transformation* specifies the structure of the *pre-sentence* that can be inserted in the structural position occupied by that symbol, as long as contextual conditions (specified by means of variables in the transformation) are met. A system that allows for both conjoining and embedding instantiates the kind of *mixed dependencies* we advocate for without the need to invoke further mechanisms: the general formulation of *embedding* transformations is Context-Sensitive, whereas the format of *conjoining* transformations is Context-Free. In a sentence in which both have applied, we will have local units of different computational complexity, for example:

192) The fate of the man who Mary loved and Sue loathed was unknown

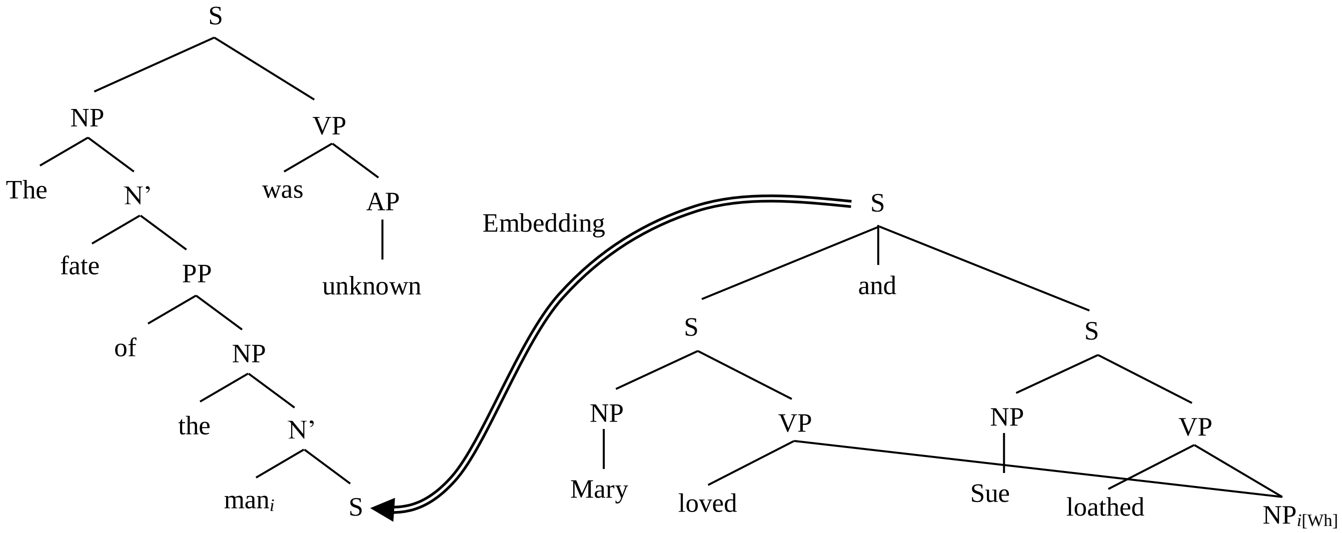
Relative clauses are inserted via embedding, and symmetric coordination (Schmerling, 1975) is a good example of conjoining. The relevant derivation, thus, would go along the following lines (we omit some labelled nodes for simplicity):

193)



The *conjoining* transformation generates a paratactic dependency between the two Ss, which materializes as a conjunction *and*. After conjoining, the two instances of $NP_{[Wh]}$ are identified, by virtue of being defined by the same vector components (a.k.a., being identical addresses, in the terminology used in Karttunen & Kay, 1985): we represented this identification by means of multidominance (see de Vries, 2009 for extensive discussion about the implementation of multidominance as a way to model non-local dependencies in generative grammar). Note that in a semantically-driven system we would require some additional stipulation to ban the identification of nodes (whichever these are, NP, VP, S…) that are defined by identical vector components: a Hausdorff space with a weak separation axiom (the kind of space that emerges with syntactic computation) maintains topological distinguishability *for distinct points,* but we are dealing with *identical points* belonging to structural units which are derived in parallel and related syntactically (i.e., brought close together, *as per* Uriagereka's insight). Now we must get to embedding. The target of Relative Clause adjunction must be an NP with a designated S node (adapted from Fillmore, 1963: 223), which will be substituted by the root S that results from *conjoining* under the condition that the adjoined S is *not* interrogative. Thus we get a structure along the lines of (194) –again, using traditional phrase labels for concreteness, note that all we really require is that the root of the embedded tree and a node in the frontier of the target tree be defined by the same components *as per* Joshi, 1985-:

194)



The derivation sketched in (194), then, involves *both* parataxis and hypotaxis, combining different kinds of computational dependencies (we saw a similar case in (112), above). Note that (194) could *not* have been derived monotonically, for we are relating *separate preliminary strings* (to each of which corresponds an elementary tree) which display dynamically varying computational dependencies (Type 2 for each separate unit *Mary loved NP* and *Sue loathed NP*, Type 3 once these units are symmetrically coordinated, and Type 2 again when the complex phrase marker is flattened and inserted in a node at the frontier of the target IT); the structure does grow all the way, but *not always at the same rate*.

The possibility of applying singulary transformations *after* embedding transformations is particularly interesting for purposes of the mixed computational model we propose here, since *adjunction* and *substitution* (in the TAG sense) are precisely *embedding transformations*. Therefore, we can formulate hypothesis about the relative ordering of singulary and generalized transformations, within and across derivational currents. In this context, we propose the following trans-derivational locality condition (which we anticipated in the physical discussion of cycles *qua* manifolds):

Let γ and β be two sub-trees such that γ contains a node X that corresponds to the root of β. A singulary transformation T_S triggered from γ can affect β ***iff*** *T_S is intrinsically ordered* ***after*** *an embedding transformation that adjoins β to γ at X.*

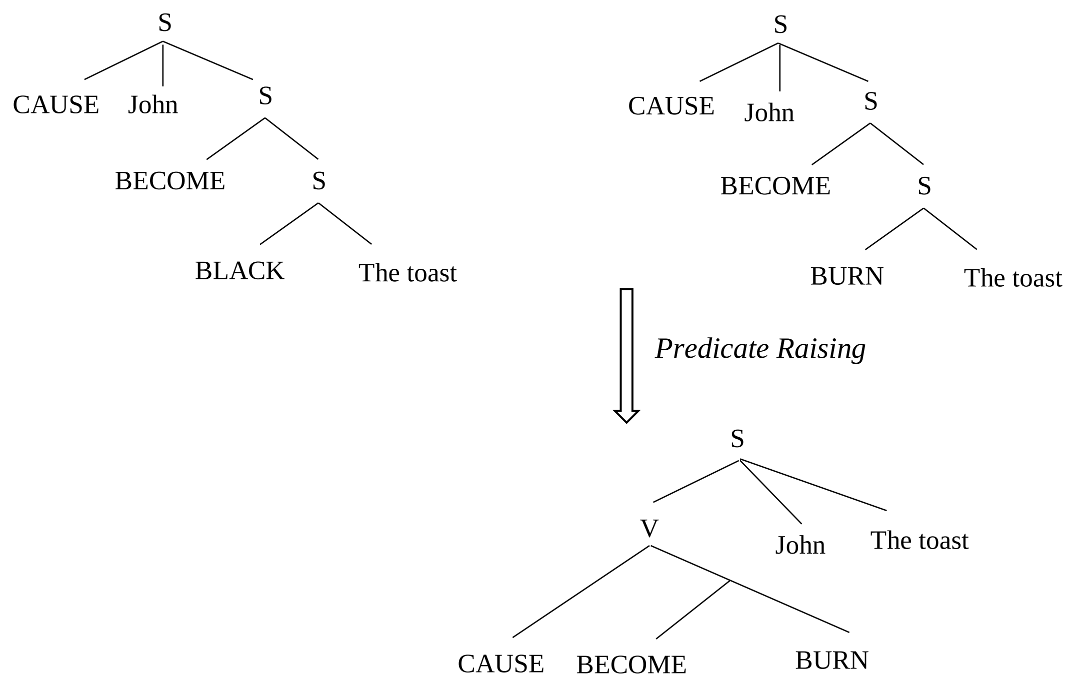
What singulary transformations *cannot have access to*, we argue, is *elements embedded within β*; only *β* as a whole can be affected by a singulary transformation at γ ordered after *adjunction* of β to γ. Now consider our proposal about the derivation of *resultative* constructions and complex attributive expressions: in a resultative construction, primary and secondary predication do not coexist in a derivational space *before* adjunction, and after adjunction the Initial Tree is opaque to operations triggered at Auxiliary Tree (the target of adjunction), including of course *Wh-movement* (a singulary transformation). Consider, for instance:

195) a. John burnt the toast black (transitive resultative)
b. The river froze solid (ergative resultative)
c. Mary shouted herself hoarse (unergative resultative with post-verbal NP)

Lexical-syntactic decomposition approaches (e.g., Mateu Fontanals, 2002; Acedo Matellán, 2010) assume the presence of more than a single syntactic object in the derivation of resultatives like (195 a-c). That perspective owes much to the lexical-semantic perspective first proposed in Lakoff (1965), and more in detail McCawley (1968, 1971) –see also Dowty, 1979-, but pushes it forward with the addition of trans-derivational relations, which can in fact be subsumed to Fillmore's two kinds of generalized transformations. Let us give an example, using (195 a) as our case in point. The following diagram for a resultative construction assumes McCawley's (1968, 1973) 'pre-lexical syntax' proposal and uses the terminology in Joshi (1985) for each sub-derivation; we also show *predicate raising* applying inside the IT (we need to note that more recent proposals about lexical derivations are either notational variants of *predicate raising* or have the same empirical coverage, and we see no reason to fix what ain't broken):

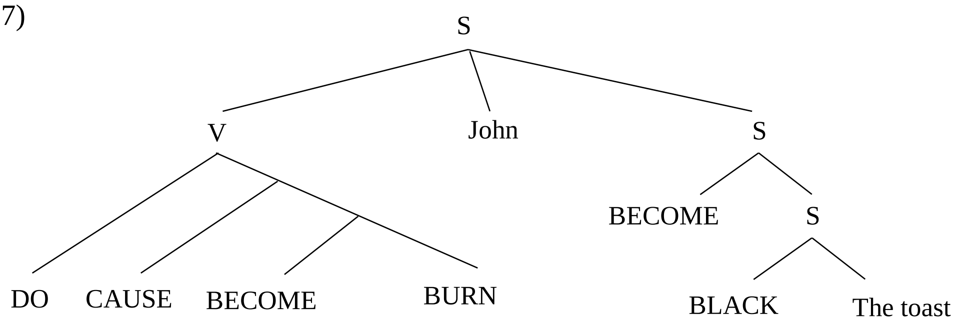
196) John burnt the toast black

Initial Tree (AT) *Auxiliary Tree* (IT)



The derivation in (196) is weakly equivalent to an event composition approach, insofar as each sub-event has its associated *elementary tree*, and we can relate those sub-derivations via *adjunction*. However, unlike Neo-Davidsonian-based approaches (e.g., Ramchand, 2008; Pylkkänen, 2008; Rothstein, 2004), our system is *not* monotonic, requiring a system to relate separate derivational units. The trans-derivational extension that we propose here for semantic-syntactic structures (which crucially makes use of both context-free and context-sensitive operations) allows us to manipulate two parallel clausal domains (the AT and the IT); semantically, this corresponds to *two events with their corresponding arguments*. These domains each have a node with an identical categorial specification, which licenses the application of *adjunction*. Concretely, we apply *predicate raising* in AT, and the resulting V node is adjoined to V in IT, thus generating (197):

197)



Now, by the ordering constraint formulated above, a singulary transformation like Wh-movement would not be able to target a non-root element from the adjoined domain. Thus,

198) *What$_i$ did John burn the toast *e$_i$*?

is adequately excluded, since the extraction site (marked with *e*) is inside an *adjunction-induced island*, a domain that was atomized (computationally 'flattened', in the words of Lasnik & Uriagereka) and inserted in a wider phrasal context. Indeed, reordering transformations (like Wh-movement and Topicalization) *cannot* apply to resultatives targeting either the result or the affected object respectively:

198') a. *What did the river freeze? (answer: 'solid')
b. *What did Mary shout herself? (answer: 'hoarse')
c. */%…that / because, the metal, John hammered *t* flat
d. */%…that / because, the river, *t* froze solid
e. *…that / because, herself, Mary shouted *t* hoarse

On the other hand, if we are dealing with a monotonic structure, extraction should be permitted since we are working within a single derivational space, and structure is preserved in all its complexity. Let us now see the case of garden-variety depictive secondary predication:

199) John drinks his tea hot

There is no reason to propose a multiple derivation structure for (199), thus, (200 a-b) are correctly predicted to be grammatical

200) a. What does John drink *t* hot?
b. …because his tea, John drinks *t* hot

Consider also Complex NP Constraint violations like (201):

201) *Who$_i$ does the fate of the man that Mary loved t_i and Sue loathed t_i is unknown

If the derivation of coordinate structures and relative clause adjunction proceeds along the lines suggested here and in the references mentioned above, then the ungrammaticality of (201), and indeed CNPC violations without loss of generality follows from the fact that CNP always involve more than a single derivational space (one for the target NP, one for the adjoined RC); the target of extraction cannot be affected by a singulary transformation triggered at the IT.

Changing the perspective on syntactic computation from uniform to locally variable allows us to capture well-established syntactic conditions as properties of the shift in computational complexity and the opacity of adjoined domains. In a system that allows for mixed computation, and in which the relation between derivational units is mediated by structural flattening (via *center manifold dynamics*), the impossibility of extracting material from a relative clause emerges from the dynamics of the computation: relative clauses are non-monotonically introduced in the derivation via substitution, and a transformation triggered at the target of substitution cannot have access to the internal structure of the adjoined domain, which was built separately.

We would like to finish this section by presenting an example of the interaction between atomization-adjunction and a different kind of process: *inflection*, rather than *extraction*[223]. We will see that the realization of a complex derivational unit as a terminal in a wider syntactic context

It has been known for some time that certain constructions that look as if they include variants of infinitival complements of verbs that are introduced by *and* instead of *to* are ungrammatical just in case the matrix verb is audibly inflected (we have analyzed these cases of *mirage coordination* in Krivochen & Schmerling, 2017, which must be distinguished from *true* coordinations in both their *et*- and *–que* varieties, which we have analyzed here in **Section 4.4** and in Krivochen & Schmerling, 2016):

202) a. I/you/we/they (will) always try and check the battery fluid.
b. *She always tries and checks the battery fluid.

The present section focuses on another construction that is likewise not inflectable:

203) a. Put your arm out nice and straight.
b. *Put your arm our nicer and straight.
c. *Put your arm out nice and straighter.
d. *Put your arm out nicer and straighter.

These apparently conjoined adjectives show that the lack of inflectability in (202) is part of a phenomenon that is more general than one limited to *apparently* conjoined verbs. Because of the poorly known character of paradigms like (203), it is our focus here. The paradigm in (204) is analogous:

204) a. Make a fist for me, good and hard.
b. *Make a fist for me, better and hard.

[223] The examples and analyses in this sub-section build on an LSA abstract co-authored with Susan Schmerling, who has very generously allowed me to use that material in this work. The usual disclaimers apply.

c. ?*Make a fist for me, good and harder.
d. *Make a fist for me, better and harder. (* as a comparative analogue of (201a))

The examples in (203) and (204) have the appearance of conjoined adjectives, but the impossibility of inflecting them indicates that we are dealing here with something else. We suggest that *nice and* in (203) and *good and* in (204) do not involve true adjectives (note the absence of the usual meanings for *nice* and *good*, respectively) but are *modifiers* of the adjectives that follow them. Moreover, this modification is *not* monotonically recursive, for *The soup is nice and very hot* can only be interpreted as a *true* coordination. The semantic representation of the relevant portion of (203 a) in propositional terms for the reading we are interested in here is *not* the compositional

203') nice(arm) & straight(arm)

But rather

203") straight(arm)

By the same token, (203) and (204) do not involve true coordination, despite the presence of the word *and*. An example like (205 a) is ambiguous between *true* coordination and a construction like that in (203 a); inflecting one of the adjectives in (205) yields unambiguous *true* coordination:

205) a. The soup is nice and hot → a. nice(soup) & hot(soup)
b. hot(soup)
b. ?The soup is nicer and hot.
c. ?The soup is nice and hotter.
d. The soup is nicer and hotter → a. nice(soup) & hot(soup)
b. *hot(soup)

Let us see how an oscillatory computational system can help us describe this phenomenon adequately. There are some points to note:

First, it cannot be the case that we are dealing with a monotonic structure. That is, a structural description along the lines of (206) is insufficient on at least two grounds:

206) [$_S$ [the soup] [$_{VP}$ is [nice [and hot]]]]

Syntactically, there is nothing in (206) that could block the inflection of either of the adjectives, meaning we should be able to get the strings in (205 b, c) freely (thus, we *overgenerate*). Note that this problem cannot be circumvented by saying that we are dealing with *que-coordinated* As within a single derivational space (which would yield the local structural description [[nice] and [hot]]) because the problem is not –now- the level of computational complexity that the phrase marker presents, but the fact that a rule –inflection- simply cannot be allowed to apply. Semantically, given a structural description like (206) or the *que-coordinated* variant, only the *true* coordination reading would be available.

