Convergent least-squares optimisation methods for variational data assimilation

Cartis, C., Kaouri, M. H., Lawless, A. S. ORCID: https://orcid.org/0000-0002-3016-6568 and Nichols, N. K. ORCID: https://orcid.org/0000-0003-1133-5220 (2024) Convergent least-squares optimisation methods for variational data assimilation. Optimization, 73 (11). pp. 3451-3485. ISSN 1029-4945 doi: 10.1080/02331934.2024.2390119 Available at https://centaur.reading.ac.uk/117515/

Optimization
A Journal of Mathematical Programming and Operations Research

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/gopt20

Convergent least-squares optimization methods for variational data assimilation

Coralia Cartis, Maha H. Kaouri, Amos S. Lawless & Nancy K. Nichols

To cite this article: Coralia Cartis, Maha H. Kaouri, Amos S. Lawless & Nancy K. Nichols (2024) Convergent least-squares optimization methods for variational data assimilation, Optimization, 73:11, 3451-3485, DOI: 10.1080/02331934.2024.2390119

To link to this article: https://doi.org/10.1080/02331934.2024.2390119

Published online: 19 Aug 2024.

OPTIMIZATION
2024, VOL. 73, NO. 11, 3451–3485
https://doi.org/10.1080/02331934.2024.2390119

Taylor & Francis
Taylor & Francis Group

Convergent least-squares optimization methods for variational data assimilation

Coralia Cartis[a*], Maha H. Kaouri[b*], Amos S. Lawless[b,c,d*] and Nancy K. Nichols[b,c,d*]

[a]Mathematical Institute, University of Oxford, Oxford, UK; [b]Department of Mathematics and Statistics, University of Reading, Reading, UK; [c]Department of Meteorology, University of Reading, Reading, UK; [d]National Centre for Earth Observation, Leicester, UK

ABSTRACT

Data assimilation combines prior (or background) information with observations to estimate the initial state of a dynamical system over a given time-window. A common application is in numerical weather prediction where a previous forecast and atmospheric observations are used to obtain the initial conditions for a numerical weather forecast. In four-dimensional variational data assimilation (4D-Var), the problem is formulated as a nonlinear least-squares problem, usually solved using a variant of the classical Gauss-Newton (GN) method. However, we show that GN may not converge if poorly initialized. In particular, we show that this may occur when there is greater uncertainty in the background information compared to the observations, or when a long time-window is used in 4D-Var allowing more observations. The difficulties GN encounters may lead to inaccurate initial state conditions for subsequent forecasts. To overcome this, we apply two convergent GN variants (line search and regularization) to the long time-window 4D-Var problem and investigate the cases where they locate a more accurate estimate compared to GN within a given budget of computational time and cost. We show that these methods are able to improve the estimate of the initial state, which may lead to a more accurate forecast.

Highlights

- Poor initialization of Gauss-Newton method may result in failure to converge.
- Safeguarded Gauss-Newton improves initial state estimate within limited time/cost.
- Results using twin experiments with long time-window and chaotic Lorenz models.

ARTICLE HISTORY
Received 23 May 2023
Accepted 27 July 2024

KEYWORDS
Data assimilation; Gauss-Newton method; least-squares problems; optimization algorithms

2020 MATHEMATICS SUBJECT CLASSIFICATION
90C30

CONTACT Coralia Cartis cartis@maths.ox.ac.uk, Coralia.Cartis@maths.ox.ac.uk

*The order of the authors is alphabetical; the second author (M.H.Kaouri) is the lead contributor.

- Apply state of the art least-squares convergence theory to data assimilation.
- Improvements to initial state estimate may lead to a more accurate forecast.

1. Introduction

Four-dimensional variational data assimilation (4D-Var) aims to solve a nonlinear least-squares problem that minimizes the error in a prior estimate of the initial state of a dynamical system together with the errors between observations and numerical model estimates of the states of the system over time. In Numerical Weather Prediction (NWP), 4D-Var is used to estimate the initial conditions for a weather forecast [1]. The 4D-Var scheme is able to incorporate information from a previous forecast along with observations over both temporal and spatial domains, weighted by the uncertainties in the prior and the observations. From a Bayesian point of view the solution is the *maximum a posteriori* estimate of the initial state [2]. The nonlinear least-squares objective function is minimized using an iterative method. The quality of the estimate and the subsequent forecast depends on how accurately the 4D-Var problem is solved within the time and computational cost available.

In this paper, we investigate the application of *globally convergent* optimization methods to the 4D-Var problem; such methods use safeguards to guarantee convergence from an arbitrary initial estimate by ensuring a sufficient, monotonic/strict decrease in the objective function at each iteration. We focus on the strong-constraint 4D-Var problem where we assume that the numerical model of the system perfectly represents the true dynamics of the system or the model errors are small enough to be neglected. This results in the formulation of variational data assimilation as an unconstrained nonlinear least-squares problem and is employed by many operational meteorological centres [3], including the Meteorological Service of Canada [4], the European Centre for Medium-range Weather Forecasts (ECMWF) [5,6] and the Met Office [7].

Ideally, in large-scale unconstrained optimization, we seek a fast rate of convergence, which can be achieved in nondegenerate cases using a Newton-type method. However, these methods require the use of second order derivatives of the objective function, which are too costly to compute and store operationally. Therefore, optimization methods that approximate the high order terms, such as limited memory Quasi-Newton [8–11], Inexact Newton [12], Truncated Newton [13,14], Adjoint Newton [15], Hessian-free Newton [16], Gauss-Newton [17,18] and Approximate Gauss-Newton [19] methods have been considered.

To compute efficiently the first derivatives of the objective function required by these techniques, the adjoint of the numerical model is generally used [1]. More recently, optimization methods that do not require the first derivatives of the objective function are being examined to avoid the development and maintenance costs associated with using the adjoint [20]. Alternative data assimilation techniques that use ensemble methods to approximate the objective function gradients, rather than using the adjoint, are also being investigated [21,22].

The incremental 4D-Var technique, used commonly in operational centres, approximately solves a sequence of linear least-squares problems and has been shown to be equivalent to the Gauss-Newton (GN) method under standard conditions [23]. In the GN (or incremental) method the linearized problem is solved in an inner loop of the algorithm; the solution to the nonlinear problem is then updated in an outer loop and the problem is re-linearized. The accuracy with which the inner loop is solved is known to affect the convergence of the outer loop [23–25]. In our work, we focus on the convergence of the outer loop, where accurate gradient information is used (as is the case when an adjoint is available) and we assume that the inner loop linear least-squares problem is solved either exactly or inexactly. Furthermore, we use a variable transformation usually applied in operational 4D-Var to precondition the optimization problem, see [26].

A general drawback of the GN method is that given a poor initialization, it is not guaranteed to converge to a solution, known as the 'analysis' state, of the 4D-Var problem [17]. In NWP, the initial guess for the minimization is generally chosen to be the predicted initial state from a previous forecast, known as the 'prior' or 'background' state. However, for some applications of 4D-Var this choice may not be a good enough estimate of the analysis. We show that in such cases, the GN minimization procedure may fail to converge. There are three main strategies that safeguard GN and make it convergent from an arbitrary initial guess: line search, regularization and trust region [18,27]. GN with quadratic regularization (REG) is strongly related to GN with trust region (see Lemma 10.2. of [18]), also referred to as the Levenberg-Marquardt method (LM) [27–29]. Indeed trust region and REG are similar safeguarding strategies for the Gauss Newton method, but their subproblem solutions are different in that the trust region subproblem is nonlinear, requiring more involved and expensive computational techniques while the REG subproblem is linear. Furthermore, REG has interesting connections to Tikhonov regularization which is a crucial statistical tool in data assimilation; REG automatically provides such a problem regularization numerically, at every iteration. An adaptive regularization method using cubic models (ARC) [30,31] is a popular variant that requires (more accurate) second derivative problem information than REG, which is prohibitively expensive for data assimilation. Furthermore, the subproblems in ARC are nonlinear, which again brings in additional computational expense. Therefore, within our

work, we choose to focus on the simpler variant REG and compare its performance to GN with backtracking Armijo line search (LS) and GN alone, applied to the preconditioned 4D-Var problem when there is limited computational time and evaluations available, such as in NWP.

In previous work, the use of a line search strategy in combination with a Quasi-Newton approach was implemented in the ECMWF NWP system to solve the 4D-Var problem and was found to improve the minimization of the objective function [6,8,32]. Météo France [33] and the Meteorological Service of Canada [34] also adopted the method known as M1QN3. This method uses the Wolfe line search conditions [35] to safeguard the convergence on the inner loop level. The Wolfe conditions require the use of additional evaluations of the objective function and especially its gradient, however, which is computationally costly and impractical for use in the outer loop. This is unlike the Armijo condition [36] used in our work, as in [37], which through the use of backtracking only requires additional evaluations of the objective function but not its gradient [18]. We pair GN with backtracking Armijo line search and use a fixed number of computational evaluations to guarantee a reduction in the outer loop objective function (assuming the inner loop is solved to a high accuracy). We compare this method to the GN method and to the GN method safeguarded by quadratic regularization, using a simple, inexpensive updating strategy.

The use of the LM method has been of interest in the data assimilation community because of its similarities with GN and its convergence guarantees. Bergou et al. [38] apply a variation of LM to the 4D-Var problem combined with the use of ensemble methods for the linearized subproblems and prove global convergence under an assumption that only approximate gradient and Jacobian values are available and accurate within a certain probability. The work of [39] builds on that of [38] and proposes a stochastic Levenberg-Marquardt framework that handles both random models and noise in the function evaluations.

Other studies include that of Mandel et al. [40], who apply the incremental 4D-Var method to the weak-constraint (where model error is accounted for) 4D-Var problem with the two-level quasi-geostrophic model, using the exact tangent and adjoint models. They find that the method diverges due to the nonlinearity of the model along a 10 day (long) time-window. They contrast the results of the incremental 4D-Var method with those obtained when they use a LM method to control the convergence of the incremental 4D-Var method, using an inexact solver for the 4D-Var inner-problem. The regularization is fixed across all iterations, and the algorithm's performance is assessed using different values of this parameter. They find that convergence is not guaranteed when the regularization parameter is small while larger values enable the function value to decrease as the method iterates. They conclude that investigations into an adaptive method to adjust the regularization parameter at each iteration would be of interest; this is an aspect present in our work here.

More recently, the authors of [41] propose a novel LM method for application to both zero and non-zero residual problems. This method is similar to the REG method used in our work, except for the use of an additional parameter that corresponds to a successful step of the method balancing local and global convergence requirements. The authors use an example to show how their local convergence is satisfied by the standard 3D-Var problem and assess the performance of their proposed LM method using preliminary numerical experiments. The approach in [42] teams novel preconditioning techniques with the trust region globalization strategy for large scale, nonlinear problems and finds that their method only requires an additional function evaluation per outer loop iteration, a trade-off for robustness of a global convergent solver.

Within this paper, we aim to investigate whether the use of globally convergent optimization methods, namely LS and REG, is beneficial in variational data assimilation, where there is limited time and computational cost available. We focus on this particular question in simple frameworks so that we can fully ascertain its effect. We do not require more sophisticated techniques as we want to be able to see some precise effects of safeguarding the steps, without other features. Hence, we consider the use of globally convergent strategies where we focus on the convergence of the 4D-Var problem on the outer loop level, where accurate gradient information is used (as is the case when an adjoint is available) and the regularization parameter in REG is updated using a simple, inexpensive strategy. We also consider the case of both an exact and inexact subproblem when analyzing the quality of the analysis, for robustness of our conclusions.

Using two test models within the 4D-Var framework, we show that where there is more uncertainty in the background information compared to the observations, the GN method may fail to converge, yet the convergent methods, LS and REG, are able to improve the estimate of the analysis. Assimilation over long time windows is of particular interest. We use data profiles to show numerically that in the long time-window case and in cases where there is higher uncertainty in the background information versus the observations, the globally convergent methods are able to solve more problems than GN in the limited cost available. By 'solve' we mean satisfying a criterion requiring a reduction in the objective function within a set number of evaluations. We also show the effect that poor background information has on the quality of the estimate obtained. We consider the case where the background information is highly inaccurate compared to the observations and find that, even when using an inexact solver, the globally convergent methods outperform GN and the convergence of all three methods is improved when more observations are included along the time-window. Finally, for the case where GN performs well, we recommend further research into the parameter updating strategies used within the globally convergent methods.

The structure of this paper is organized as follows. In Section 2 we outline the strong-constraint 4D-Var problem as a nonlinear least-squares problem and the GN method that is frequently used to solve it. In Section 3 we outline the globally

convergent methods used within this paper. In Section 4 we describe the experimental design including the dynamical models used. In Section 5 we present the numerical results obtained when applying GN and the globally convergent methods to the 4D-Var problem with different features. Finally, we conclude our findings in Section 6. In an appendix we detail the proofs of convergence for the REG and LS methods.

2. Variational data assimilation

2.1. 4D-Var: least-squares formulation

In four-dimensional variational data assimilation (4D-Var), the analysis $\mathbf{x}_0^a \in \mathbb{R}^n$ is obtained by minimizing a objective function consisting of two terms: the background term and the observation term, namely;

$$\mathcal{J}(\mathbf{x}_0) = \frac{1}{2}(\mathbf{x}_0 - \mathbf{x}_0^b)^T \mathbf{B}^{-1}(\mathbf{x}_0 - \mathbf{x}_0^b) + \frac{1}{2}\sum_{i=0}^{N}(\mathbf{y}_i - \mathcal{H}_i(\mathbf{x}_i))^T \mathbf{R}_i^{-1}(\mathbf{y}_i - \mathcal{H}_i(\mathbf{x}_i)). \tag{1}$$

The background term measures the difference between the initial state of the system and the background state vector $\mathbf{x}_0^b \in \mathbb{R}^n$, which contains prior information. The observation term measures the difference between information from observations at times t_i in the observation vector $\mathbf{y}_i \in \mathbb{R}^{p_i}$ and the model state vector $\mathbf{x}_i \in \mathbb{R}^n$ at the same time through use of the observation operator $\mathcal{H}_i : \mathbb{R}^n \to \mathbb{R}^{p_i}$ that maps from the model state space to the observation space. Both terms are weighted by their corresponding covariance matrices to represent the uncertainty in the respective measures, the background error covariance matrix $\mathbf{B} \in \mathbb{R}^{n\times n}$ and the observation error covariance matrices at times t_i, $\mathbf{R}_i \in \mathbb{R}^{p_i \times p_i}$, which are assumed to be symmetric positive definite. We note that observations are distributed both in time and space and there are usually fewer observations available than there are state variables so $p < n$, where $p = \sum_{i=0}^{N} p_i$. The 4D-Var objective function (1) is subject to the nonlinear dynamical model equations which contain the physics of the system

$$\mathbf{x}_i = \mathcal{M}_{0,i}(\mathbf{x}_0), \tag{2}$$

where the nonlinear model $\mathcal{M}_{0,i} : \mathbb{R}^n \to \mathbb{R}^n$ evolves the state vector from the initial time point t_0 to the time point t_i.

We precondition the 4D-Var problem using a variable transform, which has been shown to improve the conditioning of the variational optimization problem [43,44]. To be able to use the negative square root of $\mathbf{B}$ in our variable transformation, we first require the assumption that the matrix $\mathbf{B}$ is full rank. This assumption is satisfied for our choices of $\mathbf{B}$ in Section 5. We define a new variable $\mathbf{v}$ to be,

$$\mathbf{v} = \mathbf{B}^{-1/2}(\mathbf{x}_0 - \mathbf{x}_0^b). \tag{3}$$

The 4D-Var objective function can then be written in terms of $\mathbf{v}$, known as the control variable in data assimilation (DA), and minimized with respect to this instead. Furthermore, by including the model information within the objective function, we are able to write the constrained optimization problem (1)–(2) in the form of an unconstrained optimization problem and apply the minimization methods described later in this paper. The preconditioned 4D-Var objective function is given by

$$\mathcal{J}(\mathbf{v}) = \frac{1}{2}\mathbf{v}^T\mathbf{v} + \frac{1}{2}\sum_{i=0}^{N}(\mathbf{y}_i - \mathcal{H}_i(\mathcal{M}_{0,i}(\mathbf{B}^{1/2}\mathbf{v} + \mathbf{x}_0^b)))^T\mathbf{R}_i^{-1} \times (\mathbf{y}_i - \mathcal{H}_i(\mathcal{M}_{0,i}(\mathbf{B}^{1/2}\mathbf{v} + \mathbf{x}_0^b))). \tag{4}$$

We note that the function (4) is continuously differentiable if the operators $\mathcal{H}_i$ and $\mathcal{M}_{0,i}$ are continuously differentiable. To save both computational cost and time in 4D-Var, tangent linear approximations of the nonlinear operators in (4) are used in the inner loop [5]. The tangent linear model (TLM) and tangent linear observation operator are usually derived by linearizing the discrete nonlinear model equations.