What can we do, then? Well, we can segment (205 a), in the reading that interests us, in two derivational units:

207) a. [$_S$ The soup is [X]]
b. [$_X$ nice and hot]

We *must* construct (207 b) separately from (207 a), because it is clear that *nice and A* does *not* behave like a usual adjective. For example, it cannot appear in pre-nominal position:

208) *A nice and hot soup (grammatical in the compositional reading)

Moreover, unlike sequences of true adjectives, *nice and* affixation is not (head-tail) recursive:

209) *The soup is nice and hot and garlicky (idem ant.)

Note that (209) can only be segmented as [nice and hot [and garlicky]] with prosodic assistance, pretty much like [fish and chips [and a smoothie]] in (161) above. Now, having provided arguments that we are not dealing with an adjective, we can ask: is *nice and* (or *good and*, etc.) an *intensifier*, like *very*? *Prima facie*, there seem to be some common features:

Both *very* and *nice and* block *comparative formation*

210) a. *The soup is very hotter
b. *The soup is nice and hotter

And both *very* and *nice and* can be fronted with their As:

211) a. Very hot is how John likes his soup
b. Nice and hot is how John likes his soup

So, is *nice and* an element of the same category as *very*? We don't think so. To begin with, *nice and hot* is not paraphrased as *very hot*, but as *adequately hot*, same with *good and straight*. *Nice and* does not share key properties of an adverbial intensifier like *very*, since we can have *this soup is very hot and garlicky* and *a very hot and garlicky soup*. Moreover, *very* can be iterated to indicate intensitiy: *very very very hot*; this is impossible with *nice and*. Prosody may come to the rescue when deciding whereas *very* has scope over *hot* only or both *hot* and *garkicly*, but that is a different kind of problem than that we face with *nice and*. Note also that *nice* does not behave in (202a) as its garden variety counterpart, for example:

212) a. The house is nice-ish and rather big
b. *The soup is nice-ish and hot

The impossibility of having the comparative form, as well as the differences between the distribution of regular APs and *nice and A* (including the fact that *nice and* cannot be iterated, nor is it recursive) suggest that we are not dealing with a syntactic phrase headed by an adjective, but rather with a lexical terminal #nice and A#, assembled separately from the rest of the derivation. This unit, we suggest, does not satisfy the structural description for *comparative formation* because it is not an A or an AP (and much less a coordination). Note that we can extract the first term of a coordination between adjectives in clefting, but this is impossible with the *nice and* construction:

213) a. Hot is how John likes his soup, and garkicly
b. *Nice is how John likes his soup, and hot

Our proposal, then, is twofold:

On the one hand, we propose that *nice and*, in the meaning that we have examined here, forms a syntactic terminal with the adjective, and as such it cannot be internally modified. Moreover, this terminal is not of category A, for it does not share distribution (a.k.a. 'functional potential') with garden-variety adjectives (and what is a part of speech if not a set of distributional constraints?). Furthermore, we propose that this terminal does not come to be in the same derivational current as the rest of the sentence in which it appears; rather, it is inserted by *substitution* at the designated node in (207 a), the precise identity of which is of no consequence for this argument.

Of course, a more general linguistic account should take into consideration how coinage processes are heavily influenced by extra-systematic factors, including language contact and cultural phenomena. However, the general principle holds: the system as a whole can go back and forth in an oscillatory pattern of changing dimensionality as manifolds are atomized to serve as inputs for further computations in parallel workspaces; in turn, we need to relax the requirement that structure should be computationally uniform in order to accommodate these oscillatory dynamics.

Epilogue: Recapitulation and conclusions[224]

In the present work we have argued for a non-uniform, prejudice-free approach to the physics of linguistic computations, and the problems that arise when an algorithmic approach to cognition is faced with the task of assigning structure to the phenomenological world. **Part I** was concerned with the definition and formal properties of grammars that generate structural descriptions for strings of symbols of any nature; and the formal automata that can generate and recognize such strings. We discussed function-based and interaction-based approaches to computations, and introduced one of the *leitmotifs* of the thesis, which is the concept of *dynamical frustration*. **Part II** further explored these issues, focusing on the interplay between theories of computation and theories of cognition, tackling the issues of the apparent incommensurability of neurocognitive and linguistic explanation, as well as other conceptual tensions that seem to appear at different levels of analysis. We suggested that, instead of trying to resolve these in favor of one or the other possibility, more interesting dynamics arise if we assume that there is a tension which is never resolved in the sense of reaching a state of equilibrium. The dynamical frustration thereby generated produces interesting emergent properties if combined with specific assumptions we enunciated about the nuts and bolts of the systems involved in linguistic computations. Among these properties, we set our focus on *locality*. After the presentation of both the framework and the problem, **Part III** formalized some aspects of our proposal about the connections between computation and physical systems within a physical-mathematical framework, that of quantum field theory. After outlining our assumptions, and deriving some equations, we were ready to tackle structure building and structure mapping as topological operations over a metrizable ultrametric space, and to focus on *island* phenomena as natural emergent properties of a physically realized formal system, which we characterized mathematically. This system is instantiated in the physical world in the form of interfering continuous fields which oscillate in space and time, under conditions we also made explicit. **Part IV** was devoted to the analysis of island conditions (using mainly English examples) under field-theoretic assumptions, as conditions over possible interference patterns over fields oscillating in both time and space. Constraints on variables in syntax-semantics were analyzed both physically and topologically.

[224] Or: *Physics of Language: a New Hope.*

Like Jess Tauber said, wisely, ‘It’s just music, dude’.

For the sake of closure and completeness, these are the main points of the thesis:

1. A distinction between natural and formal languages must be drawn, for only the latter obey function-based computation uniformly. Natural languages, as cognitive systems, are better described by means of interactive computation. There is a long history of careful grammatical description of natural languages some of whose results defy traditional, function-based approaches to cognitive computation. We are *not* throwing Turing away, we are simply limiting the applicability of CTT to the kind of phenomena that Turing originally considered (number theoretic functions over naturals).

2. Natural language strings combine aspects of normal and non-normal grammars, which hold for different sub-strings. In this respect, the proposal put forth here and in past works (mainly, Krivochen, 2015a, 2016d; also Krivochen and Schmerling, 2016; Bravo et al., 2015; García Fernández et al., 2017) presents a serious empirical and theoretical challenge to the ‘principle of invariance of linguistic complexity’, which Sampson (2009: 1) calls a strong deep uniformity assumption.

3. Natural language strings are not computationally uniform. Thus, at most we can say that a set of weakly generated strings or sub-strings is characterizable by a specific formal grammar in the Chomsky Hierarchy (FS, CF, CS). Empirically, we have seen that claiming that all strings that belong to a natural language L are generated by the same engine and display uniform computational dependencies yields a *putatively* simple, but really procrustean approach to natural language study. In this context, the proposal of ‘mixed computation’ aims at adequately constraining the configurations in which interpretations arise by establishing dependencies between syntactic objects in separate derivational units.

4. A computationally mixed grammar does not *need* to be a parser. Mixed computation emerges from meaning being a ‘generative’ component, and restricting the class of adequate structural descriptions for NL substrings. Thus, mixed computation is *not* an ‘interface’ phenomenon in the traditional MGG sense: there is no autonomous syntax that sound and meaning systems ‘interface’ with. Sound and meaning systems (of which we focus on the latter) *are* syntactic. The bases for mixed computation, which are essentially internal to the dynamics of the formal system, must thus be distinguished from the bases for *cyclicity*, which only partially emerge from the formal system: a *physically implemented* formal system displaying *dynamical frustration* will necessarily –we argue- display cyclic behaviour.

5. A dynamical field-theoretic approach to cognition, at the levels of computation and physiology, not only reveals possible motivations for ubiquitous properties of cognition (like locality and chunking), but also integrates cognition with other –better understood and more completely modelled- physical systems. Such integration must be based on explicit claims about *scale* and the way in which different processes at different scales can emerge from the same underlying (oscillatory) process.

6. In connection with the last point, we have claimed that one of the most important properties of cognition is that it displays tensions between mutually incompatible tendencies at different levels (*scale frustration*): such a tension is called a *dynamical frustration*, and it is this frustration that derives syntactic cycles and atomized domains in coordinated structures, chains of auxiliary verbs, resultative constructions and path-of-motion, *inter alia*. These were the empirical domains on which our syntactic-semantic proposals were tested.

7. Bearing in mind questions of scale, it is possible and even necessary to take advantage of developments in physics and mathematics (e.g., quantum field theory, electrodynamics, topology) in order to characterize aspects of cognition that resist characterization in classical terms (where 'classical' is to be understood, roughly, as 'the mind is a digital computer'). On the approach presented in this thesis, 'syntax' is just a (non-generative, as there is no recursive enumeration) topological operation over metric and non-metric fields 'all throughout the mind', not a set of purely formal operations specific to language with little if any connection to neurocognition. In the present perspective, operations do *not act on objects, they affect the layout of the space where objects are defined* (this process being the metrization of the space which yields the intersection and 'condensation' of manifolds).

More generally, however, the take-home message we would like the reader to bear in mind if he made it through all these pages, is that *no object in the physical world should or can be studied in substantive and methodological isolation*, since there are crucial aspects that only come to light when we change perspectives, shift our focus, try to look at something familiar as if it were completely novel (a process that Viktor Shklovski referred to, in art, as *остранение* 'ostranenie', 'defamiliarization'). To no small extent, a multidisciplinary approach allows us to defamiliarize what in discipline X is taken for granted, using insights drawn from discipline Y (and Z, and…), respecting the approaches of both X and Y, without forcing concepts or blindly re-naming old objects to follow current fashions. If we have persuaded the reader at least of the usefulness and even the necessity of the physical approach to language and cognition (and language *in* cognition), this thesis will have not been in vain.

References

Abbott, Edwin A. (1991) *Flatland: A romance of many dimensions*. Princeton: Princeton University Press.

Abels, Klaus and Kristine Bentzen (2009) A note on the punctuated nature of movement paths. *Catalan Journal of Linguistics* 8. 19-40.

Abney, Steven Paul (1987) *The English Noun Phrase in its Sentential Aspect.* PhD Thesis, MIT.

Acedo-Matellán, Víctor (2010) *Argument Structure and the Syntax-Morphology Interface. A Case Study in Latin and other Languages*. Ph.D. Thesis, December 2010, Universitat de Barcelona.

Acquaviva, Paolo and Phoevos Panagiotidis (2012) Lexical decomposition meets conceptual atomism. *Lingue e linguaggio* XI(2). 165-180.

Adger, David (2012) *A Syntax of Substance*. Cambridge, Mass.: MIT Press.

(2015) Mythical myths: Comments on Vyvyan Evans' "The Language Myth". *Lingua* 158. 76 - 80.

Afalo, Tiffany and Michael Graziano (2008) Four-Dimensional Spatial Reasoning in Humans. *Journal of Experimental Psychology* 34(5). 1066-1077.

Aguayo, José, Juan M. Saavedra, and Matt Wallace (2004) Attractor and Repeller Points for a Several-Variable Analytic Dynamical System in a Non-Archimedean Setting. *Theoretical and Mathematical Physics* 140(2). 1175-1181.

Ajdukiewicz, Kazimierz (1935) Die Syntaktische Konnexität. *Studia Philosophica*, 1. 1-27.

Altmann, Gerry (1998) *The Ascent of Babel*. Oxford: OUP.

Amari, Shun-ichi (1977). Dynamics of pattern formation in lateral-inhibition type neural fields. *Biological Cybernetics*, 27(2). 77–87.

Ambinder, Michael, Ranxiao Frances Wang, James Crowell, George Francis, and Peter Brinkmann (2009) Human four-dimensional spatial intuition in virtual reality. *Psychonomic Bulletin & Review*, 16(5). 818-823.

Anderson, Stephen (1969) *West Scandinavian Vowel Systems*. PhD Dissertation, MIT.

Aoun, Joseph and Dominique Sportiche (1983) On the Formal Theory of Government. *The Linguistic Review* 2. 211–236.

Apostol, Tom M. (1967) *Calculus, Vol. 1: One-Variable Calculus with an Introduction to Linear Algebra* (2nd ed.). New York: John Wiley & Sons.

Arbib, Michael (1987) *Brains, Machines, and Mathematics*. New York: Springer.

Ashby, W. Ross (1947) Principles of the self-organizing dynamic system. *Journal of General Psychology* 37. 125–128.

(1962) Principles of the self-organizing system. In H. Von Foerster and G. W. Zopf, Jr. (eds.) *Principles of Self-Organization: Transactions of the University of Illinois Symposium*. London: Pergamon Press. 255-278.

Attneave, Fred (1959) *Applications of information theory to psychology: a summary of basic concepts, methods, and results*. New York: Holt, Reinhart & Winston.

(1981) Three approaches to perceptual organization. In Michael Kubovy & James R. Pomerantz (eds.) *Perceptual Organization*. London: Routledge.

Austin, John L. (1962) *How to do Things with Words: The William James Lectures delivered at Harvard University in 1955* (eds. James O. Urmson and Marina Sbisà). Oxford: Clarendon Press.

Austin, Peter and Joan Bresnan (1996) Non-configurationality in Australian aboriginal languages. *Natural Language and Linguistic Theory* 14(2). 215-268.

Bach, Emmon (1970) Problominalization. *Linguistic Inquiry*, 1. 121 - 122.

(1979) Montague Grammar and Classical Transformational Grammar. In Steven Davies & Marianne Mithun (eds.), *Linguistics, Philosophy, and Montague Grammar*. Austin: University of Texas Press.

Baddeley, Alan (2003) Working Memory and Language: An Overview. *Journal of Communication Disorders* 36 (2003). 189–208.

(2007) *Working Memory: Thought and Action*. Oxford: OUP.

Balari, Sergio, Antonio Benítez-Burraco, Marta Camps, Víctor M. Longa, Guillermo Lorenzo & Juan Uriagereka (2011) The archaeological record speaks: Bridging anthropology and linguistics. *International Journal of Evolutionary Biology*. doi:10.4061/2011/382679.

Baranger, Michael (2004) Chaos, Complexity and Entropy. Ms. Cambridge, Mass. MIT.

Bates, Andrew and Anthony Maxwell (2005) *DNA Topology*. Oxford: OUP.

Behme, Christina (2014) *Evaluating Cartesian Linguistics: From Historical Antecedents to Computational Modelling*. Frankfurt am Main: Peter Lang.

(2015a) Is the ontology of biolinguistics coherent? *Language Sciences* 47. 32–42.

(2015b) Chomskyan Ambiguity. Ms. Invited contribution to *Language and Cognition*.

Beim Graben, Peter, Bryan Jurish, Douglas Saddy, and Stefan Frisch (2004) Language Processing by Dynamical Systems. *International Journal of Bifurcation and Chaos* 14(2). 599-621.

Beim Graben, Peter, Sabrina Gerth, Douglas Saddy and Roland Potthast (2007) Fock space representation in neural field theories. In N. Biggs et al. (Eds.), *Proceedings of Waves 2007. The 8th International Conference on Mathematical and Numerical Aspects of Waves*. Department of Mathematics, University of Reading. 120 – 122.

Beim Graben, Peter, Dimitris Pinotsis, Douglas Saddy, and Roland Potthast (2008) Language processing with dynamic fields. *Cognitive Neurodynamics* 2(2). 79-88.