In our nonlinear least-squares problem, namely,

$$\min_{\mathbf{v}} \ \mathcal{J}(\mathbf{v}) := \frac{1}{2}\|\mathbf{r}(\mathbf{v})\|_2^2, \tag{5}$$

the function $\mathcal{J} : \mathbb{R}^n \to \mathbb{R}$ has a special form, where $\mathbf{r}(\mathbf{v}) = [r_1(\mathbf{v}), \ldots, r_{n+p}(\mathbf{v})]^T$ and each $r_j : \mathbb{R}^n \to \mathbb{R}$, for $j = 1, 2, \ldots, n+p$, is referred to as a residual. In (5), $\|\cdot\|_2$ denotes the l_2-norm, which will be used throughout this paper. The function $\mathcal{J}(\mathbf{v})$ in (4) can be written, equivalently, in the form of the objective in (5), where the residual vector $\mathbf{r}(\mathbf{v}) \in \mathbb{R}^{(n+p)}$ and its Jacobian $\mathbf{J}(\mathbf{v})$ are given by

$$\mathbf{r}(\mathbf{v}) = \begin{pmatrix} \mathbf{v} \\ \mathbf{R}_0^{-1/2}(\mathbf{y}_0 - \mathcal{H}_0(\mathbf{B}^{1/2}\mathbf{v} + \mathbf{x}_0^b)) \\ \mathbf{R}_1^{-1/2}(\mathbf{y}_1 - \mathcal{H}_1(\mathcal{M}_{0,1}(\mathbf{B}^{1/2}\mathbf{v} + \mathbf{x}_0^b))) \\ \vdots \\ \mathbf{R}_N^{-1/2}(\mathbf{y}_N - \mathcal{H}_N(\mathcal{M}_{0,N}(\mathbf{B}^{1/2}\mathbf{v} + \mathbf{x}_0^b))) \end{pmatrix} \quad \text{and}$$

$$\mathbf{J}(\mathbf{v}) = \begin{pmatrix} \mathbf{I} \\ -\mathbf{R}_0^{-1/2}\mathbf{H}_0\mathbf{B}^{1/2} \\ -\mathbf{R}_1^{-1/2}\mathbf{H}_1\mathbf{M}_{0,1}\mathbf{B}^{1/2} \\ \vdots \\ -\mathbf{R}_N^{-1/2}\mathbf{H}_N\mathbf{M}_{0,N}\mathbf{B}^{1/2} \end{pmatrix}, \tag{6}$$

where

$$\mathbf{M}_{0,i} = \left.\frac{\partial \mathcal{M}_{0,i}}{\partial \mathbf{v}}\right|_{\mathcal{M}_{0,i}(\mathbf{B}^{1/2}\mathbf{v}+\mathbf{x}_0^b)} \quad \text{and} \quad \mathbf{H}_i = \left.\frac{\partial \mathcal{H}_0}{\partial \mathbf{v}}\right|_{\mathcal{M}_{0,i}(\mathbf{B}^{1/2}\mathbf{v}+\mathbf{x}_0^b)} \tag{7}$$

are the Jacobian matrices of the model operator $\mathcal{M}_{0,i}$ and observation operator $\mathcal{H}_i$ respectively, $\mathbf{M}_{0,i} \in \mathbb{R}^{n\times n}$ is the tangent linear of $\mathcal{M}_{0,i}$ and $\mathbf{H}_i \in \mathbb{R}^{p_i\times n}$ is the tangent linear of $\mathcal{H}_i$ [2]. Note that the second entries of $\mathbf{r}(\mathbf{v})$ and $\mathbf{J}(\mathbf{v})$ in (6) do not feature the nonlinear and tangent linear model operators as these entries reflect time point $i=0$. In practice, an adjoint method is used to calculate the gradient of (4), defined as

$$\nabla \mathcal{J}(\mathbf{v}) = \mathbf{J}(\mathbf{v})^T \mathbf{r}(\mathbf{v}). \tag{8}$$

The Hessian is the matrix of second-order partial derivatives of (4),

$$\nabla^2 \mathcal{J}(\mathbf{v}) = \mathbf{J}(\mathbf{v})^T \mathbf{J}(\mathbf{v}) + \sum_{j=1}^{n+p} r_j(\mathbf{v}) \nabla^2 r_j(\mathbf{v}). \tag{9}$$

In data assimilation, the second-order terms in (9) are often difficult to calculate in the time and cost available and too large to store, and so one cannot easily use Newton-type methods for 4D-Var. Therefore, a first-order approximation to the Hessian of the objective function (4) is used, resulting in a GN method, and is given by

$$\mathbf{S} = \mathbf{J}(\mathbf{v})^T \mathbf{J}(\mathbf{v}) = \mathbf{I} + \sum_{i=0}^{N} \mathbf{B}^{1/2} \mathbf{M}_{0,i}^T \mathbf{H}_i^T \mathbf{R}_i^{-1} \mathbf{H}_i \mathbf{M}_{0,i} \mathbf{B}^{1/2}, \tag{10}$$

which is, by construction, full rank and symmetric positive definite. The condition number in the l_2-norm of (10), $\kappa(\mathbf{S})$, is the ratio of its largest and smallest eigenvalues and is related to the number of iterations used for the linear minimization problems in 4D-Var and how sensitive the estimate of the initial state is to perturbations of the data. We can use $\kappa(\mathbf{S})$ to indicate how quickly and accurately the optimization problem can be solved [45].

2.2. 4D-Var implementation

The incremental 4D-Var method, which was first proposed for practical implementation of the NWP problem in [5], has been shown to be equivalent to the GN method when an exact TLM is used in the inner loop. When an approximate TLM is used, the method is equivalent to an inexact GN method [19,23]. A summary of the GN method is given next.

In Algorithm 1, the updated control variable $\mathbf{v}^{(k+1)}$ is computed by finding a step $\mathbf{s}^{(k)}$ that satisfies (11), which is known as the preconditioned linearized subproblem. By substituting $\mathbf{v}^{(k+1)}$ into (3) and rearranging, we obtain the current estimate $\mathbf{x}_0^{(k+1)}$ of the initial state to the original nonlinear 4D-Var problem.

To reduce the computational cost in large DA systems and to solve the DA problem in real time, the series of problems (11) can be solved approximately in

Algorithm 1 GN algorithm applied to (5) [17].

Step 0: Initialisation. Given $\mathbf{v}^{(0)} \in \mathbb{R}^n$ and some stopping criteria. Set $k = 0$.
Step 1: Check stopping criteria. While the stopping criteria are not satisfied, do:
Step 2: Step computation. Compute a step $\mathbf{s}^{(k)}$ that satisfies

$$\mathbf{J}(\mathbf{v}^{(k)})^T\mathbf{J}(\mathbf{v}^{(k)})\mathbf{s}^{(k)} = -\mathbf{J}(\mathbf{v}^{(k)})^T\mathbf{r}(\mathbf{v}^{(k)}). \tag{11}$$

Step 3: Iterate update. Set $\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} + \mathbf{s}^{(k)}$, $k := k + 1$ and go to Step 1.

the inner loop using iterative optimization methods such as Conjugate Gradient (CG) where a limited number of CG iterations are allowed and an exact or approximate $\mathbf{J}$ is used [19].

We note that the step calculation (11) uniquely defines $\mathbf{s}^{(k)}$, and $\mathbf{s}^{(k)}$ is a descent direction when $\mathbf{J}(\mathbf{v})$ is full column rank. This is the case in 4D-Var as the Jacobian, $\mathbf{J}(\mathbf{v})$ in (6) is full column rank due to the presence of the identity matrix, thus ensuring that $\mathbf{s}^{(k)}$ is a descent direction.

The definitions of two solution types, namely, local and global minima, are stated in Appendix, along with a brief explanation of the local convergence property of GN. Although the GN method benefits from local convergence properties, convergence can only be guaranteed if the initial guess $\mathbf{v}^{(0)}$ of the algorithm is in some neighborhood around an (unknown) local solution $\mathbf{v}^*$, that is, convergence from an arbitrary initial guess is not guaranteed [17]. Even if the GN method does converge, it may not necessarily converge to the global minimum due to the fact that multiple local minima of a nonlinear least-squares objective function may exist.

GN has no way of adjusting the length of the step $\mathbf{s}^{(k)}$ and hence, may take steps that are too long and fail to decrease the objective function value and thus to converge, see Example 10.2.5 in [17] and later in Section 5 where the poor performance of GN is demonstrated. As GN only guarantees local convergence, we are interested in investigating methods that converge when $\mathbf{v}^{(0)}$ is far away from a local minimizer $\mathbf{v}^*$. We refer to these methods as 'globally convergent'. Mathematical theory on global strategies can be found in [18] and [17]. Two globally convergent methods are GN with line search and GN with quadratic regularization, which use a strategy within the GN framework to achieve convergence to a stationary point given an arbitrary initial guess by adjusting the length of the step. These methods will be presented in the next section.

3. Globally convergent methods

Within this section, we outline the two globally convergent algorithms that we apply in Section 5 to the preconditioned 4D-Var problem.

3.1. Gauss-Newton with line search (LS)

A line search method aims to restrict the step $\mathbf{s}^{(k)}$ in (11) so as to guarantee a decrease in the value of $\mathcal{J}$. Within our work, an inexact line search method known as the backtracking-Armijo (bArmijo) algorithm is used within the inner loop of GN to find a step length $\alpha > 0$ that satisfies the Armijo condition [36]. The Gauss-Newton with backtracking-Armijo line search (LS) method is as follows.

Algorithm 2 LS algorithm applied to (5) [18].

Step 0: Initialisation. Given $\mathbf{v}^{(0)} \in \mathbb{R}^n$, $\tau \in (0,1)$ and $\beta \in (0,1)$ and $\alpha_0 > 0$ and some stopping criteria. Set $k = 0$.

Step 1: Check stopping criteria. While the stopping criteria are not satisfied, do:

Step 2: Step computation. Compute a step $\mathbf{s}^{(k)}$ that satisfies

$$\mathbf{J}(\mathbf{v}^{(k)})^T\mathbf{J}(\mathbf{v}^{(k)})\mathbf{s}^{(k)} = -\mathbf{J}(\mathbf{v}^{(k)})^T\mathbf{r}(\mathbf{v}^{(k)}) \tag{12}$$

and set $\alpha^{(k)} = \alpha_0$.

Step 3: Check Armijo condition. While the following (Armijo) condition is not satisfied

$$\mathcal{J}(\mathbf{v}^{(k)} + \alpha^{(k)}\mathbf{s}^{(k)}) \leq \mathcal{J}(\mathbf{v}^{(k)}) + \beta\alpha^{(k)}{\mathbf{s}^{(k)}}^T\nabla\mathcal{J}(\mathbf{v}^{(k)}), \tag{13}$$

do:

Step 4: Shrink stepsize. Set $\alpha^{(k)} := \tau\alpha^{(k)}$ and go to Step 3.

Step 5: Iterate update. Set $\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} + \alpha^{(k)}\mathbf{s}^{(k)}$, $k := k+1$ and go to Step 1.

In Algorithm 2, the control parameter β in (13) is typically chosen to be small (see [18]). The step Equation (12) is the same as the GN step Equation (11); thus when $\alpha^{(k)} = 1$, the GN and LS iterates coincide at (the same) point $\mathbf{v}^{(k)}$. The use of condition (13) in this method ensures that the accepted steps produce a sequence of strictly decreasing function values given $\nabla\mathcal{J}(\mathbf{v}^{(k)})^T\mathbf{s}^{(k)} < 0$. This latter condition is satisfied by $\mathbf{s}^{(k)}$ defined in (12) whenever $\mathcal{J}(\mathbf{v}^{(k)})$ is full column rank (which is the case here) as mentioned in Section 2 [18].

Despite its global convergence property (see Appendix A.1), the LS method has some disadvantages. We remark that the use of the step length $\alpha^{(k)}$ may sometimes unnecessarily shorten the step $\mathbf{s}^{(k)}$, slowing down the convergence. Furthermore, LS may be computationally costly due to the need to calculate the value of the function $\mathcal{J}$ each time $\alpha^{(k)}$ is adjusted, although more sophisticated updating strategies for α may be used to try to reduce this effect.

Other line search strategies are possible such as Wolfe, Goldstein-Armijo and more [18], but they are more involved and potentially more computationally

costly. As LS requires the re-evaluation of the outer loop objective function each time it adjusts its line search parameter, its applicability to real systems has been in doubt due to the computational cost limitations in 4D-Var [6]. In Section 5, we show that given the same cost as the GN method, the LS method can in some cases, better minimize the preconditioned 4D-Var objective function.

3.2. Gauss-Newton with regularization (REG)

The GN method may also be equipped with a globalization strategy by including a regularization term $\gamma^{(k)}\mathbf{s}^{(k)}$ in the step calculation (11) of Algorithm 1. This ensures that the accepted steps produce a sequence of monotonically decreasing function values. This is a common variation of the GN method known as the Levenberg-Marquardt method, proposed in [28] and [29]. The effect of the regularization parameter $\gamma^{(k)}$ is to implicitly control the length of the step $\mathbf{s}^{(k)}$. Increasing $\gamma^{(k)}$ shortens the steps, thus increasing the possibility that the procedure will decrease the objective function in the next iteration. The REG method can be summarised as follows; see Algorithm 3.

As in Algorithms 1 and 2, the step Equation (14) is solved either exactly or inexactly in the numerical experiments in Section 5. We note that when $\gamma^{(k)} = 0$ in (14), the REG step in (14) is the same as the GN step in (11). By comparing (14) with (11), we are able to see how the REG step differs from the GN step. The diagonal entries of the Hessian of the 4D-Var objective function (4) are increased by the regularization parameter $\gamma^{(k)}$ at each iteration of the REG method. The method is able to vary its step between a GN and a gradient descent step by adjusting $\gamma^{(k)}$ (see [18]) but may be costly due to the need to calculate the value of the function $\mathcal{J}$ to assess the step. Note that other choices of the factors $\frac{1}{2}$ and 2 for updating $\gamma^{(k)}$ in (18) are possible and even more sophisticated variants for choosing $\gamma^{(k)}$ have been proposed. The proof of global convergence of the REG method is presented in Appendix A.2.

4. Experimental design

Before evaluating the GN, LS and REG methods numerically, we first explain the experimental design.

Twin experiments are commonly used to test DA methods. They use error statistics that satisfy the DA assumptions as well as synthetic observations generated by running the nonlinear model forward in time to produce a reference state (not generally a local minimum of (5)). Within this section, we define our choices for the twin experimental design. We begin by briefly outlining two commonly used dynamical models, which are sensitive to initial conditions (chaotic nature), a property shared with NWP models.

Algorithm 3 REG algorithm applied to (5) [46].

Step 0: Initialisation. Given $\mathbf{x}^{(0)} \in \mathbb{R}^n$, $1 > \eta_2 \geq \eta_1 > 0$, $\gamma^{(0)} > 0$ and some stopping criteria. Set $k = 0$.

Step 1: Check stopping criteria. While the stopping criteria are not satisfied, do:

Step 2: Step computation. Compute a step $\mathbf{s}^{(k)}$ that satisfies

$$\left(\mathbf{J}(\mathbf{v}^{(k)})^T\mathbf{J}(\mathbf{v}^{(k)}) + \gamma^{(k)}\mathbf{I}\right)\mathbf{s}^{(k)} = -\mathbf{J}(\mathbf{v}^{(k)})^T\mathbf{r}(\mathbf{v}^{(k)}). \tag{14}$$

Step 3: Iterate update. Compute the ratio

$$\rho^{(k)} = \frac{\mathcal{J}(\mathbf{v}^{(k)}) - \mathcal{J}(\mathbf{v}^{(k)} + \mathbf{s}^{(k)})}{\mathcal{J}(\mathbf{v}^{(k)}) - m(\mathbf{s}^{(k)})}, \tag{15}$$

where

$$m(\mathbf{s}^{(k)}) = \frac{1}{2}\|\mathbf{J}(\mathbf{v}^{(k)})\mathbf{s}^{(k)} + \mathbf{r}(\mathbf{v}^{(k)})\|_2^2 + \frac{1}{2}\gamma^{(k)}\|\mathbf{s}^{(k)}\|_2^2. \tag{16}$$

Set

$$\mathbf{v}^{(k+1)} = \begin{cases} \mathbf{v}^{(k)} + \mathbf{s}^{(k)}, & \text{if } \rho^{(k)} \geq \eta_1 \\ \mathbf{v}^{(k)}, & \text{otherwise.} \end{cases} \tag{17}$$

Step 4: Regularisation parameters update. Set

$$\gamma^{(k+1)} = \begin{cases} \frac{1}{2}\gamma^{(k)}, & \text{if } \rho^{(k)} \geq \eta_2 \text{ (very successful iteration)} \\ \gamma^{(k)}, & \text{if } \eta_1 \leq \rho^{(k)} < \eta_2 \text{ (successful iteration)} \\ 2\gamma^{(k)}, & \text{otherwise, (unsuccessful iteration)} \end{cases} \tag{18}$$

Let $k := k + 1$ and go to Step 1.

4.1. Models

Lorenz 1963 model (L63) Proposed in [47], the Lorenz 63 model (L63) is a popular experimental dynamical system that represents meteorological processes using a simple model. The model consists of three nonlinear, ordinary differential equations given as

$$\begin{aligned} \frac{\mathrm{d}x}{\mathrm{d}t} &= \sigma(y - x), \\ \frac{\mathrm{d}y}{\mathrm{d}t} &= x(\rho - z) - y, \\ \frac{\mathrm{d}z}{\mathrm{d}t} &= xy - \beta z, \end{aligned} \tag{19}$$

where the state vector consists of $n=3$ time-dependent variables $\mathbf{x} = [x(t), y(t), z(t)]^T \in \mathbb{R}^3$. The scalar parameters are chosen to be $\sigma = 10$, $\rho = \frac{8}{3}$ and $\beta = 28$, making the system chaotic. A second-order Runge-Kutta method is used to discretize the model equations using a time step $\Delta t = 0.025$.