Bejan, Adrian and James Marden (2006) Unifying constructal theory for scale effects in running, swimming and flying. *Journal of Experimental Biology* 209. 238-248.

Bernecker, Sven (2016) Extended minds in vats. In Sanford Goldberg (ed.) *The Brain in a Vat.* Cambridge: CUP. 54-72.

Berwick, Robert (1984) Strong Generative Capacity, Weak Generative Capacity, and Modern Linguistic Theories. *Computational Linguistics* 10(3-4). 189-202.

Bever, Thomas (1970) The cognitive basis for linguistic structures. In Robert Hayes (ed.) *Cognition and Language Development*. New York: Wiley & Sons. 279–362.

(2013) The Biolinguistics of Language Universals. In Sanz, Montserrat, Itziar Laka & Michael Tanenhaus (eds.) *Language Down the Garden Path*. Oxford: OUP. 385-404.

Bickerton, Derek (2007) Language evolution: A brief guide for linguists. *Lingua* 117. 510–526.

Bierwisch, Manfred (1967) Some semantic universals of German adjectivals. *Foundations of Language* 3. 1-36.

Binder, Philipe (2008) Frustration in Complexity. *Science* 320. 322-323.

Blake, Randolph, and Robert Sekuler (2006) *Perception*. 5th ed. New York, NY: McGraw-Hill.

Blank, Martin and Reba Goodman (2011) DNA is a fractal antenna in electromagnetic fields. *International Journal of Radiation Biology*, 87(4). 409-415.

Blevins, Jim (1990) *Syntactic Complexity*. University of Massachusetts, Amherst.

Bloomfield, Leonard (1926) A Set of Postulates for the Science of Language. *Language* 2(3). 153-164.

Blutner, Reinhard (2000) Some Aspects of Optimality in Natural Language Interpretation. *Journal of Semantics*, 17(3). 189–216.

Blutner, Reinhard & Anatoli Strigin (2011) Bidirectional grammar and bidirectional optimization. In Anton Benz & Jason Mattausch (eds.) *Bidirectional Optimality Theory*. Amsterdam: John Benjamins. 221-248.

Boccara, Nino (2002) *Modeling Complex Systems*. Springer. [2nd edition: 2010]

Boden, Margaret (2006) *Mind as Machine: A History of Cognitive Science*. 2 Vols. Oxford: Clarendon Press.

Boeckx, Cedric (2008) *Aspects of the Syntax of Agreement*. London: Routledge.

(2010) Defeating Lexicocentrism. Ms. ICREA / UAB. http://ling.auf.net/lingbuzz/001130 [Partially reprinted in Boeckx, 2014: Chapter 1]

(2014) *Elementary Syntactic Structures: Prospects of a Feature-Free Syntax*. Cambridge: Cambridge University Press.

Boothby, William (1975) *An Introduction to Differentiable Manifolds and Riemannian Geometry*. New York: Academic Press.

Borer, Hagit (2005a) *In Name Only. Structuring Sense, Volume I*. Oxford: Oxford University Press.

(2005b) *The Normal Course of Events. Structuring Sense, Volume II*. Oxford: Oxford University Press.

(2014) The category of roots. In Artemis Alexiadou, Hagit Borer, and Florian Schäfer (eds.) *The Syntax of Roots and the Roots of Syntax*. Oxford: OUP. 112-148.

Bourgine, Paul and John Stewart (2004) Autopoiesis and Cognition. *Artificial Life* 10(3). 327-345.

Bowers, Jeffrey (2017) Parallel Distributed Processing Theory in the Age of Deep Networks. *Trends in Cognitive Sciences* 21(12). 950-961. http://dx.doi.org/10.1016/j.tics.2017.09.013

Bravo, Ana, Luis García Fernández, and Diego Krivochen (2015) On Auxiliary Chains: Auxiliaries at the Syntax-Semantics Interface. *Borealis* 4(2). 71-101. http://dx.doi.org/10.7557/1.4.2.3612

Breitling, Rainer (2005) Modelling biological systems with differential equations. Ms. Bioinformatics Research Center, Glasgow University.

Bresnan, Joan (1971) Sentence Stress and Syntactic Transformations. *Language* 47(2). 257–81.

(1976) Transformations and categories in syntax. In Ronald Butts & Jaakko Hintikka (eds.) *Basic problems in methodology and linguistics*. Dordrecht: Reidel. 261–282.

(1977) Variables in the theory of Transformations. In Culicover, Peter, Thomas Wasow, and Adrian Akmajian (eds.) *Formal Syntax*. New York: Academic Press. 157-196.

(2001) *Lexical-Functional Syntax*. Oxford: Blackwell.

(2017) Linguistics: The Garden and the Bush. *Computational Linguistics* 42(4). 599-617.

Bresnan, Joan, Ash Asudeh, Ida Toivonen, and Stephen Wechsler (2016) *Lexical functional syntax, 2nd edition*. Oxford: Blackwell.

Brody, Michael (1995) *Lexico-Logical Form: A Radically Minimalist Theory*. Cambridge, Mass.: MIT Press.

Brøndal, Viggo (1928) *Ordclasserne (partes orationis)*. Copenhagen: Gad.

Bromberger, Sylvain (1992) Types and Tokens in Linguistics. In Sylvain Bromberger, *On What We Know We Don't Know*. Chicago: University of Chicago Press. 170-208.

Brown, Joshua (2014) The tale of the neuroscientists and the computer: why mechanistic theory matters. *Front. Neurosci.* 8: 349. http://dx.doi.org/10.3389/fnins.2014.00349

Buchmüller, Wilfried and Christoph Lüdeling (2006) Field Theory and Standard Model. arXiv:hep-ph/0609174v1

Calabretta Rafaele, & Domenico Parisi (2005) Evolutionary connectionism and mind/brain modularity. In Werner Callebaut & Diego Rasskin-Gutman (eds.) *Modularity. Understanding the development and evolution of complex natural systems*. Cambridge, Mass.: MIT Press. 309–330.

Caracciolo, Sergio and Luigi Radicati (1989) Entropy Ultrametric for Dynamical and Disordered Systems. *Journal de Physique* 50(19). 2919-2930.

Carandini, Matteo (2012). From circuits to behavior: a bridge too far? *Nat. Neurosci*. 15. 507–509. doi: 10.1038/nn.3043

Carr, Jack (2006) Center Manifold. Scholarpedia, 1(12).1826.

Carruthers, Peter (2006) *The Architecture of the Mind: massive modularity and the flexibility of thought*. Oxford: OUP.

Chater, Nick (1996) Reconciling Simplicity and Likelihood principles in perceptual organization. *Psychological Review* 103(3). 566-581.

Chater, Nick and Paul Vitányi (2003) Simplicity: a unifying principle in cognitive science? *Trends in Cognitive Sciences* 7(1). 19-22.

Chaves, Rui (2012) On the grammar of extraction and coordination. *Natural Language & Linguistic Theory* 30. 465-512.

Chemero, Anthony (2009) *Radical Embodied Cognitive Science.* Cambridge, MA: MIT Press.

Cherniak, Christopher (2009) Brain Wiring Optimization and Non-Genomic Nativism. In Piattelli-Palmarini, Massimo, Juan Uriagereka, and Pello Salaburu (eds.) *Of Minds and Language*. Oxford: OUP. 108-119.

Chomsky, Noam (1955) *The Logical Structure of Linguistic Theory*. Mimeographed, MIT. http://alpha-leonis.lids.mit.edu/wordpress/wp-content/uploads/2014/07/chomsky_LSLT55.pdf [accessed 15/5/2015]

(1956) Three models for the description of language. *IRE Transactions on Information Theory* 2(3). 113-124.

(1957) *Syntactic Structures*. The Hague: Mouton.

(1959) On Certain Formal Properties of Grammars. *Information and Control* 2. 137-167.

(1963) Formal Properties of Grammars. In R. D. Luce, R. R. Bush, and E. Galanter (eds.), *Handbook of Mathematical Psychology*. New York: John Wiley & Sons. 323–418.

(1965) *Aspects of the Theory of Syntax*. Cambridge, Mass.: MIT Press.

(1966) *Cartesian Linguistics: A Chapter in the History of Rationalist Thought*. Cambridge: CUP.

(1970) Remarks on nominalization. In Roderick Jacobs and Peter Rosenbaum (eds.) *Readings in English Transformational Grammar*. Waltham: Ginn. 184-221.

(1973) Conditions on transformations. In Stephen Anderson and Paul Kiparsky (eds.) *A festschrift for Morris Halle*. New York: Holt, Rinehart, and Winston. 232–286.

(1981a) *Lectures on Government and Binding.* Dordrecht: Foris.

(1981b) Principles and parameters in syntactic theory. In Norbert Hornstein and David Lightfoot (eds.) *Explanation in Linguistics.* London: Longman. 32-76.

(1986a) *Barriers*. Cambridge, Mass.: MIT Press.

(1986b) *Knowledge of Language: Its Nature, Origin, and Use*. New York: Praeger Publishing

(1995) *The Minimalist Program*. Cambridge, Mass.: MIT Press.

(2002) *On Nature and Language*. Cambridge: CUP.

(2004) Beyond explanatory adequacy. In Adriana Belleti and Luigi Rizzi (eds.) *Structures and beyond*. Oxford: Oxford University Press. 104-131.

(2005) The Biolinguistic Perspective after 50 years. URL: http://www-old.accademiadellacrusca.it/img_usr/Chomsky%20-%20ENG.pdf [Accessed on 21/08/2011]

(2007) Approaching UG from Below. In Uli Sauerland and Hans-Martin Gartner (eds.), *Interfaces + recursion = language? Chomsky's minimalism and the view from syntax-semantics*. Berlin: Mouton de Gruyter. 1-29.

(2008) On Phases. In Robert Freidin, Carlos P. Otero, and Maria Luisa Zubizarreta (eds.) *Foundational issues in linguistic theory*. Cambridge, Mass.: MIT Press. 133-166.

(2009) Opening Remarks. In Massimo Piattelli-Palmarini, Juan Uriagereka, and Pello Salaburu (eds.) *Of Minds and Language*. Oxford: OUP. 13-43.

(2013) Problems of Projection. *Lingua* 130. 33-49.

(2015) Problems of Projection: Extensions. In Elisa Di Domenico, Cornelia Hamann and Simona Matteini (eds.) *Structures, Strategies and Beyond: Studies in honour of Adriana Belletti*. Amsterdam: John Benjamins. 1-16.

Chomsky, Noam and George Miller (1963) Introduction to the Formal Analysis of Natural Languages. In Duncan R. Luce, Robert R. Bush, and Eugene Galanter (eds.), *Handbook of Mathematical Psychology 2*. New York: John Wiley & Sons. 269–321.

Chung, Sandra, William Ladusaw, and James McClosky (1995) Sluicing and Logical Form. *Natural Language Semantics* 3. 239-282.

Church, Alonzo (1936) A note on the Entscheidungsproblem. *Journal of Symbolic Logic* 1. 40–41.

Cinque, Guglielmo (1999) *Adverbs and Functional heads: A Cross-Linguistic Perspective*. Oxford: OUP. 51-65.

(2014) On the movement account of Greenberg's Universal 20: Refinements and Replies. Ms., University of Venice.

Cinque, Guglielmo and Luigi Rizzi (2009) The Cartography of Syntactic Structures. In Bernd Heine and Heiko Narrog (eds.) *The Oxford Handbook of Linguistic Analysis*. Oxford: OUP. 51-65.

Citko, Barbara (2011) Multidominance. In Cedric Boeckx (ed.) *The Oxford Handbook of Linguistic Minimalism*. Oxford: OUP. DOI:10.1093/oxfordhb/9780199549368.013.0006

Conway, Andrew, Michael Kane, Michael Bunting, Zach Hambrick, Oliver Wilhelm, and Randall Engle (2005) Working memory span tasks: A methodological review and user's guide. *Psychonomic Bulletin & Review* 12(5). 769-786.

Collins, Chris and Edward Stabler (2016) A formalization of Minimalist syntax. *Syntax* 19(1). 43-78. DOI: 10.1111/synt.12117

Copeland, Jack (2002a) The Church-Turing Thesis. *Stanford Encyclopedia of Philosophy*. http://plato.stanford.edu/archives/sum2015/entries/church-turing/ [Accessed on 30/09/2015]

(2002b) Hypercomputation. *Minds and Machines* 12. 461–502.

Cosmelli, Diego and Evan Thompson (2010) Embodiment or Envatment? Reflections on the Bodily Basis of Consciousness. In John Stewart, Olivier Gapenne, and Ezequiel A. Di Paolo (eds.) *Enaction: Towards a New Paradigm for Cognitive Science*. Cambridge, MA: MIT Press. 361–385.

Courant, Richard and Fritz John (1965) *Introduction to Calculus and Analysis, Volume 1*. New York: Interscience Publications.

Courant, Richard and Fritz John (1974) *Introduction to Calculus and Analysis, Volume II. 2nd Edition*. New York: Interscience Publications.

Cowan, Nelson (2001) The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioral and Brain Sciences* 24. 87-185.

Croft, William (2001) *Radical Construction Grammar*. Oxford: OUP.

Culicover, Peter (1999) *Syntactic Nuts: Hard Cases in Syntax*. Oxford: Oxford University Press.

(2006) Syntactic nuts, the core-periphery distinction, and universal grammar. Ms. Ohio State University. http://www.ling.ohio-state.edu/~culicove/Publications/Syntactic%20nuts,%20the%20core-periphery%20distincti.pdf [Accessed on 30/07/2015]

(2013) The role of linear order in the computation of referential dependencies. *Lingua* 136. 125-144. http://dx.doi.org/10.1016/j.lingua.2013.07.013

Culicover, Peter and Ray Jackendoff (2005) *Simpler Syntax*. Oxford: OUP.

(2013) The Simpler Syntax Hypothesis. In Peter Culicover (2013) *Explaining Syntax*. Oxford: OUP. 1-11.

Damper, Robert, and Stevan Harnad (2000). Neural network models of categorical perception. *Perception and Psychophysics* 62. 843–867.

Daneman, Meredyth, and Patricia Carpenter (1980). Individual differences in working memory and reading. *Journal of Verbal Learning and Verbal Behavior* 19. 450–466.

Davis, Martin (1958) *Computability & unsolvability*. McGraw-Hill.

De Belder, Marijke (2011) *Roots and Affixes: Eliminating lexical categories from syntax.* Utrecht: LOT Publications.

De Belder, Marijke and Jeroen van Craenenbroeck (2011) How to merge a root. Ms., HUBrussel & Utrecht University. Available online at http://www.crissp.be/wp-content/uploads/papermjroots.pdf [Accessed on 25/11/2014]

Debusmann Ralph & Marco Kuhlmann (2010) Dependency Grammar: Classification and Exploration. In Matthew W. Crocker & Jörg Siekmann (eds) *Resource-Adaptive Cognitive Processes. Cognitive Technologies*. Springer, Berlin, Heidelberg. 365-388.

Dehaene, Stanislas (2011) *The Number Sense*. [2nd ed.] Oxford: OUP.

Dell, Gary, Franklin Chang, and Zenzi Griffin (1999) Connectionist Models of Language Production: Lexical Access and Grammatical Encoding. *Cognitive Science* 23(4). 517-542.

De Saussure, Ferdinand (1916) *Cours de linguistique générale* Charles Bally and Albert Sechehaye (eds.). Lausanne and Paris: Payot.

Deutsch, David (1985) Quantum theory, the Church-Turing principle and the universal quantum computer. *Proceedings of the Royal Society of London A*, 400. 97-117.

Van Dijk, Teun (1992) *La Ciencia del Texto: Un Enfoque Interdisciplinario*. Barcelona: Paidós.

den Dikken, Marcel (2007): Phase Extension: Contours of a Theory of the Role of Head Movement in Phrasal Extraction, *Theoretical Linguistics* 33. 1–41.

Dineva, Evelina, Gregor Schöner, and Esther Thelen (2008) Behavioral history matters: A dynamic field theory of spontaneous and perseverative errors in infant reaching. Ms.

Dirac, Paul A. M. (1958) *The Principles of Quantum Mechanics* [4th edition]. Oxford: Clarendon Press.

Di Sciullo, Anna-Maria and Daniela Isac (2008) The Asymmetry of Merge. *Biolinguistics* 2(4). 260-290.

Di Sciullo, Anna-Maria and Cedric Boeckx (eds.) (2011) *The Biolinguistic Enterprise: New Perspectives on the Evolution and Nature of the Human Language Faculty*. Oxford: OUP.

Dotov, Dobromir (2014) Putting reins on the brain. How the body and environment use it. *Front. Hum. Neurosci*., 09 October 2014. doi: 10.3389/fnhum.2014.00795

Dotsenko, Viktor (1995) *An Introduction to the Theory of Spin Glasses and Neural Networks*. World Scientific.

Dowty, David (1979) *Word Meaning and Montague Grammar*. Dordrecht: Reidel.

Einstein, Albert (1905) Concerning a Heuristic Point of View Towards the Emission and Transformation of Light. *Annalen der Physik* 17. 132-148. Available online at http://einsteinpapers.press.princeton.edu/vol2-trans/101 [Accessed on 18/12/2014]

Elman, Jeffrey (1993) Learning and development in neural networks: The importance of starting small. *Cognition* 48. 71-99.

Embick, David and Rolf Noyer (2007) Distributed Morphology and the Syntax/Morphology Interface, In Ramchand, Gillian (ed.) *The Oxford Handbook of Linguistic Interfaces*. Oxford: Oxford University Press. 289–324.

Embick, David and David Poeppel (2014) Towards a computational(ist) neurobiology of language: correlational, integrated and explanatory neurolinguistics. *Language, Cognition and Neuroscience*, 2014. http://dx.doi.org/10.1080/23273798.2014.980750

Emonds, Joseph (1970) *Root and Structure Preserving Transformations*. PhD Thesis, MIT.

(1972) A reformulation of certain syntactic transformations. In Stanley Peters (ed.) *Goals of Linguistic Theory*. Englewood Cliff, NJ.: Prentice-Hall. 21-62.

(1976) *A transformational approach to English syntax: Root, structure preserving, and local transformations*. New York: Academic Press.

(2007) *Discovering Syntax: Clause Structures of English, German and Romance*. Walter de Gruyter.

(2012) Augmented Structure Preservation and the *Tensed S Constraint*. In Loebke Albrecht, Liliane Haegeman & Rachel Nye (eds.) *Main Clause Phenomena: New Horizons*. Amsterdam: John Benjamins. 23-46.

Epstein, Samuel (2000) *Essays in Syntactic Theory*. London: Routledge.

Epstein, Samuel and T. Daniel Seely (2002) Rule applications as cycles in a level-free syntax. In Samuel Epstein & T. Daniel Seely (eds) *Explanation and Derivation in the Minimalist Program*. Oxford: Blackwell. 65–89.

(2006) *Derivations in Minimalism*. Cambridge: Cambridge University Press.

Escandell-Vidal, María Victoria and Manuel Leonetti (2000) Categorías conceptuales y semántica procedimental. In *Cien años de investigación semántica: de Michél Bréal a la actualidad. Tomo I*. Madrid: Ediciones Clásicas. 363-378.

Escandell-Vidal, María Victoria and Manuel Leonetti (2002) Coercion and the Stage/Individual Distinction. In *From Words to Discourse. Trends in Spanish Semantics and Pragmatics*, J. Gutiérrez-Rexach (ed.). Amsterdam: Elsevier. 159-180.

Evans, Vyvyan (2009) *How Words Mean: Lexical Concepts, Cognitive Models, and Meaning Construction*. Oxford: OUP.

Everett, Daniel (2005) Cultural Constraints on Grammar and Cognition in Pirahã. *Current Anthropology* 46(4). 621-646.

(2009) You drink. You drive. You go to jail. Where's recursion? Paper presented in the *UMass conference on Recursion*, May 2009. http://ling.auf.net/lingbuzz/001141 [Accessed on 22/05/2014]

(2015) Review article: David Adger's *A Syntax of Substance*. *American Anthropologist*, 117(2). 414-415.

(2016) *Dark Matter of the Mind: The Culturally Articulated Unconscious*. Chicago: University of Chicago Press.

Falcade, Rossana, Maria Alessandra Mariotti & Colette Laborde (2004) Towards a definition of function. *Proceedings of the 28th Conference of the International Group for the Psychology of Mathematics Education*, 2004 Vol 2. 367–374.

Falconer, Kenneth (2014) *Fractal Geometry*. London: Wiley.

Farah, Martha (1994) Neuropsychological inference with an interactive brain: A critique of the "locality" assumption. *Behavioral and Brain Sciences* 17. 43-104.

Feigenson, Lisa (2011) Objects, Sets, and Ensembles. In Stanislas Dehaene and Elizabeth Brannon (eds.) *Space, Time and Number in the Brain: Searching for the Foundations of Mathematical Thought*. London: Elsevier. 13-22.

Fels, Daniel, Michal Cifra, and Felix Scholkmann (eds.) (2015) *Fields of the Cells* [Open Access e-Book]. Research Signpost. Available online at https://www.researchgate.net/publication/281409607_Fields_of_the_cell [Accessed on 17/02/2015]

Feynman, Richard P. (1942) *The Principle of Least Action in Quantum Mechanics*. PhD Thesis, Princeton University.

(1972) *Statistical Mechanics: A Set of Lectures*. Reading, Mass.: W. A. Benjamin.

(2006) [1985] *QED: The Strange Theory of Light and Matter*. Princeton University Press.

(2011) [1963] *The Feynman lectures on Physics*. 3 Vols. Available online at http://www.feynmanlectures.caltech.edu/III_toc.html [Accessed on 27/01/2015]

Feynman, Richard P., Albert Hibbs, and Daniel Styer (2005) *Quantum Mechanics and Path Integrals*. New York, Dover.

Fiengo, Robert (1977) On trace theory. *Linguistic Inquiry* 8. 35–61.

Fillmore, Charles J. (1963) The Position of Embedding Transformations in a Grammar. *Word* 19(2). 208-231.

(1968) The case for case. In Emmon Bach and Robert Harms (eds.) *Universals in Linguistic Theory*. New York: Holt, Rinehart, and Winston. 1-89.

Fitch, Tecumseh (2010) *The Evolution of Language*. Cambridge: CUP.

Fitz, Harmut (2006) Church's Thesis and Physical Computation. In Adam Olszewski, Jan Wolenski and Robert Janusz (eds.) *Church's Thesis After 70 Years*. Heusenstamm: Ontos. 175-219.

FitzHugh Richard (1961) Impulses and physiological states in theoretical models of nerve membrane. *Biophysical* J. 1. 445–466.

Fodor, Jerry (1975) *The Language of Thought*. New York: Thomas Crowell.

Fodor, Jerry (1983) *Modularity of Mind: An Essay on Faculty Psychology*. Cambridge, Mass.: MIT Press.

Fodor, Jerry (2000) *The Mind Doesn't Work That Way*. Cambridge, Mass.: MIT Press.

Foucart, Alice, Elisa Ruiz-Tada & Albert Costa (2015) How do you know I was about to say "book"? Anticipation processes affect speech processing and lexical recognition. *Language, Cognition and Neuroscience* 30(6). 768-780.

Fraenkel, Aviezri (1990) De-exponentializing complex computational mathematical problems using physical or biological systems. Technical Report CS90-30. Department of Applied Mathematics and Computer Science, Weizmann Institute of Science.

(1993) Complexity of Protein Folding. *Bulletin of Mathematical Biology* 55(6). 1199 1210.

František Jelínek, Jiří Pokorný, Jaroslav Šaroch, and Jiří Hašek (2005) Experimental investigation of electromagnetic activity of yeast cells at millimeter waves. *Electromagnetic Biology and Medicine* 24 (3). 301–308.

František Jelínek, Michal Cifra, Jiří Pokorný, Jan Vaniš, Jan Šimša, Jiří Hašek, and Ivana Frýdlováb (2009) Measurement of electrical oscillations and mechanical vibrations of yeast cells membrane around 1 kHz. *Electromagnetic Biology and Medicine* 28(2). 223–232.

Freeman, Walter (2000) A proposed name for aperiodic brain activity: stochastic chaos. *Neural Networks* 13. 11-13.

Frege, Gottlob (1891) Funktion und Begriff. English translation in *The Frege Reader*, ed. by Michael Beaney, 1997. London: Blackwell. 130–148.

(1952) On Sense and Reference. In Peter Geach and Max Black (eds.) *Translations from the Philosophical Writings of Gottlob Frege.* Oxford: Blackwell. 56-78.

Fromm, Jochen (2004) *The Emergence of Complexity*. Kassel University Press.

Fujita, Koji and Cedric Boeckx (eds.) (2016) *Advances in Biolinguistics: The Human Language Faculty and its Biological Basis.* London: Routledge.

Gallego, Ángel (2007) *Phase Theory and Parametric Variation*. PhD thesis, Universitat Autónoma de Barcelona, Barcelona.

(2010) *Phase Theory*. Amsterdam: John Benjamins.

(ed.) (2012) *Phases: Developing the Framework*. Berlin: Walter De Gruyter.

Gallego, Angel and Juan Uriagereka (2007) Conditions on Sub-Extraction. In Luis Eguren and Olga Fernández-Soriano (eds.), *Correference, Modality, and Focus: Studies on the syntax–semantics interface*. Amsterdam: John Benjamins. 45–70.

Galley, Michel & Kathleen R. McKeown (2007) Lexicalized Markov Grammars for Sentence Compression. *Proceedings of NAACL HLT 2007*. 180–187.

Gallistel, Charles and Adam King (2010) *Memory and the computational brain: Why cognitive science will transform neuroscience.* London: Blackwell.

García Fernández, Luis and Diego Krivochen (2018) Variability in syntactic-semantic cycles: evidence from auxiliary chains. In press.

García Fernández, Luis, Diego Gabriel Krivochen and Ana Bravo (2017) Aspectos de la semántica y sintaxis de las cadenas de verbos auxiliares en español. *Moenia*, 23.

Gazdar, Gerald (1981) Unbounded Dependencies and Coordinate Structure. *Linguistic Inquiry*, 12(2). 155-184.

Gärdenfors, Peter (1997) Meanings as conceptual structures. In Martin Carrier and Peter Machamer (eds.) *Mindscapes: Philosophy, Science, and the Mind*. Pittsburgh: Pittsburgh University Press. 61-86.

Van Geert, Paul (2003) Dynamic systems approaches and modeling of developmental processes. In Jaan Valsiner & Kevin J. Connolly (Eds.), *Handbook of developmental psychology*. London, England: Sage. 640–672.

Gelman, Andrew, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin (2013) *Bayesian Data Analysis*. CRC Press.

Georgopoulos, Carol (1985) Variables in Palauan syntax. *Natural Language & Linguistic Theory* 3. 59-94.

Gil, David (2009) How Much Grammar Does It Take to Sail a Boat? In Geoffrey Sampson, David Gil and Peter Trudgill (eds.), *Language Complexity as an Evolving Variable*. Oxford: OUP. 19-33.

Gödel, Kurt (1931) Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme, I. *Monatshefte für Mathematik und Physik 38*. 173-98. [Translated as 'On formally undecidable propositions of Principia Mathematica and related systems I', in Solomon Feferman, ed., 1986. *Kurt Gödel Collected works, Vol. I*. Oxford: Oxford University Press. 144-195]

Goldberg, Adele (2006) *Constructions at Work: the nature of generalization in language*. Oxford University Press.

Goldin, Dina and Peter Wegner (2005) The Church-Turing Thesis: Breaking the Myth. *LNCS 3526*. 152-168.

(2007) The Interactive Nature of Computing: Refuting the Strong Church–Turing Thesis. *Minds and Machines* 18 (1). 17-38.

Goldsmith, John (1985) A Principled Exception to the Coordinate Structure Constraint. *Papers from the general session at the twenty-first regional meeting of the Chicago Linguistic Society*. University of Chicago, Illinois. Available online at https://www.researchgate.net/publication/244998803_A_Principled_Exception_to_the_Coordinate_Structure_Constraint [Accessed on 13/6/2015]

Goldstone, Robert and Andrew Hendrickson (2009) Categorical Perception. *WIREs Cognitive Science*. Available online at http://pubs.cogs.indiana.edu/pubspdf/23349/23349_catperc.pdf [Accessed on 6/1/2015]

Goldwater, Sharon (2006) *Non Parametric Bayesian Models of Language Acquisition*. Ph.D. thesis, Brown University.

Goodall, Grant (1984) *Parallel Structures in Syntax*. PhD Thesis, University of California, San Diego.

(2009) *Parallel Structures in Syntax*. Cambridge: CUP.

Green, Georgia (2011) Elementary Principles of Head-Driven Phrase Structure Grammar. In Robert Borsley and Kersti Börjars (eds.) *Non-Transformational Syntax*. Oxford: Blackwell. 9-53.

Greibach, Sheila (1965) A New Normal-Form Theorem for Context-Free Phrase Structure Grammars. *Journal of the ACM* **12** (1). 42-52.

Grice, H. Paul (1957) Meaning. *The Philosophical Review* 66(3). 377-388

(1975) Logic and Conversation. In Peter Cole and Jerry Morgan (eds.) *Syntax and Semantics, vol. III: Speech Acts*. London: Elsevier. 41-58.

Grindrod, Peter (2015) *Mathematical Underpinnings of Analytics: Theory and Applications*. Oxford: OUP.

Grindrod, Peter & Tasmin Lee (2017) On strongly connected networks with excitable-refractory dynamics and delayed coupling. *R. Soc. open sci*. 4: 160912. http://dx.doi.org/10.1098/rsos.160912

Grindrod, Peter; Zhivko Stoyanov; Garry Smith; and James Douglas Saddy (2014) Resonant Behaviour of Irreducible Networks of Neurons with Delayed Coupling. Ms.

Grohmann, Kleanthes K. (2003) *Prolific Domains. On the Anti-Locality of Movement Dependencies*. Amsterdam: John Benjamins.

Guckenheimer, John and Philip Holmes (1997) *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*. Berlin, New York: Springer-Verlag.

Guyton, Arthur and John Hall (2016) Membrane potentials and action potentials. In *Medical Physiology*. 13th Edition. Elsevier. 57-70.

Hale, Kenneth (1986) Notes on world view and semantic categories: some Warlpiri examples. In Peter Muysken & Henk van Riemsdijk (eds.) *Features and Projections*. Dordrecht: Foris. 233-254.

Hale, Kenneth and Samuel Jay Keyser (1997) On the complex nature of simple predicators. In Alex Alsina, Joan Bresnan, and Peter Sells (eds.) *Complex Predicates*, CSLI Lecture Notes 64. Stanford: CSLI Publications. 29-65

Hale, Kenneth and Samuel Jay Keyser (2002) *Prolegomenon to a Theory of Argument Structure*. Cambridge, Mass.: MIT Press.

Halle, Morris and Alec Marantz (1993) Distributed Morphology and the Pieces of Inflection. In Ken Hale and Samuel Jay Keyser (eds.) *The View from Building 20*. Cambridge, Mass.: MIT Press. 111-176.

Halliday, Michael Alexander Kirkwood and Ruqaiya Hasan (1976) *Cohesion in English*. London: Longman.

Hameroff, Stuart, and Roger Penrose (2013) Consciousness in the universe: A review of the 'Orch OR' theory. *Physics of Life Reviews* 11(1). 39–78.

Hansel, David, Germán Mato, and Claude Meunier (1995) Synchrony in excitatory neural networks. *Neural computation* 7(2). 307-337.

Haragus, Mariana, and Gérard Iooss (2011) *Local Bifurcations, Center Manifolds, and Normal Forms in Infinite-Dimensional Dynamical Systems*. London: Springer.

Harley, Heidi (2012) External arguments and the Mirror Principle: On the distinctness of Voice and *v*. *Lingua* 125. 34-57.

Harnad, Steven (1987) *Categorical Perception: The Groundwork of Cognition*. New York: Cambridge University Press.

Harris, Zellig (1946) From Morpheme to Utterance. *Language* 22(3).161–183

(1951) *Methods in Structural Linguistics*. Chicago: University of Chicago Press.

(1957) Co-Occurrence and Transformation in Linguistic Structure. *Language*, 33(3). 283-340.

Hartmanis, Juris (1989) Gödel, von Neumann, and the P = NP problem. *Bulletin of the European Association for Theoretical Computer Science* 38. 101–107.