Lorenz 1996 model (L96) Another popular experimental system is the atmospheric Lorenz 96 model (L96) [48] given by the following n equations,

$$\frac{\mathrm{d}x_j}{\mathrm{d}t} = -x_{j-2}x_{j-1} + x_{j-1}x_{j+1} - x_j + F, \tag{20}$$

where $j = 1, 2, \ldots, n$ is a spatial coordinate. For a forcing term $F=8$ and $n=40$ state variables, the system is chaotic [48]. The variables are evenly distributed over a circle of latitude of the Earth with n points with a cyclic domain and a single time unit is equivalent to approximately 5 atmospheric days. A fourth-order Runge-Kutta method is used to discretize the model equations using a time step $\Delta t = 0.025$ (approximately 3 hours).

For both the L63 and L96 models, the time-window length t_a is varied in the numerical experiments in Section 5.1. We will now outline how we formulate the twin experiments, beginning with generating the reference state.

4.2. Twin experiments

The reference state at time t_0, $\mathbf{x}_0^{ref}$ is used as the basis of a twin experiment in the definition of the background state (the initial guess for the optimization algorithms) as well as to generate the observations using a nonlinear model run called the 'nature' run. We begin by explaining how we obtain $\mathbf{x}_0^{ref}$.

Reference state A vector of length n is drawn from the uniform distribution and used as the initial vector of state variables $\mathbf{x}^{rand}$. For the L63 model, $\mathbf{x}^{rand}$ is integrated forward using a second-order Runge-Kutta method, which is spun-up over 1000 time steps to obtain the reference state on the model attractor for the L63 twin experiments, $\mathbf{x}_0^{ref} \in \mathbb{R}^3$. This is the same for the L96 model except a fourth-order Runge-Kutta method is used to obtain $\mathbf{x}_0^{ref} \in \mathbb{R}^{40}$. The reference state at time t_0, $\mathbf{x}_0^{ref}$ can then be used to obtain the full nonlinear model trajectory.

We next explain how we obtain the background state vector used within our twin experiments to be used as the initial guess for the optimization algorithms.

Background In 4D-Var, the initial guess for the optimization algorithm is taken to be the background state at time t_0, $\mathbf{x}_0^b$, which incorporates information from previous forecasts. In our experiments, the background state vector $\mathbf{x}_0^b$ is generated by adding Gaussian noise

$$\varepsilon_{\mathbf{b}} \sim \mathcal{N}(0, \mathbf{B}), \tag{21}$$

to the reference state at time t_0, $\mathbf{x}_0^{ref}$. For the background error covariance matrix, we choose $\mathbf{B} = \sigma_b^2 \mathbf{I}_n$ where σ_b^2 is the background error variance. The standard

deviations of the errors from the reference state for each model are based on the average order of magnitude of the entries of $\mathbf{x}_0^{ref}$. For the L63 experiments, $\sigma_b^2 = 0.25, 1, 6.25$ and 25 represent a 5%, 10%, 25% and 50% error respectively. Similarly for the L96 experiments we set $\sigma_b^2 = 0.0625, 0.25, 1.5625$ and 6.25.

As previously mentioned, we generate synthetic observations from a nonlinear model run using the reference state at time t_0, $\mathbf{x}_0^{ref}$. We next describe the choices we made when specifying these observations.

Observations We consider both the spatial and temporal locations of the observations. We assume that for both models observations of single state variables are taken and $\mathbf{H}_i$ are the exact observation operators at times t_i used to map to observation space. For the L63 model, we consider $p = 2$ observations, one of x and one of z per observation location in time. For the L96 model, we consider an observation of the first half of the state variables per observation location in time. This choice mimics what we may expect in reality where we have more observations concentrated in one part of the globe. For both models, we first consider only one set of observations at time N (Nobs1) and then show the effect of using more observations along the time-window in later experiments. We use imperfect observations where the observations $\mathbf{y}_i$ are generated by adding Gaussian noise

$$\varepsilon_{\mathbf{o}} \sim \mathcal{N}(0, \mathbf{R}_i), \tag{22}$$

to $\mathbf{H}_i \mathbf{x}_i^{ref}$ for each observation location in time. For the observation error covariance matrix we choose $\mathbf{R}_i = \sigma_o^2 \mathbf{I}_p$ where σ_o^2 is the observation error variance. We expect the problem (4) to be more ill-conditioned, thus difficult to solve accurately, when the ratio

$$\frac{\sigma_b}{\sigma_o} \tag{23}$$

is large [43,44]. The ratio (23) controls the influence of the observation term in the preconditioned objective function (4). For all experiments, we set the standard deviation of the observation error to be 10% of the average order of magnitude of the entries of $\mathcal{H}(\mathbf{x}_i^{ref})$ for both models. For the L63 model, this is $\sigma_o^2 = 1$ and for the L96 model, this is $\sigma_o^2 = 0.25$. We vary the background error variance σ_b^2 above and below σ_o^2 such that the ratio (23) varies. This can be thought of as having more confidence in the observations compared to background when $\sigma_b > \sigma_o$ and vice versa. Furthermore, as the initial guess is set to be the background state vector, which is dependent on the value of σ_b, by varying σ_b^2 we are essentially varying the initial guess of the algorithms, thus eliminating starting point bias from our results [49]. It is important to recall here that under certain conditions, the GN method is known for its fast convergence properties when in close vicinity to a local minimum, see [17]. By choosing a small value of σ_b^2, we expect the performance of GN to beat that of both LS and REG as it does not require the adjustment of the additional parameters $\alpha^{(k)}$ and $\gamma^{(k)}$. Also, when

assuming that the observations are more accurate than the background, the use of more observation locations in time means that we are constraining the estimate of the initial state more tightly to the reference state in the twin experiment design. The effect this has on the convergence of the optimization methods will be investigated. We next outline the algorithmic choices we have made.

4.3. Algorithmic choices

Stopping criteria We now outline the criteria used to terminate Algorithms 1, 2 and 3. Due to the limited time and computational cost available in practice, the GN method is not necessarily run to convergence and a stopping criterion is used to limit the number of iterations. Each calculation of the residual vector $\mathbf{r}(\mathbf{v})$ requires the non-linear model to be run forward to obtain the state at each observation location in time. This can then be used to calculate the value of the objective function. Furthermore, one run of the adjoint model is required to calculate the gradient.

To reduce computational cost in practical implementations of 4D-Var, the inner loop problem is solved at a lower resolution than the outer loop problem [50]. However, as the dimension of the problems used within this paper are relatively small compared to DA systems in practice, we solve the full resolution inner loop problem using the full resolution residual and Jacobian given in (6) and for the majority of our experiments we solve the inner loop problem using MATLAB's backslash operator where an appropriate solver is chosen according to the properties of the Hessian matrix $\nabla^2\mathcal{J}(\mathbf{v})$ (see [51] for more details). For those experiments where we state that we are solving the inner loop problem inexactly, we use MATLAB's preconditioned CG method (see [52] for more details). We use MATLAB version R2016b throughout. The limit on the total number of function and Jacobian evaluations is achieved by using the following criterion

$$k_J + l \leq \tau_e, \tag{24}$$

where k_J is the total number of Jacobian evaluations (which is equivalent to the number of outer iterations k in 4D-Var), l is the total number of function evaluations and τ_e is the tolerance. The tolerance τ_e can be chosen according to the maximum number of evaluations desired. We note that for GN, $k_J = l$ as the method requires as many Jacobian evaluations as function evaluations. However, for both LS and REG there could be more than one function evaluation per Jacobian evaluation since for unsuccessful steps, the Jacobian is not updated so $k_J \leq l$.

To ensure that the algorithms are stopped before the function values stagnate, the following common termination criterion based on the relative change in the function at each iteration is also used

$$\frac{|\mathcal{J}(\mathbf{v}^{(k-1)}) - \mathcal{J}(\mathbf{v}^{(k)})|}{1 + \mathcal{J}(\mathbf{v}^{(k)})} \leq \tau_s, \tag{25}$$

for $k \geq 1$, where τ_s is the tolerance, chosen to be 10^{-5}. The criterion (25) is used throughout Section 5 unless indicated otherwise.

We expect the norm of the gradient of the objective function, $\|\nabla \mathcal{J}(\mathbf{v}^{(k)})\|$ to be close to zero at a stationary point. The following termination criterion will be used in Section 5.2 to identify whether or not a given method has located a stationary point

$$\|\nabla \mathcal{J}(\mathbf{v}^{(k)})\| \leq 10^{-5}. \tag{26}$$

Parameter choices For the LS method, we choose $\alpha_0 = 1$ so that the first step assessed by the bArmijo rule is the GN step. We set $\beta = 0.1$ and to adjust the step length, $\tau = 0.5$.

For the REG method, we select the initial regularization parameter to be $\gamma^{(0)} = 1$ so that the condition in Algorithm 3, $\gamma^{(0)} > 0$, is satisfied and the REG step differs from the GN step. Furthermore, we choose $\eta_1 = 0.1$ and $\eta_2 = 0.9$ to assess how well the model (16) approximates the true function value at the next iteration.