Haspelmath, Martin (2015) Defining vs. diagnosing linguistic categories: a case study of clitic phenomena. In Joanna Baszczak, Dorota Klimek-Jankowska and Krzysztof Migdalski (eds.) *How categorical are categories?* Berlin: De Gruyter.

Hasselblat, Boris and Anatole Katok (2003) *Introduction to the Modern Theory of Dynamical Systems*. CUP.

Hauser, Marc (2009) Discussion with Christopher Cherniak. In Massimo Piattelli-Palmarini, Juan Uriagereka, and Pello Salaburu (eds.) *Of Minds and Language*. Oxford: OUP. 118.

Hauser, Marc, Noam Chomsky, and Tecumseh Fitch (2002) The Language Faculty: What is it, who has it, and how did it evolve? *Science* 298. 1569–1579.

Hefferon, Jim (2014) *Linear Algebra*. Available online at http://joshua.smcvt.edu/linearalgebra/book.pdf [Retrieved on 5/1/2015]

Heim, Irene (1982) *The Semantics of Definite and Indefinite Noun Phrases*. PhD dissertation, University of Massachusetts Amherst. Available online at http://semarch.linguistics.fas.nyu.edu/Archive/jA2YTJmN/Heim%20Dissertation%20with%20Hyperlinks.pdf [Retrieved on 13/11/2015]

Heinz, Jeffrey (2007) *Inductive Learning of Phonotactic Patterns*. PhD Thesis, UCLA. Available online at http://www.linguistics.ucla.edu/general/Dissertations/heinz-2007-UCLA-diss.pdf [Retrieved on 5/1/2015]

Helmberg, Gilbert (2007) *Getting acquainted with Fractals*. Berlin, New York: Walter de Gruyter.

Von Helmholtz, Herman (1962) *Treatise on physiological optics* (Vol. 3). New York: Dover.

Van Heuveln, Bram (2015) Recursive sets and relations. Lecture notes for *Computability and Logic*. Available online at http://www.cogsci.rpi.edu/~heuveb/teaching/Logic/CompLogic/Web/Presentations/Recursive%20Sets%20and%20Relations.pdf [Retrieved on 20/5/2016]

Heycock, Caroline and Roberto Zamparelli (2005) Friends and colleagues: plurality, coordination, and the structure of DP. *Natural Language Semantics* 13. 201-270.

Hilbert, David and Wilhelm Ackermann (1928). *Grundzüge der theoretischen Logik* [Principles of Mathematical Logic]. Springer-Verlag.

Hilpert, Martin (2014) *Construction Grammar and its Application to English*. Edinburgh University Press.

Hjelmslev, Louis (1961) *Prolegomena to a theory of language*. Madison: The University of Wisconsin Press.

Hockett, Charles (1954) Two models of grammatical description. *Word* 10. 210–234.

(1958) *A Course in Modern Linguistics*. The Macmillan Company: New York.

Hofstadter, Douglas R. (1979) *Gödel, Escher, Bach: an Eternal Golden Braid*. Harvester Press.

Holpert, David (2008) Comparing complex systems. Response to Binder (2008). *Science*. Available online: http://www.sciencemag.org/content/320/5874/322.full/reply#sci_el_10903 [Accessed 27/05/2015]

Holzer, Markus and Peter Rossmanith (1996) A Simpler Grammar for Fibonacci Numbers. http://www.fq.math.ca/Scanned/34-5/holzer.pdf [Accessed on 12/11/2014]

Hopcroft, John and Jeffrey Ullman (1969) *Formal Languages and their relation to Automata*. London: Addison-Wesley.

Horn, Lawrence (1988) Pragmatic Theory. In Frederick Newmeyer (ed.) *Linguistics: The Cambridge Survey. Volume I, Linguistic Theory: Foundations*. Cambridge: Cambridge University Press. 113-145.

Hornstein, Norbert (1995) *Logical Form: From GB to Minimalism*. New York: Wiley and Sons.

(2001) *Move! A Minimalist Theory of Construal*. London: Wiley.

(2009) *A Theory of Syntax: Minimal Operations and Universal Grammar*. Cambridge, Mass.: CUP.

Hornstein, Norbert & Paul Pietroski (2009) Basic Operations: Minimal Syntax-Semantics. *Catalan Journal of Linguistics* 8. 113-139.

Horst, Steven (2015) The Computational Theory of Mind. In Edward N. Zalta (ed.) *The Stanford Encyclopedia of Philosophy* (Summer 2015 Edition). Available in http://plato.stanford.edu/archives/sum2015/entries/computational-mind/ [Accessed on 10/08/2015]

Huang, C. T. James (1982) *Logical relations in Chinese and the theory of grammar*, Ph.D. Dissertation, MIT.

Hughes, Bruce (2003) Trees and ultrametric spaces: a categorical equivalence. *Advances in Mathematics* 189. 148–191.

Jackendoff, Ray (1983) *Semantics and Cognition*. Cambridge, Mass.: MIT Press.

(2002) *Foundations of Language*. Oxford: OUP.

(2011) Alternative minimalist visions of language. In Robert Borsley & Kresti Börjars (eds.), *Non-transformational syntax. Formal and explicit models of grammar.* London: Blackwell. 268–296.

Jackendoff, Ray and Eva Wittenberg (2014) What You Can Say Without Syntax: A hierarchy of grammatical complexity. In Frederick J. Newmeyer and Laurel B. Preston (eds.) *Measuring Grammatical Complexity*. Oxford: OUP. 65-82.

(2017) Linear grammar as a possible stepping-stone in the evolution of language. *Psychonomic Bulletin & Review* 24(1). 219–224.

Jacobson, Pauline (1977) *The Syntax of Crossing Correference Sentences*. PhD Thesis, University of California, Berkeley.

(1997) Where (if anywhere) is transderivationality located? In Peter Culicover and Louise McNally (eds.) *The Limits of Syntax*. New York: Academic Press. 303-336.

Jakubowicz, Celia (2011) Measuring derivational complexity: New evidence from typically developing and SLI learners of L1 French. *Lingua* 121(3). 339-351.

Jäger, Gerhard (2002) Some Notes on the Formal Properties of Bidirectional Optimality Theory. *Journal of Logic, Language, and Information* 11(4). 427-451.

Jeong, Youngmi (2003) Deriving Anti-Superiority effects. In Cedric Boeckx and Kleanthes Grohmann (eds.) *Multiple Wh-Fronting*. Amsterdam: John Benjamins. 131–140.

Jespersen, Otto (1933) *Essentials of English Grammar*. George Allan & Unwin Ltd.

Johansson, Fredrik, Dan-Anders Jirenhed, Anders Rasmussen, Riccardo Zucca, and Germund Hesslow (2014) Memory trace and timing mechanism localized to cerebellar Purkinje cells. *PNAS* 111(41). 14930–14934. doi: 10.1073/pnas.1415371111

Johansson, Sverker (2013) Biolinguistics or Physicolinguistics? Is the Third Factor Helpful or Harmful in Explaining Language? *Biolinguistics* 7. 249–275.

Joshi, Aravind K. (1985) Tree adjoining grammars: How much context-sensitivity is required to provide reasonable structural descriptions? In David Dowty, Lauri Karttunen, and Arnold Zwicky (eds.) *Natural Language Parsing*. Cambridge, Mass.: CUP. 206-250.

Joshi, Aravind K. and Takashi Yokomori (1983) Parsing of tree adjoining grammars. Technical Report. Department of Computer and Information Science, University of Pennsylvania.

Joshi, Aravind K. and Yves Schabes (1991) Tree-Adjoining Grammars and Lexicalized Grammars. *Technical Reports (CIS).* Paper 445. http://repository.upenn.edu/cis_reports/445

Kamide, Yuki (2008) Anticipatory Processes in Sentence Processing. *Language and Linguistics Compass* 2(4). 647–670.

Kandel, Eric and James Schwartz (1991) Directly Gated Transmission at Central Synapses. In Eric Kandel, James Schwartz and Thomas Jessell (eds.) *Principles Of Neural Science*. 3rd Edition. Norwalk: Appleton and Lange. 153-172.

Kaplan, Ronald (1972) Augmented transition networks as psychological models of sentence comprehension. *Artifcial Intelligence* 3. 77-100.

(1995) The Formal Architecture of Lexicla Functional Grammar. In Mary Dalrymple, Ronald Kaplan, John T. Maxwell III & Annie Zaenen (eds.) *Formal Issues in Lexical Functional Grammar*. Stanford: CSLI. 1-21.

Kaplan, Roland and Joan Bresnan (1982) Lexical-Functional Grammar: A Formal System for Grammatical Representation. In Joan Bresnan (ed.) *The Mental Representation of Grammatical Relations*. Cambridge, Mass.: MIT Press. 173-281.

Karttunen, Lauri (1971) Definite Descriptions with Crossing Reference. *Foundations of Language* 7. 157-182.

Karttunen, Lauri & Martin Kay (1985) Structure sharing with binary trees. *Proceedings of the 23rd Association for Computational Linguistics*. Chicago, Illinois: University of Chicago. 133-136.

Katz, Jerrold & Paul M. Postal (1964) *An Integrated Theory of Linguistic Descriptions*. Cambridge, Mass.: MIT Press.

(1991) Realism vs. conceptualism in linguistics. *Linguistics and Philosophy* 14. 515-554.

Kayne, Richard (1984) *Connectedness and Binary Branching*. Dordretch: Foris.

(1994) *The Antisymmetry of Syntax*. Cambridge, Mass.: MIT Press.

(2011) Antisymmetry and the Lexicon. In Anna-Maria Di Sciullo & Cedric Boeckx (eds.) *The Biolinguistic Enterprise: New Perspectives on the Evolution and Nature of the Human Language Faculty*. Oxford: OUP. 329-53.

Kelso, Scott (1995) *Dynamic Patterns: The Self-Organization of Brain and Behaviour*. Cambridge, Mass.: MIT Press.

Kelso, Scott; J. DelColle and Gregor Schöner (1990) Action-Perception as a pattern formation process. In Marc Jeannerod (Ed.), *Attention and Performance* XIII. Hillsdale, NJ.: Erlbaum. 139–169

Kilmer, W., W. S. McCulloch and J. Blum (1969) A model of the vertebrate central command system. *International Journal of Man-Machine Studies* 1(3). 279-309.

Kinsella, Anna (2009) *Language Evolution and Syntactic Theory*. Cambridge: CUP.

Kiparsky, Paul (1968) Linguistic universals and linguistic change. En Emmon Bach & Robert T. Harms (eds.) *Universals in linguistic theory*. New York: Holt, Rinehart & Winston. 171–202.

Kiparsky, Paul and Carol Kiparsky (1970) Fact. In Manfred Bierwisch and Karl Erich Heidolph (eds.), *Progress in Linguistics*. The Hague: Mouton. 143-173.

Kitahara, Hisatsugu (1996) Raising Q without QR. In Werner Abraham, Samuel Epstein, Höskuldur Thrainsson and Jan-Wouter Zwart (eds.), *Minimal ideas*. Amsterdam: John Benjamins. 189-198.

Knuth, Donald (1976) Big Omicron and big Omega and big Theta. *SIGACT News*, Apr-Jun 1976. 18-24.

(1998) *The Art of Computer Programming Vol. 3: Sorting and Searching*. Addison-Wesley.

Koester, John (1991) Voltage-Gated Ion Channels and the Generation of the Action Potential. In Eric Kandel, James Schwartz and Thomas Jessell (eds.) *Principles Of Neural Science*. 3rd Edition. Norwalk: Appleton and Lange. 104-118.

Kornai, Andras (1985) Natural Languages and the Chomsky Hierarchy. In M. King (ed): *Proceedings of the 2nd European Conference of the Association for Computational Linguistics 1985*. 1-7.

Kosta, Peter and Diego Krivochen (2012) Book Review: *The Biolinguistic Enterprise: New Perspectives on the Evolution and Nature of the Human Language Faculty*. Anna-Maria Di Sciullo and Cedric Boeckx (eds). Oxford: OUP. *International Journal of Language Studies*, 6(4). 154-182.

Kosta, Peter and Diego Gabriel Krivochen (2014a) Flavors of Movement: Revisiting the A-A' distinction. In Peter Kosta, Steven L. Franks, Teodora Radeva-Bork and Lilia Schürcks (eds.), *Minimalism and Beyond: Radicalizing the interfaces*. Amsterdam: John Benjamins. 236-266.

Kosta, Peter and Diego Gabriel Krivochen (2014b) Interfaces, Impaired Merge, and the Theory of the Frustrated Mind. *Czech and Slavic Linguistic Review* 1/2014. Special number: "Informational Fundamentals of Life: Genomes and Languages." Peter Kosta and Michaela Zemková (eds.). 32-75.

Koutsoudas, Andreas (1972) The Strict Order Fallacy. *Language* 48(1). 88-96.

Koyama, Masanori; David Neel; and Michael Orrison (2007) Irreducible Graphs. *Journal of Combinatorial Mathematics and Combinatorial Computing* 62. 35-43.

Kratzer, Angelika (1996) Severing the external argument from its verb. In Johan Rooryck and Laurie Zaring (eds.), *Phrase structure and the lexicon*. Dordrecht: Kluwer. 109–137.

Kripke, Saul (1980) *Naming and Necessity*. Cambridge, Mass.: Harvard University Press.

Krivochen, Diego (2011) An Introduction to Radical Minimalism I: on Merge and Agree. *IBERIA* 3(2). 20-62.

(2012a) *The Syntax and Semantics of the Nominal Construction*. Frankfurt am Main: Peter Lang.

(2012b) Dependencias (anti-)locales en latín. *Auster* 12. 97-122

(2014) Clitics, Procedural Elements, and Spanish Syntax. *ReVEL* 12(22). 43-79.

(2015a) On Phrase Structure building and Labeling algorithms: towards a non-uniform theory of syntactic structures. *The Linguistic Review* 32(3). 515-572.

(2015b) Types vs. Tokens: Displacement Revisited. *Studia Linguistica* 70(3). 250-296. DOI: 10.1111/stul.12044

(2015c) (Anti-)Causativity and the morphophonology-semantics tension. *Semantics-Syntax Interface* 1(2). 82-117.

(2015d) An Interface Approach to Binding: Spelling Out referential expressions as a function of syntactic-semantic construal. Ms. University of Reading. https://www.academia.edu/6375306/An_Interface_Approach_to_Binding_Spelling_Out_referential_expressions_as_a_function_of_syntactic-semantic_construal

(2016a) The Emptiness of the Realism-Conceptualism debate: Its consequences for (bio-) Linguistics. *Linguistic and Philosophical Investigations* 15. 115–166.

(2016b [2014c]) Language, chaos and entropy: A physical take on biolinguistics. *Iberia: IJTL*, 6. 27–74. https://ojs.publius.us.es/ojs/index.php/iberia/article/view/3006/2618

(2016c) Classical problems, non-classical proposals. Presented at *7th International Conference in Formal Linguistics*. Dec. 4th, Nankai University, Tianjin, China. https://www.academia.edu/30310835/_2016_Classical_problems_non-classical_proposals

(2016d) Divide and…conquer? On the limits of algorithmic approaches to syntactic structure. *Czech and Slovak Linguistic Review* 1(2016). 15-38.

(2018a) Dinamicidad topológica en fractales sintácticos. In Manuel Leonetti, Isabel Pérez Jiménez & Silvia Gumiel Molina (eds.) *Investigaciones actuales en Lingüística* Vol. III. Universidad de Alcalá. 79-93.

(2018b) Syntax as Graph Theory. Ms. University of Reading. DOI: 10.13140/RG.2.2.14367.25765/1 https://ling.auf.net/lingbuzz/003842

Krivochen, Diego and Peter Kosta (2013) *Eliminating Empty Categories: A Radically Minimalist view on the Ontology and Justification*. Frankfurt am Main: Peter Lang.

Krivochen, Diego and Katarzyna Mathiasen (2012) Numerals, Numbers and Cognition: Towards a Localist Theory of the Mind. Poster presented at *Cognitive Modules and Interfaces*, international workshop at SISSA, Trieste, Italy. http://ling.auf.net/lingbuzz/001653

Krivochen, Diego and Vladimir Matlach (2015) L-systems, parallel computing, and the limits of the Chomsky Hierarchy. Ms. University of Reading and Palacky University.

Krivochen, Diego & Douglas Saddy (2016) Structure mapping: Uniformity vs. Mixture in Displacement. *Czech and Slovak Linguistic Review* 2(2016). 16-45.

(2017) Classical Problems, Non-Classical Proposals: Notes on the Physics of Language. Ms. University of Reading.

Krivochen, Diego and Susan Schmerling (2016) Two kinds of coordination and their theoretical implications: An Aspect-Based Approach. Ms. Under Review.

Kroch, Anthony and Aravind K. Joshi (1985) Linguistic significance of TAG's. Ms. Available online at http://babel.ling.upenn.edu/papers/faculty/tony_kroch/papers/relevance3.pdf [Accessed on 22/03/2016].

Kuno, Susumu (1973) Constraints on Internal Clauses and Sentential Subjects. *Linguistic Inquiry* 4. 363-385.

Kuroda, Sige-Yuki (1964) Classes of languages and Linear-Bounded Automata. *Information and Control* 7. 207-223.

Ladusaw, William (1980) *Polarity sensitivity as inherent scope relations*. Bloomington, Indiana: University of Iowa, Indiana University Linguistics Club.

Ladyman, James, James Lambert, and Karoline Wiesner (2012) What is a complex system? *European Journal for Philosophy of Science*. 3(1). 33-67

Lakatos, Imre (1978) *The Methodology of Scientific Research Programmes: Philosophical Papers Volume 1*. Cambridge: Cambridge University Press.

Lakoff, George (1970) Linguistics and natural logic. *Synthese* 22(1–2). 151–271

(1971) On Generative Semantics. In Danny D. Steinberg and Leon A. Jakobovits (eds.) *Semantics - An Interdisciplinary Reader in Philosophy, Linguistics, Anthropology and Psychology*. Cambridge: Cambridge University Press.

(1976) Pronouns and Reference. In James D. McCawley (1976) (ed.). *Notes from the linguistic underground* (Syntax and Semantics 7). New York: Academic Press. 275-336.

Lakoff, George and John Robert Ross (1967) Is Deep Structure Necessary? Personal letter to Arnold Zwicky, printed in James D. McCawley (ed.) (1976). 159-164.

Landweber, Peter (1963) Three Theorems on Phrase Structure Grammars of Type 1. *Information and Control* 6. 131-136.

Langacker, Ronald (1969) On pronominalization and the chain of command. In David Reibel & Sanford Schane (eds.) *Modern Studies in English*. Engelwood Cliffs, NJ: Prentice-Hall. 160-186.

Langendoen, D. Terence (2010) Just how big are natural languages? In Harry van der Hulst (ed.), *Recursion and Human Language*. Berlin: De Gruyter Mouton. 139-146.

Langendoen, D. Terence and Paul M. Postal (1984) *The Vastness of Natural Languages*. Oxford: Blackwell.

Lapidus, Michel and Machiel van Frankenhuijsen (2006) *Fractal Geometry, Complex Dimensions and Zeta Functions: Geometry and Spectra of Fractal Strings*. Springer.

Lappin, Shalom, Robert D Levine, David E Johnson (2000) The structure of unscientific revolutions. *Natural Language & Linguistic Theory*, 18(3). 665-671

Lasnik, Howard (1999) *Minimalist Analysis*. Oxford: Blackwell.

(2011) What Kind of Computing Device is the Human Language Faculty? In Anna-Maria Di Sciullo & Cedric Boeckx (Eds.) *The Biolinguistic Enterprise: New Perspectives on the Evolution and Nature of the Human Language Faculty*. Oxford: OUP. 354-365.

Lasnik, Howard and Joseph Kuppin (1977) A restrictive theory of transformational grammar. *Theoretical Linguistics* 4. 173–196.

Lasnik, Howard and Juan Uriagereka (2012) Structure. In Ruth Kempson, Tim Fernando and Nicholas Asher (eds.), *Handbook of philosophy of science* Volume 14: Philosophy of linguistics. Elsevier. 33–61.

Lebeaux, David (2009) *Where does Binding Theory Apply?* Cambridge, Mass.: MIT Press.

Lees, Robert (1960) *The grammar of English nominalizations*. Bloomington: Indiana University Press.

(1976) What are transformations? In James D. McCawley (Ed.) *Syntax and Semantics 7: Notes from the Linguistic Underground*. New York: Academic Press. 27-41.

Lees, Robert & Edward Klima (1963) Rules for English Pronominalization. *Language*, 39(1). 17-28.

Leung, Tommi (2010) On the mathematical foundations of crash-proof grammars. In Michael T. Putnam (ed.) *Exploring Crash-Proof Grammars*. Amsterdam: John Benjamins. 213–244.

Levelti, Willem J. M. (1970) Hierarchical chunking in sentence processing. *Perception & Psychophysics* 8(2). 99-103. ISSN 0031-5117. doi: 10.3758/BF03210182.

Lewis, Harry and Christos Papadimitriou (1998) *Elements of the Theory of Computation* (2nd ed.), Upper Saddle River, NJ: Prentice-Hall.

Li, Ming and Paul Vitányi (1997) *An Introduction to Kolmogorov Complexity and Its Applications*. Springer.

Littlejohn, Robert (2007) The Mathematical Formalism of Quantum Mechanics. Ms. Berkeley University. Retrieved from http://bohr.physics.berkeley.edu/classes/221/0708/notes/hilbert.pdf [Accessed on 27/01/2015]

Lobina, David (2014) What linguists are talking about when talking about… *Language Sciences* 45. 56-70.

(2017) *Recursion: A Computational Investigation into the Representation and Processing of Language*. Oxford: OUP.

Lohndal, Terje (2014) *Phrase Structure and Argument Structure: A Case Study of the Syntax-Semantics Interface.* Oxford: OUP.

Lorenz, Edward (1993) *The essence of chaos*. London: University College of London.

Lyons, John (1968) *Introduction to Theoretical Linguistics*. Cambridge: CUP.

(1977) *Semantics*. Cambridge: CUP.

Mandelbrot, Benoît (1983) *The Fractal Geometry of Nature*. W.H. Freeman & Co.

(2004) *Fractals and Chaos*. *The Mandelbrot Set and Beyond*. 3 Vols. Springer Verlag.

Marantz, Alec (1995) A reader's guide to the minimalist program. In Gert Webelhuth (ed.), *Government and binding theory and the minimalist program*. Oxford: Blackwell. Pages 351–367.

(2005) Generative linguistics within the cognitive neuroscience of language. *The Linguistic Review* 22. 429-445

(2007) Phases and Words. In Sook-Hee Choe (ed.) *Phases in the Theory of Grammar*. Seoul: Dong In. 191-222.

Marr, David (1982) *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. New York: Freeman.

Martin, John (2010) *Introduction to Languages and the Theory of Computation*. McGraw Hill.

Martin, Roger and Juan Uriagereka (2008) Competence for Preferences. In Xabier Artiagoitiak and Joseba Lakarrak (eds.) *Gramatika Jaietan: Patxi Goenagaren Omenez (Grammar Festival: A Tribute to Patxi Goenaga)*. Bilbao: University of the Basque Country. 561–572.

(2013) Equi redux. In Yoichi Miyamoto, Daiko Takahashi, Hideki Maki, Masao Ochi and Koji Sugisaki, Asako Uchibori (eds.), *Deep Insights, Broad Perspectives: Essays in Honor of Mamoru Saito*. Tokyo: Kaitakusha. 292–308.

(2014) Chains in Minimalism. In Peter Kosta, Steven L. Franks, Teodora Radeva-Bork and Lilia Schürcks (eds.), *Minimalism and Beyond: Radicalizing the interfaces*. Amsterdam: John Benjamins. 169-194.

(in preparation) Fields. Ms. Yokohama National University and University of Maryland.

van der Mass, Han and Maartje Raijmakers (2009) Transitions in Cognitive Development: Prospects and Limitations of a Neural Dynamic Approach. In John Spencer et al. (eds.) *Toward a Unified Theory of Development Connectionism and Dynamic System Theory Re-Considered*. Oxford: OUP. Retrieved from Oxford Scholarship Online. DOI:10.1093/acprof:oso/9780195300598.003.0015

Mateu Fontanals, Jaume (2002) *Argument structure. Relational construal at the syntax-semantics interface*. Doctoral dissertation, Bellaterra. Available online at http://www.tesisenxarxa.net/TDX-1021103–173806/ [Accessed on 13/06/2012]

Mateu Fontanals, Jaume (2005) Impossible primitives. In Markus Werning (ed.) *The composition of meaning and content: Foundational issues*. Berlin: De Gruyter. 213–230.

Mathy, Fabien and Jacob Feldman (2012) What's magic about magic numbers? Chunking and data compression in short-term memory. *Cognition* 122. 346-362.

Matlach, Vladimir and Diego Krivochen (2016) Measuring string 'randomness': applications of quantitative methods to natural and formal languages. Presented at OLINCO, June 9th, 2016.

May, Kenneth (1962) *Elements of Modern Mathematics*. Reading, Mass. London.

May, Robert (1985) *Logical Form: Its Structure and Derivation*. Cambridge, Mass.: MIT Press.

McCarthy, John (1960) Recursive Functions of Symbolic Expressions and Their Computation by Machine, Part I. Ms. MIT. Available online at http://www-formal.stanford.edu/jmc/recursive.pdf [Accessed on 22/12/2014]

McCarthy, John J. (2010) An Introduction to Harmonic Serialism. Ms. University of Massachusetts, Amherst. Available online at: http://works.bepress.com/john_j_mccarthy/103 [Accessed on 20/12/2014]

McCarthy, John J. and Alan Prince (1995) Faithfulness and reduplicative identity. In Jill Beckman, Laura Walsh Dickey & Suzanne Urbanczyk (eds.), *Papers in Optimality Theory. University of Massachusetts Occasional Papers 18.* Amherst, Mass.: Graduate Linguistic Student Association. 249–384.

McCawley, James (1968) Lexical insertion in a transformational grammar without deep structure. *CLS* 4. 71-80.

(1971) Interpretative semantics meets Frankenstein. *Foundations of Language* 7. 285-296.

(1973) Prelexical syntax. In James McCawley, *Grammar and Meaning*. Tokyo: Taishukan. 343-356.

(1976) (ed.). *Notes from the linguistic underground* (Syntax and Semantics 7). New York: Academic Press.

(1982a) The non-existence of syntactic categories. In James D. McCawley, *Thirty Million Theories of Grammar*. Chicago: University of Chicago Press. 176-203.

(1982b) Parentheticals and Discontinuous Constituent Structure. *Linguistic Inquiry* 13(1). 91-106.

(1988) Review of Chomsky's *Knowledge of Language* (1986). *Language* 64. 355-366.

(1998) *The Syntactic Phenomena of English*. [2nd edition] Chicago: University of Chicago Press.

McClelland, James and Gautam Vallabha (2009) Connectionist Models of Development: Mechanistic Dynamical Models with Emergent Dynamical Properties. In John Spencer et al. (eds.) *Toward a Unified Theory of Development Connectionism and Dynamic System Theory Re-Considered*. Oxford: OUP. Retrieved from Oxford Scholarship Online. DOI: 10.1093/acprof:oso/9780195300598.001.0001

McGilvray, James (2013) *Chomsky: Language, Mind and Politics, 2nd Edition*. London: Wiley.

Medeiros, David (2008) Optimal Growth in Phrase Structure. *Biolinguistics* 2(2–3). 152–195.

(2012) *Economy of Command*. PhD Thesis, University of Arizona.

(2015a) Stack Sorting Syntax: Linking sound and meaning without trees or movement. Ms. University of Arizona.

(2015b) Word Order & The Structure of Memory: Queue Left, Stack-Sort Right. Presented at UCLA Syntax and Semantics workshop, November 13th, 2015.

(2016) Queue Left, Stack-sort Right: Syntactic structure without Merge. Presented at the First Symposium on the Physics of Language, Sophia University, Tokyo. March 4-5, 2016.

(2017) ULTRA: Universal Grammar as a Universal Parser. *Front. Psychol*.: *Language Sciences* 15. https://doi.org/10.3389/fpsyg.2018.00155

Mel'čuk, Igor (1988) *Dependency Syntax: Theory and Practice*. Albany: State University of New York Press.

Merchant, Jason (2001) *The Syntax of Silence*. Oxford: Oxford University Press.

(2008) Variable island repair under ellipsis. In Kyle Johnson (ed.) *Topics in Ellipsis*. Oxford: OUP. 132-153.

Michaelis, Jens (2001) Derivational Minimalism is Mildly Context-Sensitive. In M. Moortgat (ed.), *Logical Aspects of Computational Linguistics (LACL '98), Lecture Notes in Artificial Intelligence* Vol. 2014. 179-198.

Miller, George A. (1956) The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review* 63(2). 81–97.

(2003) The Cognitive Revolution: A Historical Perspective. *Trends in Cognitive Science* 7(3). 141-144.

Milner, Robin (2006) Turing, Computing and Communication. In Dina Goldin, Scott Smolka, and Peter Wegner (eds.) *Interactive Computation: The New Paradigm*. Berlin: Springer Verlag. 1-8.

Moessner, Roderich and Arthur Ramírez (2006) Geometrical Frustration. *Physics Today*, February 2006. 24-29.

Montague, Richard (1973) The proper treatment of quantification in ordinary English. In Jaakko Hintikka, Julius Moravcsik and Patrick Suppes (eds.), *Approaches to Natural*

Language (Synthese Library 49), Dordrecht: Reidel, 221–242. Reprinted in Paul Portner and Barbara Hall Partee (eds.) (2002) *Formal Semantics: the Essential Readings*. Oxford: Blackwell. 17–35.

Mootha, Vamsi (1993) Unary Fibonacci numbers are context sensitive. *Fibonacci Quarterly* 31(1). 41-43.

Mori, Nobue (2005) *A Syntactic Structure of Lexical Verbs.* PhD Thesis, University of Maryland.

Murawski, Roman (1999) *Recursive Functions and Metamathematics*. Dordrecht: Springer-Verlag.

Murphy, Elliot (2014) *Computation and Continuity in the Evolution of Grammar*. MA dissertation, University College London.

(2015) The brain dynamics of linguistic computation. *Front. Psychol*. 6: 1515 http://dx.doi.org/10.3389/fpsyg.2015.01515 [Accessed on 27/12/2015]

(2016) Evolutionary monkey oscillomics: Generating linking hypotheses from preserved brain rhythms. *Theoretical Linguistics* 42. 117-138.

Müller, Gereon (2011) *Constraints on Displacement: A Phase-Based Approach*. Amsterdam: John Benjamins.

Müller, Stefan (2016) *Grammatical theory: From transformational grammar to constraint-based approaches*. Berlin: Language Science Press.

Müller, Stefan K., Umut A. Acar, William Duff & Robert Harper (2015) Interactive Computation in an Open World. Working paper, School of Computer Science. Carnegie Mellon University. http://repository.cmu.edu/cgi/viewcontent.cgi?article=3677&context=compsci [Accessed on 11/12/2016]

Na, Younghee and Geoffrey Huck (1992) On Extracting from Asymmetrical Structures. In Diane Brentari, Gary N. Larson and Lynn A. MacLeod (eds.), *The Joy of Grammar: A festschrift in honor of James D. McCawley*. Amsterdam: John Benjamins. 251–274.

Nagumo J., Arimoto S., and Yoshizawa S. (1962) An active pulse transmission line simulating nerve axon. *Proc. IRE*. 50. 2061–2070.

Nash, John (1954) C^1-isometric imbeddings. *Annals of Mathematics* 60(3). 383–396.

Von Neumann, John (1945) First Draft of a Report on the EDVAC. Ms. Moore School of Electrical Engineering, University of Pennsylvania.

Newman, Stanley S. (1946) On the stress system of English. *Word* 2. 171–187.

Nordlinger, Rachel and Joan Bresnan (2011) Lexical Functional Grammar: Interactions between Morphology and Syntax. In Robert Borsley and Kresti Börjars (eds.) *Non Transformational Syntax: Formal and Explicit Models of Grammar*. London: Blackwell. 112-140.

Nordström, Jackie (2013) Language as a discrete combinatorial system, rather than a recursive-embedding one. *The Linguistic Review* 31(1). 151-191.

Noyer, Rolf (1999) Distributed Morphology: Frequently Asked Questions. www.ling.upenn.edu/~rnoyer/dm/ [Accessed on 10/02/2010]

Oberauer, Klaus and Reinhold Kliegel (2006) A formal model of capacity limits in working memory. *Journal of Memory and Language* 55. 601–626.