For all three optimization methods, we set $\tau_e = 8, 100$ or 1000 depending on the experiment. The choice of $\tau_e = 8$ comes from that which is used operationally in the ECMWF Integrated Forecasting System [53], whereas the choice of $\tau_e = 100$ or 1000 is used to measure the performance of the optimization methods when closer to convergence.

In order to best present our results, we use data profiling described as follows.

Data profiles A *data profile* shows the proportion of problems a given method can solve within a fixed amount of work (τ_e) and a given tolerance (τ_f) of the change in the function value [54]. To ensure the robustness of our results, we apply the three optimization methods to a series of n_r randomly generated problems, where the randomness occurs through the background and observation error vectors, $\varepsilon_{\mathbf{b}}$ and $\varepsilon_{\mathbf{o}}$. For each realization, a new $\varepsilon_{\mathbf{b}}$ and $\varepsilon_{\mathbf{o}}$ are generated from their respective distributions, (21) and (22). The following criterion proposed in [54] is used to flag that an estimate of the initial state has been obtained by an optimization method

$$\frac{\mathcal{J}(\mathbf{v}_0^{(l)}) - \mathcal{J}(\mathbf{v}_0^t)}{\mathcal{J}(\mathbf{v}_0^{(0)}) - \mathcal{J}(\mathbf{v}_0^t)} \leq \tau_f, \tag{27}$$

where $\mathbf{v}_0^t$ is a solution of (4) referred to as the 'truth' and τ_f is the tolerance. The measure (27) compares the optimality gap $\mathcal{J}(\mathbf{v}_0^{(l)}) - \mathcal{J}(\mathbf{v}_0^t)$ relative to the best reduction $\mathcal{J}(\mathbf{v}_0^{(0)}) - \mathcal{J}(\mathbf{v}_0^t)$ [54]. This ensures that the 4D-Var problem is only flagged as solved by the optimization method once the value of the objective function is within some error (τ_f) of the truth.

For our problems, the truth is unknown. We only know that, due to the nonlinearity of the 4D-Var problem, there may exist many values of $\mathbf{v}_0$ that could

minimize (4) locally. We are interested in the estimate $\mathbf{v}_0^t$ that gives the greatest reduction in (4) that any of the three methods can obtain. Therefore, we set the truth to be the $\mathbf{v}_0^{(l)}$ obtained by any of the three methods that gives the smallest function value within the given number of evaluations. Using this criterion allows us to benchmark the methods against each other using data profiles.

For each experiment, we plot the proportion of the same $n_r = 100$ realisations solved by each method against the relative accuracy obtained, τ_f. The relative accuracy obtained is varied using $\tau_f = 10^{-i}$, where $i = 0, 0.01, 0.02, \ldots, 5$.

5. Numerical results

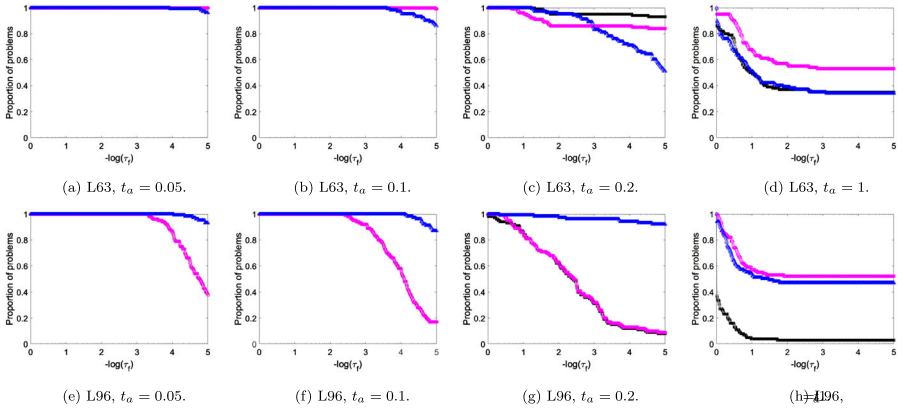
In this section, we present the results when applying GN, LS and REG using the experimental design described in the previous section. We begin by conducting experiments showing the effect of the length of the assimilation time-window on the convergence of the three methods.

5.1. Effect of time-window length

We produce data profiles for different time-window lengths to understand the effect this has on the convergence of each method while limiting the number of function and Jacobian evaluations to $\tau_e = 8$. We choose a background error of 50% and an observation error of 10% so that the ratio (23) is large relative to the other cases we consider. For both the L63 and L96 models, we consider both short and long time-window lengths of 6 hours ($t_a = 0.05$), 12 hours ($t_a = 0.1$), 1 day ($t_a = 0.2$) and 5 days ($t_a = 1$) with the results shown in Figure 1.

From Figure 1, we see that as the length of the time-window of both the L63 and L96 problems is increased, the performance of the GN, LS and REG methods suffers.

For the L63 problems, Figure 1(a,b) show that GN and LS perform similarly and solve more problems to high accuracy than REG. However, as the tolerance τ_f is increased in both of these figures, we find that REG is solving all of the problems. Therefore, the REG estimate when the tolerance τ_f is small must be close to that of GN and LS. In Figure 1(c), both LS and REG solve fewer problems compared to GN, even for relatively large choices of τ_f. However, there is a choice of τ_f where all three methods are solving all problems, again indicating that the LS and REG estimates are close to the GN estimate. The initial guess for the three methods (the background) appears to be close enough to the solution and so the GN step is able to attain a sufficient decrease in the objective function as predicted by its local convergence properties. LS and REG are inadvertently shortening the GN step, which is a good step in the short time-window case. As we know, LS and REG need to adjust their respective parameters, $\alpha^{(k)}$ and $\gamma^{(k)}$ to attain GN's fast local convergence, so LS and REG are requiring more evaluations than GN to achieve the same result. For the L96 short time-window results in Figure 1(e–g),



(a) L63, $t_a = 0.05$. (b) L63, $t_a = 0.1$. (c) L63, $t_a = 0.2$. (d) L63, $t_a = 1$.

(e) L96, $t_a = 0.05$. (f) L96, $t_a = 0.1$. (g) L96, $t_a = 0.2$. (h) L96,

Figure 1. Data profiles for the GN (black squares), LS (magenta circles) and REG (blue triangles) methods applied to the L63 and L96 problems using different time-window lengths t_a. These show the proportion of $n_r = 100$ problems solved by each of the methods against the specified accuracy $-\log(\tau_f)$ when $\tau_e = 8$. The GN line is below the LS line in (a), (b), (e), (f) and (g). (a) L63, $t_a = 0.05$. (b) L63, $t_a = 0.1$. (c) L63, $t_a = 0.2$. (d) L63, $t_a = 1$. (e) L96, $t_a = 0.05$. (f) L96, $t_a = 0.1$. (g) L96, $t_a = 0.2$. (h) L96, $t_a = 1$.

this is not the case. In fact, REG is outperforming GN and LS and it appears that LS is mimicking the behavior of GN quite closely as the GN step is attaining a sufficient decrease in the objective function. However the decrease that the REG step is achieving appears to be much greater for the L96 problems. Therefore, REG is able to solve a greater number of problems within a higher level of accuracy, which explains the difference between the L63 results in Figure 1(a,b) and the L96 results in Figure 1(e,f).

The long time-window results for the L63 and L96 problems are shown in Figure 1(d,h), respectively. In both figures, LS is outperforming GN. For the L63 problems, the performance of GN does not differ much from the performance of REG. However, comparing the performance of GN in Figure 1(c) with Figure 1(d), we can see that performance of GN has deteriorated greatly when increasing the length of the time-window. In fact, in the results where even longer time-windows are used (not included here), LS and REG outperform the GN method for the L63 problems, as in Figure 1(h).

For the remainder of our experiments, we set $t_a = 1$ in order to consider a long time-window case only, as this is where we expect to see the greatest benefit from the globally convergent methods.

5.2. Behaviour of methods and stagnation of GN

In order to gain an understanding of how the globally convergent methods, LS and REG, compare with GN, we next demonstrate the behavior of GN, LS and REG when applied to typical preconditioned 4D-Var L63 and L96 problems, where the background error is large and the time-window length is long.