Obrębski, Tomasz & Filip Graliński (2004) Some notes on generative capacity of dependency grammars. In *Proceedings of COLING 2004 Workshop Recent Advances in Dependency Grammar*, Geneve. 65–71

Ohta Shinri, Naoki Fukui, and Kuniyoshi Sakai (2013) Syntactic Computation in the Human Brain: The Degree of Merger as a Key Factor. *PLoS ONE* 8(2): e56230. doi:10.1371/journal.pone.0056230

Ojeda, Almerindo (1987) Discontinuity, Multidominance and Unbounded Dependency in Generalized Phrase Structure Grammar: Some Preliminaries. In Geoffrey Huck & Almerindo Ojeda (eds.) *Syntax and Semantics 20: Discontinuous Constituency*. New York: Academic Press. 257-282.

Ormazábal, Javier, Juan Uriagereka and Myriam Uribe-Etxebarria (1994) Word order and wh-movement: towards a parametric account. Ms., U.Conn./UPV-EHU/UMD/MIT.

O'Rourke, Polly, and Gregory Colflesh (2014) Conflict resolution ability underpins the processing of temporary syntactic ambiguity: Evidence from ERPs. *Online Proceedings of the 2014 meeting of the Cognitive Science Society*. Available online at https://mindmodeling.org/cogsci2014/papers/359/paper359.pdf [Retrieved on 6/1/2015]

Osborne, Tim, Michael Putnam, and Thomas Groβ (2011) Bare Phrase Structure, Label-less Trees, and Specifier-less Syntax: Is Minimalism Becoming a Dependency Grammar? *The Linguistic Review* 28(3). 315-364.

(2012) Catenae: Introducing a novel unit of syntactic analysis. *Syntax* 15(4). 354-396.

Osterhout, Lee, Phillip Holcomb, and David Swinney (1994) Brain Potentials Elicited by Garden-Path Sentences: Evidence of the Application of Verb Information During Parsing. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 20(4). 786-803.

Ouali, Hamid (2010) Computation Efficiency and Feature Inheritance in Crash Proof Syntax. In Michael T. Putnam (ed.) *Exploring Crash-Proof Grammars*. Amsterdam: John Benjamins. 15-30.

Palmer, Frank (2001) *Mood and Modality*. Cambridge, CUP. [2nd edition]

Panagiotidis, Phoevos (2011) Functional heads, Agree, and Labels. Ms. University of Cyprus. [Partially printed as Chapter 5 of Panagiotidis (2014)]

(2014) *Categorial Features: a generative theory of word class categories*. Cambridge: CUP.

Papo, David, Javier M. Buldú, Stefano Boccaletti, and Edward T. Bullmore (2014) Complex network theory and the brain. *Phil. Trans. R. Soc.* B 369 (1653). DOI: 10.1098/rstb.2013.0520.

Partee, Barbara H. (1975) Some transformational extensions of Montague grammar. In D. Hockney, W. Harper & B. Freed (eds.) *Contemporary Research in Philosophical Logic and Linguistic Semantics: Proceedings of a Conference Held at the University of Western Ontario, London, Canada*. Dordrecht: Reidel. 195-220.

(2014) A Brief History of the Syntax-Semantics Interface in Western Formal Linguistics. *Semantics-Syntax Interface* 1(1). 1–21.

Patel, Ebrahim, Peter Grindrod, Douglas Saddy, and Andrew Irving (2015) L-systems: Classification and Equivalence. Ms. University of Oxford, University of Reading, and University of Manchester.

Pentus, Mati (1993) Lambek grammars are context-free. In *Proceedings of the IEEE Symposium on Logic in Computer Science*. Montreal. 429–433.

Perlmutter, David (1978) Impersonal Passives and the Unaccusative Hypothesis. *Proceedings of the Annual Meeting of the Berkeley Linguistics Society 38*. 157-189.

Pesetsky, David and Esther Torrego (2007) The syntax of valuation and the interpretability of features. In Simin Karimi, Vida Samiian, and Wendy Wilkins (eds.) *Phrasal and clausal architecture*. Amsterdam: John Benjamins. 262–294.

Petersson, Karl and Peter Hagoort (2012) The neurobiology of syntax: beyond string sets. *Phil. Trans. R. Soc. B*, **367**. DOI: 10.1098/rstb.2012.0101

Pfleegor, Robert L. and Leonard Mandel (1967) Interference of Independent Photon Beams. *Physical Review Letters* 159(5). 1084-1088. DOI: 10.1103/PhysRev.159.1084

Piattelli-Palmarini, Massimo & David Medeiros (2018) The Golden phrase: steps to the physics of language. In Angel Gallego & Roger Martin (eds.) *Language, Syntax, and the Natural Sciences*. Cambridge: CUP.

Piattelli-Palmarini, Massimo & Juan Uriagereka (2008) Still a bridge too far? Biolinguistic questions for grounding language on brains. *Physics of Life Reviews* 5. 207–224

Piattelli-Palmarini, Massimo and Giuseppe Vitiello (2015) Linguistics and some aspects of its underlying dynamics. *Physics of Life Reviews*. arXiv:1506.08663.

(2016) Third factors in language design: some suggestions from Quantum Field Theory. In James McGilvray (ed.) *Cambridge Companion to Chomsky* (2nd Edition). Cambridge: Cambridge University Press.

(2017) Quantum field theory and the linguistic Minimalist Program: a remarkable isomorphism. *IOP Conf. Series: Journal of Physics: Conf. Series* **880** (2017) 012016. DOI:10.1088/1742-6596/880/1/012016

Piccinini, Gualtiero (2011) The Physical Church–Turing Thesis: Modest or Bold? *The British Journal for the Philosophy of Science* 62(4). 733-769. Retrieved from http://www.jstor.org/stable/41410117

Pitt, David (2013) Mental Representation. In *The Stanford Encyclopedia of Philosophy* (Fall 2013 Edition), Edward N. Zalta (ed.), URL:

http://plato.stanford.edu/archives/fall2013/entries/mental-representation/ [Accessed on 1/12/2014]

Poeppel, David and David Embick (2005) Defining the relation between linguistics and neuroscience. In Anne Cutler (ed.) *Twenty-first century psycholinguistics: Four cornerstones*. London: Lawrence Erlbaum. 103-119.

Pollard, Carl and Ivan Sag (1994) *Head-Driven Phrase Structure Grammar*. University of Chicago Press.

Pomerantz, James R. and Michael Kubovy (1986) Theoretical approaches to perceptual organization: Simplicity and likelihood principles. In Kenneth Boff, Lloyd Kaufman, and James P. Thomas (eds.) *Handbook of perception and human performance, Vol. 2: Cognitive processes and performance*. Oxford: John Wiley & Sons. 1-46.

Post, Emil (1943) Formal Reductions of the General Combinatorial Decision Problem. *American Journal of Mathematics*, 65(2). 197-215.

(1944) Recursively enumerable sets of positive integers and their decision problems. *Bulletin of the American Mathematical Society* 50. 284–316.

(1948) Degrees of recursive unsolvability. Preliminary report. *Bulletin of the American Mathematical Society* 54. 641–664.

Postal, Paul M. (1964) *Constituent Structure*. Bloomington, Indiana: University of Bloomington.

(1969) On so-called 'pronouns' in English. In David Reidel & Sanford Schane (eds.) *Modern Studies in English: Readings in Transformational Grammar*. New Jersey: Prentice Hall. 201-224.

(1972) On Some Rules That Are Not Successive Cyclic. *Linguistic Inquiry* 3(2). 211-222.

(1974) *On Rising*. Cambridge, Mass.: MIT Press.

(1997) *Three Investigations on Extraction*. Cambridge, Mass.: MIT Press.

(2003) (Virtually) Conceptually Necessary. *Journal of Linguistics* 39. 599-620. DOI: 10.1017/S0022226703002111

(2004) *Skeptical Linguistic Essays*. Oxford: OUP.

(2009) The incoherence of Chomsky's biolinguistic ontology. *Biolinguistics* 3. 104-123.

Prangle, Dennis (2014) Lazy ABC. *Statistics and Computing*. http://dx.doi.org/10.1007

Prusinkiewicz, Przemysław and James Hanan (1989) *Lindenmayer Systems, Fractals, and Plants* (Lecture Notes in Biomathematics). Springer-Verlag.

Prusinkiewicz, Przemyslaw and Aristid Lindenmayer (2010) *The Algorithmic Beauty of Plants*. Springer-Verlag. [2nd edition]

Pullum, Geoffrey K. (2007) The evolution of model-theoretic frameworks in linguistics. In James Rogers and Stephan Kepser (eds.), *Model-Theoretic Syntax at 10*. Dublin: Trinity College Dublin. 1–10.

Pullum, Geoffrey K. and Barbara C. Scholz. (2001) On the distinction between model-theoretic and generative-enumerative syntactic frameworks. In Philippe de Groote, Glyn Morrill, and Christian Retoré (eds.) *Logical Aspects of Computational Linguistics: 4th International Conference* (Lecture Notes in Artificial Intelligence, 2099). Berlin: Springer Verlag. 17–43.

Putnam, Michael (2010) Exploring Crash-Proof Grammars: An introduction. In Michael T. Putnam (ed.) *Exploring Crash-Proof Grammars*. Amsterdam: John Benjamins. 1-14.

(2017) Feature reassembly as constraint satisfaction. *The Linguistic Review*. DOI: https://doi.org/10.1515/tlr-2017-0010

Putnam, Michael T. & Thomas S. Stroik (2016) Is Simplest Merge too simple?: A reply to Epstein, Kitahara & Seely (2014). Ms.

Pylkkänen, Lina (2008) *Introducing arguments*. Cambridge, Mass.: MIT Press.

Pylyshyn, Zenon (1999) Is vision continuous with cognition? The case for cognitive impenetrability of visual perception. *Behavioral and Brain Sciences* 22(3). 341-423.

(2007) *Things and Places. How the Mind Connects with the World*. Cambridge, Mass.: MIT Press.

Rabinovich, Mikhail I., Pablo Varona, Irma Tristan, and Valentin Afraimovich (2014). Chunking dynamics: heteroclinics in mind. *Frontiers in Computational Neuroscience* 8(22). http://doi.org/10.3389/fncom.2014.00022

Ramchand, Gillian (2008) *Verb Meaning and the Lexicon*. Cambridge University Press.

Ramírez Fernández, Javier (2015) Locality in Language and Locality in Brain Oscillatory Structures. *Biolinguistics* 9. 74-95

Rammal, R, G. Toulouse, and M. A. Virasoro (1986) Ultrametricity for Physicists. *Reviews of Modern Physics* 58(3). 765-788. DOI: 10.1103/RevModPhys.58.765

Reinhart, Tanya (1976) *The Syntactic Domain of Anaphora*. PhD Dissertation, MIT.

(1983) *Anaphora and Semantic Interpretation*. London: Croom Helm.

(2006) *Interface Strategies: Optimal and Costly Computations*. Cambridge, Mass.: MIT Press.

Reuland, Eric (2011) *Anaphora and Language Design*. Cambridge, Mass.: MIT Press.

Ringen, Catherine (1972) On Arguments for Rule Ordering. *Foundations of Language* 8(2). 266-273.

Ristad, Eric and Robert Berwick (1989) Computational Consequences of Agreement and Ambiguity in Natural Language. *Journal of Mathematical Psychology* 33(4). 379-396.

Rizzi, Luigi (1990) *Relativized Minimality*. Cambridge, Mass.: MIT Press.

(2007) On Some Properties of Criterial Freezing. *CISCL Working Papers on Language and Cognition* 1. 145–158.

Roberson, Debi, Ian Davies, and Jules Davidoff (2000) Color categories are not universal: Replications and new evidence from a stone-age culture. *Journal of Experimental Psychology: General*, 129(3). 369-398. http://dx.doi.org/10.1037/0096-3445.129.3.369

Roberts, Mark (2015) Ultrametric Distance in Syntax. *Prague Bulletin of Mathematical Linguistics* 103. 111-130.

Rogers, Andy (1974) A Transderivational Constraint on Richard? *Papers from the Tenth Regional Meeting of the Chicago Linguistic Society*. 551-558.

Rojas Berscia, Luis Miguel (2012) Desmitificando la categoría de "co-verbo" en chino mandarín: el caso de "zài". *Sorda y Sonora* 2(1-2). 77-100.

Rooryck, Johan (1992) Negative and Factive Islands Revisited. *Journal of Linguistics* 28(2). 343-374.

Rooryck, Johan and Guido V. Wyngaerd (2011) *Dissolving Binding Theory*. Oxford: OUP.

Rosemberg, Rebecca and Lisa Feigenson (2013) Infants hierarchically organize memory representations. *Developmental Science* 16(4). 610–621. DOI: 10.1111/desc.12055

Ross, John Robert (1967) *Constraints on Variables in Syntax*. PhD Dissertation, MIT.

(1969a) Guess Who? In Robert I. Binnick, Alice Davison, Georgia M. Green, and Jerry L. Morgan (eds.) *Papers from the Fifth Regional Meeting of the Chicago Linguistic Society*. Chicago: University of Chicago. 252-286.

(1969b) On the Cyclic nature of English Pronominalization. In David Reidel & Sanford Schane (eds.) *Modern Studies in English: Readings in Transformational Grammar*. New Jersey: Prentice Hall. 187-200.

(1970) Gapping and the order of Constituents. En *Actes du X^{e} Congress International des Linguistes*. Bucarest. 841-853.

Rothstein, Susan (2004) *Structuring Events: A Study in the Semantics of Lexical Aspect*. London: Blackwell.

Rozenberg, Gzegorz and Arto Salomaa (1980) *The Mathematical Theory of L-Systems*. New York: Academic Press.

Rumelhart, David and James McClelland (eds.) (1986) *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 1: Foundations*. Cambridge, MA: MIT Press.

Russell, Bertrand (1911) Knowledge by Acquaintance and Knowledge by Description. *Proceedings of the Aristotelian Society* 11. 108–128.

Saddy, Douglas (1990) *Investigations into Grammatical Knowledge*. PhD Thesis, MIT.

(2009) Perceiving and Processing Recursion in Formal Grammars. Presented in *Recursion: Structural Complexity in Language and Cognition Conference*. UMass.: Amherst.

(2010) What's next? Talk presented at Aarhaus University, 2010. [Available online at http://www.hum.au.dk/engelsk/engsv/nyvad-abstracts/saddy-ho.pdf Accessed on 05/06/2016]

(2018) Syntax and Uncertainty. In Angel Gallego & Roger Martin (eds.) *Language, Syntax, and the Natural Sciences*. Cambridge: CUP.

Saddy, Douglas and Peter Grindrod (2016) Irreducible Subnets and Random Delays: Limiting the Vocabulary of Neural Information. Presented at First International Symposium on the Physics of Language. Sophia University, Tokyo. March 4-5, 2016.

Saddy, Douglas and Diego Krivochen (2016a) Emergent Oscillations and Cyclicity: Physical Aspects of Frustrated Cognitive Systems. Presented at First International Symposium on the Physics of Language. Sophia University, Tokyo. March 4-5, 2016.

(2016b) Prospects for a parametrization of the L-Space. Ms. University of Reading. https://www.researchgate.net/publication/320254201_Prospects_for_a_parametrization_of_the_L-space

Saddy, Douglas and Juan Uriagereka (2004) Measuring Language. *International Journal of Bifurcation and Chaos* 14(2). 383-404.

Sag, Ivan (2010) English Filler-Gap constructions. *Language* 86(3). 486-545.

Sampson, Geoffrey (2009) A linguistic axiom challenged. In Geoffrey Sampson, David Gil and Peter Trudgill (eds.), *Language Complexity as an Evolving Variable*. Oxford: OUP. 1-18.

Sandamirskaya, Yulia (2014) Dynamic neural fields as a step toward cognitive neuromorphic architectures. *Frontiers in Neuroscience*. doi: 10.3389/fnins.2013.00276

Sandamirskaya, Yulia, Stephan K. U. Zibner, Sebastian Schneegans, and Gregor Schöner (2013) Using Dynamic Field Theory to extend the embodiment stance toward higher cognition. *New Ideas in Psychology* 31(3). 322–339

Santos, Eugene (1969) Probabilistic Turing Machines and Computability. *Proceedings of the American Mathematical Society* 22(3). 704-710.