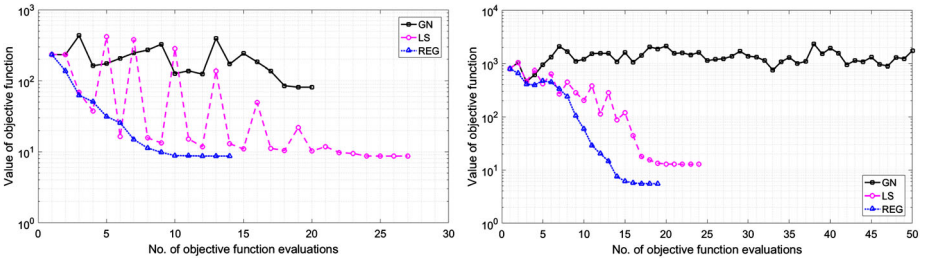


Figure 2. Convergence plots showing the value of the objective function at each iteration (including unsuccessful iterations) of the GN (black squares), LS (magenta circles) and REG (blue triangles) methods when applied to a L63 problem (a) and a L96 problem (b). (a) L63, Nobs1, $\sigma_b^2 = 25$, $\mathbf{B} = \sigma_b^2\mathbf{I}$, $t_a = 1$, $\tau_e = 100$. (b) L96, Nobs1, $\sigma_b^2 = 6.25$, $\mathbf{B} = \sigma_b^2\mathbf{I}$, $t_a = 1$ $\tau_e = 100$.

Figure 2 shows the convergence plots for two typical realisations when using the GN, LS and REG methods to obtain a solution to the preconditioned 4D-Var problem with the L63 and L96 models. In this figure, the total number of function and Jacobian evaluations allowed is set to $\tau_e = 100$ for both the L63 and the L96 problems to see if any progress is made beyond the number of evaluations allowed in practice. We recall that GN updates the gradient (8) when the function (4) is updated, so there are as many function evaluations as Jacobian evaluations. However, both LS and REG only update the Jacobian on successful iterations when there is a reduction in the objective function. Therefore, the total number of evaluations used by each of the methods could consist of a different combination of function and Jacobian evaluations. As in Section 5.1, we set the ratio (23) to be large. It is in this case that we are able to best demonstrate the benefit of the globally convergent methods, LS and REG. In Figure 2, we set $\tau_s = 10^{-3}$ to ensure that the methods stop before the function values stagnate. As Figure 2 includes function evaluations for both successful and unsuccessful step calculations, it is natural to see jumps in the function values of LS and REG while their parameters, $\alpha^{(k)}$ and $\gamma^{(k)}$ are being adjusted to guarantee a reduction in the function.

For the L63 problems (Figure 2(a)), all three methods stop when the relative change in the function criterion (25) is satisfied and before the limit on the total number of function and Jacobian evaluations (24) is met. Table 1 supports this figure by showing the algorithmic output for each of the GN, LS and REG methods when two different stopping criteria are used. From these results, we see that both LS and REG stop at the same function value, although REG requires fewer evaluations to do so, and that GN is converging towards a larger value of the objective function (4) than LS and REG. By instead stopping on the criterion (26) and setting $\tau_e = 1000$, we see in Table 1 that all three methods are still making progress on the gradient and iterate level, indicating that the methods are in fact

Table 1. Table of algorithmic output when applying, GN, LS and REG to a typical realization of the L63 problems, corresponding to Figure 2(a).

Criteria	Method	l	k_J	$\mathcal{J}(\mathbf{v}^{(k_J)})$	$\|\mathbf{v}^{(k_J)} - \mathbf{v}^{(k_J-1)}\|$	$\|\nabla\mathcal{J}(\mathbf{v}^{(k_J)})\|$
	GN	20	20	81.55	0.42	86.35
(25)	LS	27	14	8.69	0.03	5.18
	REG	14	14	8.69	0.05	1.00
	GN	101	101	78.87	3.54^{-8}	8.47^{-6}
(26)	LS	43	27	8.69	8.21^{-7}	8.31^{-6}
	REG	66	66	8.69	7.34^{-7}	9.24^{-6}

Table 2. Table of algorithmic output when applying, GN, LS and REG to a typical realization of the L96 problems, corresponding to Figure 2(b).

Criteria	Method	l	k_J	$\mathcal{J}(\mathbf{x}^{(k_J)})$	$\|\mathbf{v}^{(k_J)} - \mathbf{v}^{(k_J-1)}\|$	$\|\nabla\mathcal{J}(\mathbf{v}^{(k_J)})\|$
(25)	GN	50	50	1728.99	20.02	5758.47
	LS	24	14	12.72	0.07	10.09
	REG	19	16	5.52	0.08	1.89
(26)	GN	500	500	960.32	15.88	8015.13
	LS	967	32	12.71	0	10.09
	REG	967	32	5.51	0	0.03

locating stationary points despite a small change in the function value beyond those shown in Figure 2.

For the L96 problems (Figure 2(b)), LS and REG stop when (25) is satisfied and before (24) is satisfied, whereas GN only satisfies (24). Table 2 supports this figure by showing the algorithmic output for each of the GN, LS and REG methods when two different stopping criteria are used. From these results, we see that both GN and LS are stopping at a larger value of the objective function (4) than REG. Recall that the norm of the gradient criterion (26) can be used to identify whether or not a given method has located a stationary point. The values of $\|\nabla\mathcal{J}(\mathbf{v}^{(k_J)})\|$ for LS and REG when the relative change in the function criterion (25) is used are much smaller than that of GN. However, when we instead use the norm of the gradient criterion (26) and limit the number of iterations to $\tau_e = 1000$, the methods stop on the limit of the number of iterations. Therefore, our results do not indicate that the estimates of LS and REG may indeed be stationary points of the objective function as they did for the L63 problems. However, LS and REG are are able to make some improvement (REG more so than LS) on the gradient norm level, unlike GN, which appears to fluctuate at gradient level, even after $\tau_e = 1000$ evaluations.

Table 2 shows that as LS and REG iterate beyond what is shown in Figure 2(b), there is very little change in the value of the cost function, despite making some change on the iterate and/or gradient level. The effect of rounding error means that although we see progress made, the function value may remain stagnant because of limitations in computer precision and because of the conditioning of the problem. The condition number of the Hessian $\kappa(\mathbf{S})$ can be used to indicate the accuracy we could be able to achieve. In our work, both the L63 and L96 problems are well-conditioned.

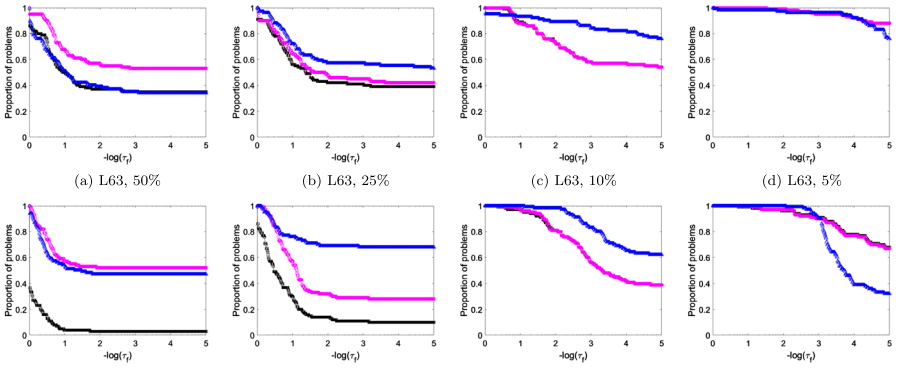


Figure 3. Data profiles for the GN (black squares), LS (magenta circles) and REG (blue triangles) methods applied to the L63 problems in (a)–(d) and the L96 problems in (e)–(h) where $n_r = 100$, $\tau_e = 8$ and where there is one observation at the end of the time-window. The observation error is 10% and the background error is varied above and below this, as indicated in the plot captions. The GN line is below the LS line in (c), (d), (g) and (h). (a) L63, 50% (b) L63, 25% (c) L63, 10% (d) L63, 5% (e) L96, 50% (f) L96, 25% (g) L96, 10% (h) L96, 5%.

The observed behavior in this section is partly due to the fact that there is no mechanism in GN to force it to converge as there is in LS and REG. The benefit of these mechanisms is clearly shown in Figure 2(b) where the GN method is stagnating while the LS and REG methods are converging, further motivating our investigation of these methods.

5.3. *Effect of background error variance*

In this section, we study the effect on the performance of the three methods when the uncertainty in the background information is increased whilst the uncertainty in the observations is fixed. Figure 3 shows the data profiles used to benchmark the performance of the GN, LS and REG methods as the tolerance τ_f is reduced, where $\tau_e = 8$, while Figure 4 allows τ_e to increase for both models with the increase chosen relative to the dimension of the models, i.e. a larger increase in τ_e is allowed for the L63 problems, where $n = 3$, than the L96 problems, where $n = 40$. From both these figures, we generally see that as the error in the background is reduced, the performance of all three methods improves. The conditioning of the problem has been shown to depend on the ratio of the standard deviations of the background and observation errors (23) [43,44]. Therefore, the estimate obtained by any of the optimization methods may not be accurate enough to produce a reliable forecast if the ratio (23) is large. The accuracy of the estimate obtained by each method will be investigated further later on in the paper.