Schmerling, Susan (1972) Apparent counterexamples to the Coordinate Structure Constraint: A canonical conspiracy. *Studies in the Linguistic Sciences* 2(1). 91-104.

(1975) Asymmetric Conjunction and rules of Conversation. In Peter Cole and Jerry Morgan (eds.) *Syntax and Semantics,Vol. 3: Speech Acts*. New York: Academic Press. 211-231.

(1982) How imperatives are special, and how they aren't. In Robinson Schneider, Kevin Tuite, and Robert Chametzky (eds.) *Papers from the parasession on nondeclaratives*. Chicago: Chicago Linguistic Society. 202–218.

(1983) Two theories of syntactic categories. *Linguistics and Philosophy* 6. 393–421.

(2017) On rhetorical meaning. Ms.

(in press) *Sound and Grammar: Towards a Neo-Sapirian Theory of Language*. To appear in Brill.

Schneegans, Sebastian (2014) Operations in Multi-Dimensional Neural Fields. Presented at KogWis 2014, University of Tübingen. http://www.uni-tuebingen.de/index.php?eID=tx_nawsecuredl&u=0&g=0&t=1422113823&hash=488dd323d865d3039c9f2a325a34de6c02e3cdae&file=fileadmin/Uni_Tuebingen/Fakultaeten/InfoKogni/WSI/Zentren/CogSciCenter/tutorialDynamicFieldTheory_Schneegans_NDFields.pdf [Retrieved on 16/01/2015]

Schöner, Gregor (2009) Development as Change of System Dynamics: Stability, Instability, and Emergence. In John Spencer et al. (eds.) *Toward a Unified Theory of Development Connectionism and Dynamic System Theory Re-Considered*. Oxford: OUP. Retrieved from Oxford Scholarship Online. DOI:10.1093/acprof:oso/9780195300598.003.0002.

Schöner, Gregor and Scott Kelso (1988) Dynamic pattern generation in behavioral and neural systems. *Science* 239. 1513–1520.

Schöner, Gregor and Esther Thelen (2006) Using dynamic field theory to rethink infant habituation. *Psychology Review* 113(2). 273–299.

Schwartz, Matthew (2014) *Quantum Field Theory and the Standard Model*. Cambridge: CUP.

Shibatani, Masayoshi (1976) The Grammar of Causative Constructions: A Conspectus. In Masayoshi Shibatani (ed.) *The Grammar of Causative Constructions*. New York: Academic Press. 1-40.

Shieber, Stuart (1986) *An Introduction to Unification-Based Approaches to Grammar*. Brookline, Mass.: Microtome Publishing.

Shirley, Elisabeth (2014) *Representing and remembering Lindenmayer-grammars*. PhD Thesis, University of Reading.

Shishikura, Mitsuhiro (1991) The Hausdorff dimension of the boundary of the Mandelbrot and Julia sets. *Stony Brook IMS* 1991/7. arXiv:math/9201282

Siegelmann, Hava (1995) Computation Beyond the Turing Limit. *Science* 268. 545-548.

Simon, Herbert (1984) *The Sciences of the Artificial*. Cambridge, Mass.: MIT Press.

Slater, Alan and Rachel Kirby (1998) Innate and learned perceptual abilities in the newborn infant. *Exp Brain Res* (123). 90–94.

Sloat, Clarence (1969) Proper nouns in English. *Language* 45(1). 26–30.

Smith, Alvy Ray (1984) Plants, Fractals, and Formal Languages. *Computer Graphics* 18(3). 1-10.

Smolensky, Paul and Geraldine Legendre (2006) *The Harmonic Mind*. Cambridge, Mass.: MIT Press.

Smolensky, Paul, Matthew Goldrick & Donald Mathis (2014) Optimization and Quantization in Gradient Symbol Systems: A Framework for Integrating the Continuous and the Discrete in Cognition. *Cognitive Science* 38. 1102–1138.

Smolensky, Paul and Matthew Goldrick (2016) Gradient Symbolic Representations in Grammar: The case of French Liaison. Ms. Rutgers Optimality Archive. Available online at http://roa.rutgers.edu/content/article/files/1552_smolensky_1.pdf [Retrieved on 25/07/2016]

Spencer, John and Gregor Schöner (2003). Bridging the representational gap in the dynamical systems approach to development. *Developmental Science* 6(4). 392–412.

Spencer, John, Sammy Perone, and Jeffrey Johnson (2009) Dynamic Field Theory and Embodied Cognitive Dynamics. In John Spencer et al. (eds.) *Toward a Unified Theory of Development Connectionism and Dynamic System Theory Re-Considered*. Oxford: OUP. Retrieved from Oxford Scholarship Online. DOI:10.1093/acprof:oso/9780195300598.003.0005

Sperber, Dan and Deirdre Wilson (1986) On defining relevance. In Richard Grandy & Richard Warner (eds.) *Philosophical Grounds of Rationality: Intentions, Categories, Ends*. Oxford: OUP. 143-158.

(1995) *Relevance: Communication and Cognition*. Second Edition, Oxford/Cambridge: Blackwell.

Spivey, Michael (2007) *The Continuity of Mind*. Oxford: OUP.

Stabler, Edward (2003) Lectures in Computational Linguistics. Available online at http://www.linguistics.ucla.edu/people/stabler/185-08.pdf [Retrieved on 24/11/2014]

Stein, D. and C. Newman (2011) Spin Glasses: Old and New Complexity. *Complex Systems* 20. 115-126.

Steedman, Mark and Jason Baldridge (2011) Combinatory Categorial Grammar. In Robert Borsley & Kresti Börjars (eds.), *Non-transformational syntax. Formal and explicit models of grammar*. London: Blackwell. 181-224.

Stowell, Tim (1980) *Origins of phrase structure*. PhD Thesis, MIT.

Strogatz, Steven (1994) *Nonlinear Dynamics and Chaos*. Reading, Mass.: Perseus Books.

Stroik, Thomas (2009) *Locality in Minimalist Syntax*. Cambridge, Mass.: MIT Press.

Stroik, Thomas and Michael T. Putnam (2013) *The Structural Design of Language*. Cambridge: CUP.

Syropoulos, Apostolos (2008) *Hypercomputation: Computing beyond the Church-Turing barrier*. Springer Verlag.

Talmy, Leonard (2000) *Towards a Cognitive Semantics* (2 Vols.). Cambridge, Mass.: MIT Press.

Taylor, Alex, Douglas Elliffe, Gavin R. Hunt, and Russell D. Gray (2010) Complex cognition and behavioural innovation in New Caledonian crows. *Proceedings of the Royal Society B: Biological Sciences* 277 (1694). 2637-2643

Taylor, Alex and Nicola Clayton (2012) Evidence from convergent evolution and causal reasoning suggests that conclusions on human uniqueness may be premature. *Behavioral and Brain Sciences* 35 (4). 241-242.

Taylor, Kristen, Helen Moss, and Lorraine Tyler (2007) The Conceptual Structure Account: A cognitive model of semantic memory and its neural instantiation. In John Hart & Michael Kraut (Eds.). *The Neural Basis of Semantic Memory*. Cambridge: Cambridge University Press. 265-301.

Tegmark, Max (2007) The Mathematical Universe. arXiv:0704.0646v2. DOI: 10.1007/s10701-007-9186-9. [Retrieved on 8/1/2015]

Tesnière, Lucien (1959) *Éléments de syntaxe structural*. Paris: Klincksieck.

Thomas, Michael (1997) *Connectionist Networks and Knowledge Representation: The Case of Bilingual Lexical Processing.* PhD Thesis, University of Oxford.

Thompson, Evan (2007) *Mind and Life*. Cambridge, Mass.: Harvard University Press.

Ticio, M. Emma (2010) *Locality Domains in the Spanish Determiner Phrase*. Studies in Natural Language and Linguistic Theory 79, Springer.

Tognoli, Emmanuelle & Scott Kelso (2014) The Metastable Brain. *Neuron* 81(1). 35-48.

Townsend, David J., and Thomas G. Bever (2001) *Sentence Comprehension: The Integration of Habits and Rules*. Cambridge, Mass.: MIT Press.

Turing, Alan (1936) On Computable Numbers, with an Application to the Entscheidungsproblem, *Proc. London Math. Soc*. 42(2). 230-265.

(1952) The chemical basis of morphogenesis. *Transactions of the Royal Society of London B* 237. 37-72.

Tzambazakis, Antonios (2015) The evolution of the biological field concept. In D. Fels, M. Cifra and F. Scholkmann (Eds.) *Fields of the Cell*. Research Signpost. 1–27.

Unger, Ron and John Moult (1993) Finding the lowest free energy conformation of a protein is an NP-hard problem: Proof and implications. *Bulletin of Mathematical Biology* 55(6). 1183-1198

Uriagereka, Juan (1998) *Rhyme and Reason*. Cambridge, Mass.: MIT Press.

(2002a) Multiple Spell-Out. In Juan Uriagereka, *Derivations: Exploring the Dynamics of Syntax.* London: Routledge. 45-65.

(2002b) Warps: Some Thoughts on Categorization. In Juan Uriagereka, *Derivations: Exploring the Dynamics of Syntax*. London: Routledge. 288-317.

(2002c) Formal and Substantive Elegance in the Minimalist Program (On the Emergence of Some Linguistic Forms). In Juan Uriagereka, *Derivations: Exploring the Dynamics of Syntax*. London: Routledge. 147-175.

(2008) *Syntactic Anchors: On Semantic restructuring*. Cambridge: CUP.

(2011) A Sketch of the Grammar in Non-Classical Conditions. Ms. UMD.

(2012) *Spell-Out and the Minimalist Program*. Oxford: OUP.

(2014) Regarding the Third Factor: Arguments for a CLASH model. In Peter Kosta, Steven L. Franks, Teodora Radeva-Bork and Lilia Schürcks (eds.), *Minimalism and Beyond: Radicalizing the interfaces*. Amsterdam: John Benjamins. 363–391.

(2015) “Natural” grammars and natural language. In Beatriz Fernández and Pello Salaburu (eds.) *Ibon Sarasola, Gorazarre*. Bilbao: Editorial de la Universidad del País Vasco. 665-674.

(2016) Factoring out language design. Presented at *7th International Conference in Formal Linguistics*. Dec. 4th, Nankai University, Tianjin, China.

Uriagereka, Juan and Norbert Hornstein (2002) Labels and Projections. In Juan Uriagereka, *Derivations: Exploring the Dynamics of Syntax*. London: Routledge. 115-135.

Uriagereka, Juan and Paul Pietroski (2002) Dimensions of natural language. In Juan Uriagereka, *Derivations: Exploring the Dynamics of Syntax*. London: Routledge. 266-287.

Varela, Francisco (1979) *Principles of Biological Autonomy*. North-Holland.

Varela, Francisco, Evan Thompson, and Eleanor Rosch (1991) *The Embodied Mind: Cognitive Science and Human Experience*. Cambridge, MA: MIT Press.

Varela, Francisco, Jean-Philippe Lachaux, Eugenio Rodriguez & Jacques Martinerie (2001) The brainweb: Phase synchronization and largescale integration. *Nat Rev Neurosci*. 2(4). 229–239.

te Velde, John R. (2005) *Deriving Coordinate Symmetries: A phase-based approach integrating Select, Merge, Copy, and Match.* Amsterdam: John Benjamins.

Vendler, Zeno (1957) Verbs and Times. *The Philosophical Review* 66(2). 143-160.

De Vries, Gertrud (1992) *On Coordination and Ellipsis*. PhD Dissertation, Catholic University of Brabant [Tilburg University].

De Vries, Mark (2009) On Multidominance and Linearization. *Biolinguistics,* 3(4). 344–403.

(2017) Across-the-Board Phenomena. In Martin Everaert and Henk van Riemsdijk (eds.) *Blackwell Companion to Syntax*, 2nd edition. DOI: 10.1002/9781118358733.wbsyncom094

Youschkevitch, Adolf (1976/1977) The concept of function up to the middle of the 19th century. *Archive for History of Exact Sciences* 16(1). 37-85.

Warner, Seth (1989) *Topological Fields*. Amsterdam: North Holland.

Wasow, Thomas (1979) *Anaphora in Generative Grammar*. Gent: E. Story-Scientia.

Watumull, Jeffrey (2012) A Turing Program for Linguistic Theory. *Biolinguistics* 6(2). 222–245.

Watumull, Jeffrey, Marc Hauser, Ian Roberts, and Norbert Hornstein (2014) On Recursion. *Frontiers in Psychology: Language Sciences* 4. 1017.

Wegner, Peter (1997) Why Interaction is More Powerful than Algorithms. *Comm. ACM*. 40(5).

(1998) Interactive Foundations of Computing. *Theoretical Computer Science* 192(2). 315-351.

Wells, Rulon S. (1947) Immediate Constituents. *Language* 23. 81-117.

Wexler, Kenneth, Peter Culicover, and Henry Hamburger (1975) Learning-theoretic foundations of linguistic universals. *Theoretical Linguistics* 2. 213-253.

Willard, Stephen (2004) *General Topology*. New York: Dover.

Williams, Edwin (1978) Across-the-Board Rule Application. *Linguistic Inquiry* 9(1). 31-43.

(2003) *Representation Theory*. Cambridge, Mass.: MIT Press.

Wilson, Deirdre and Dan Sperber (2004) Relevance Theory. In Lawrence Horn and Gregory Ward, (eds.) *The Handbook of Pragmatics*. Oxford: Blackwell. 607-632.

Wilson, Robert A. and Lucia Foglia (2011) Embodied Cognition. *The Stanford Encyclopedia of Philosophy* (Fall 2011 Edition), Edward N. Zalta (ed.), URL: http://plato.stanford.edu/archives/fall2011/entries/embodied-cognition/ [Retrieved on 23/11/2014]

Wunderlich, Dieter (1997) 'Cause' and the structure of verbs. *Linguistic Inquiry* 28(1). 27-68.

Wurmbrand, Susi (2014) The Merge Condition: A syntactic approach to selection. In Peter Kosta, Steven L. Franks, Teodora Radeva-Bork and Lilia Schürcks (eds.) *Minimalism and Beyond: Radicalizing the interfaces*. Amsterdam: John Benjamins. 130-167.

Xiang, Jing, Sylvester Chuanga, Daniel Wilsonb, Hiroshi Otsuboc, Elizabeth Pangc, Stephanie Holowkaa, Rohit Sharmac, Ayako Ochic, and Shiro Chitokuc (2002) Sound motion evoked magnetic fields. *Clinical Neurophysiology* 113. 1–9.

Yang, Charles (2000) *Knowledge and Learning in Natural Language*. Ph.D. thesis, MIT.

Yli-Jyrä, Anssi, Andras Kornai & Jacques Sakarovitch (2011) Finite-state methods and models in natural language processing. *Natural Language Engineering*, 17(2), 141-144. doi:10.1017/S1351324911000015

Zachos, Cosmas and Thomas Curtright (1999) Phase-Space Quantization of Field Theory. *Progress of Theoretical Physics Supplements* 135. 244-258.

Zamparelli, Roberto (2011) Coordination. In Klaus von Heusinger, Claudia Maienborn, and Paul Portner (eds.), *Semantics: an international handbook of natural language meaning*. Berlin: De Gruyter. 1713-1741.

Zeeman, Christopher (1977) *Catastrophe theory: selected papers 1972-1977*. Cambridge, Mass.: Addison-Wesley.

Zeijlstra, Hedde (2012) There is only one way to Agree. *The Linguistic Review* 29(3). 491–539.

Zipf, George (1949) *Human Behavior and the Principle of Least Effort.* Cambridge, Mass.: Addison-Wesley.

Zwart, Jan-Wouter (2009) Prospects for top-down derivations. *Catalan Journal of Linguistics* 9. 161-187.

Zwarts, Joost (2003) Vectors across Spatial Domains: From Place to Size, Orientation, Shape, and Parts. In Emile van der Zee and Jon Slack (eds.) *Representing Direction in Language and Space*. Oxford: OUP. DOI:10.1093/acprof:oso/9780199260195.003.0003

Zwarts, Joost and Peter Gärdenfors (2015) Locative and Directional Prepositions in Conceptual Spaces: The Role of Polar Convexity. *Journal of Logic, Language, and Information*. DOI 10.1007/s10849-015-9224-5