Figure 3(a,e) show that a globally convergent method is able to find a smaller function value than GN. As the ratio (23) is reduced, from Figure 3(b,c,f,g) we

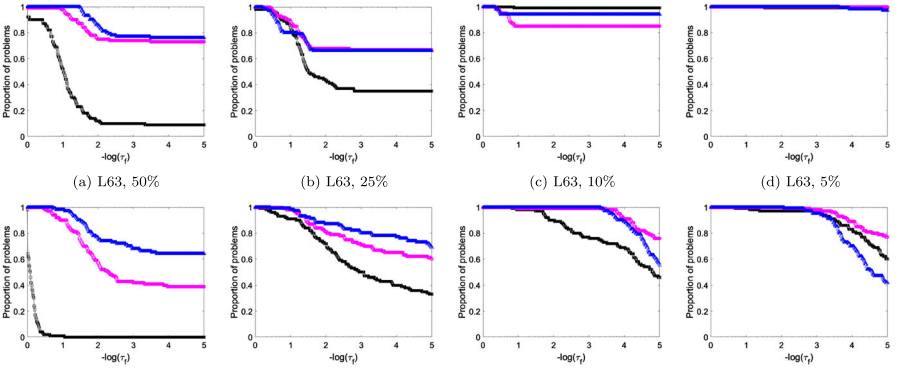


Figure 4. Data profiles for the GN (black squares), LS (magenta circles) and REG (blue triangles) methods applied to the L63 problems where $\tau_e = 1000$ in (a)–(d) and the L96 problems where $\tau_e = 100$ in (e)–(h). We set $n_r = 100$ and there is one observation at the end of the time-window. The observation error is 10% and the background error is varied above and below this, as indicated in the plot captions. (a) L63, 50% (b) L63, 25% (c) L63, 10% (d) L63, 5% (e) L96, 50% (f) L96, 25% (g) L96, 10% (h) L96, 5%.

see that the REG method is able to solve the most problems at the highest level of accuracy. When there is less uncertainty in the background versus the observations, Figure 3(d) shows that for the L63 problems, all three methods are solving close to all of the problems within a high level of accuracy. This is because the three methods are able to solve a large portion of the cases when the problem is well-conditioned, which could explain this result. However, for the L96 problems Figure 3(h) shows that the GN and LS methods are solving the majority of the problems and REG is not performing as well at higher levels of background accuracy. We can see the performance of REG improving for the L96 problems when more evaluations are allowed in Figure 4(h).

In Figure 4, where more evaluations are allowed than in Figure 3, we see a much greater difference between the globally convergent methods and GN when the background error is larger than the observation error. In Figure 4(a,b,e,f), it appears that when more evaluations are allowed, the performance of GN worsens relative to LS and REG in the case when σ_b is large. The globally convergent methods are able to locate estimates of the initial states for the preconditioned 4D-Var problem, which when compared to GN, better minimize the objective function (4). When the background error is the same as the observation error in Figure 4(c), it is GN that is performing better than LS and REG for the L63 problems. For LS, this could be because LS is unnecessarily shortening the GN step, causing slower convergence. For the REG method, the regularization parameter must be shrunk and therefore, REG requires more iterations to benefit from GN's fast convergence property.

In Figure 4(d), all three methods are solving essentially the same number of problems, with a slight decrease in success for REG, that again could be due to

the need to adjust the regularization parameter. For the L96 problems, we see a slightly different result. Figure 4(g-h) show that a globally convergent method is solving more problems, more accurately than GN despite the background error being at most equal to the observation error. This is an interesting result for this higher-dimensional model as we would expect GN to locally converge at a faster rate than the globally convergent methods due to the fact that GN does not need to adjust any parameters; however, we find this not to be the case.

In DA, we are interested in knowing the accuracy of the estimate obtained as in applications such as NWP, the estimate is used as the initial conditions for a forecast and so the quality of this forecast will depend on the errors in the estimate. In the following section, we quantify and compare the errors in the estimates obtained by each method.

5.4. Quality of the analysis

We recall that the initial guess of the algorithms is the reference state $\mathbf{x}_0^{ref}$ perturbed by the background error $\varepsilon_{\mathbf{b}}$. In order to compare the quality of the estimate obtained by each method, we compare their estimate to the reference state $\mathbf{x}_0^{ref}$ to understand how far the estimates obtained by the methods have deviated from this. The analysis error for each state variable is given by $\varepsilon_i^a = x_i^a - x_i^{ref}$. For each realization, we calculate the root mean square error (RMSE) of the analysis error, which is the difference between the reference state and the estimate obtained by each method,

$$\text{RMSE} = \frac{1}{\sqrt{n}} \|\varepsilon^a\|_2. \tag{28}$$

For each method, we plot the percentage of problems solved (according to the criterion (27) where $\tau_f = 10^{-3}$) within a specified tolerance of the RMSE (28). We acknowledge in this work that the code for the RMSE profiles has been adapted from the code for the data profiles used in [54].

The results for the L63 and L96 problems are in Figure 5, which coincides with the case shown in Figure 3 where $\tau_f = 10^{-3}$. From this, we see that the GN method solves fewer problems within the same level of RMSE accuracy as LS and REG when the background error is large in Figure 5(a,b,e,f). Furthermore, we see how the RMSE of the analyzes successfully found by each method reduces as the background error variance is reduced. This can be seen in the scale of the x axis in Figure 5(a–d) for the L63 problems and Figure 5(e–h) for the L96 problems. For both models, the concentration of points in Figure 5(a,e) shows us that the LS method is solving more problems than GN and REG within the same RMSE tolerance. A similar result can be seen for REG in Figure 5(b,c,f,g). In Figure 5(d,h), we see that all three methods are performing similarly, the RMSE errors for each of the analyzes are very close together.

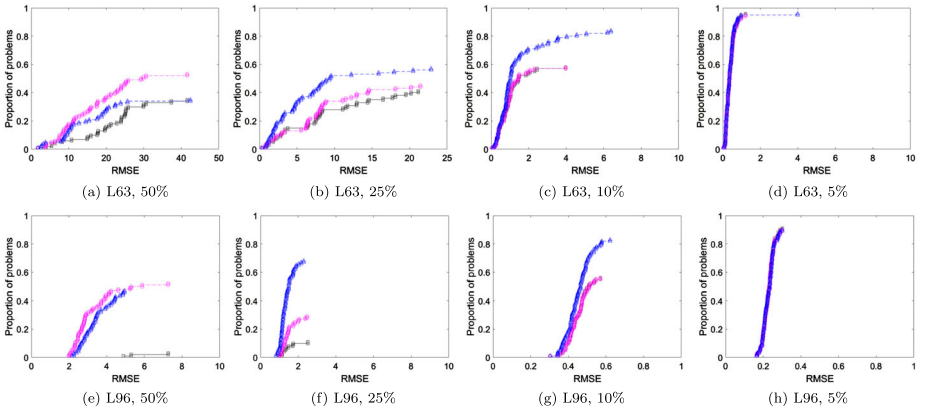


Figure 5. RMSE plots for the GN (black squares), LS (magenta circles) and REG (blue triangles) methods applied to the L63 problems in (a)–(d) and the L96 problems in (e)–(h) where $n_r = 100$, $\tau_e = 8$, $\tau_f = 10^{-3}$ and where there is one observation at the end of the time-window. The observation error is 10% and the background error is varied above and below this, as indicated in the plot captions. (a) L63, 50% (b) L63, 25% (c) L63, 10% (d) L63, 5% (e) L96, 50% (f) L96, 25% (g) L96, 10% (h) L96, 5%.

We next use RMSE profiles to understand the effect of using an inexact solver in the inner loop on the accuracy of the three methods when applied to the L63 and L96 problems. We consider the worst case for the three methods where there is a 50% error in the background. We use the CG method, where we set the desired CG tolerance to be $\tau_{cg} = 10^{-4}$, 10^{-3} or 10^{-2}. For each realization, we set the truth to be the minimum function value found by GN, LS and REG where the inner loop problem is solved exactly and classify whether the problem is solved according to the criterion (27). The results for the L63 and L96 problems are in Figure 6.

From Figure 5(a), we found that the LS method outperformed GN and REG for the L63 problems. This is also the case for all levels of inexactness in Figure 6(a–c). Furthermore, in these L63 figures we find that for the majority of cases, as the accuracy with which the inner problem is solved reduces, the proportion of problems solved by each of GN, LS and REG also reduces. This result is to be expected as it is known that the accuracy with which the inner loop is solved is known to affect the convergence of the outer loop [23–25]. Therefore, we expect the outer loop output to differ across the four levels of inexactness that we plot.

Figure 6(b) shows that the inexact LS method with $\tau_{cg} = 10^{-4}$ performs marginally better than the exact LS method. This may be because although the LS step calculation (which is the same as the GN step) differs depending on whether an exact or inexact solver is used, the LS method is able to adjust the step and thus improve convergence. This is not the case for GN, which does not have strategies to alter the length of step and REG can only do so implicitly through the REG parameter update. Therefore, it is possible that the LS method with an inexact

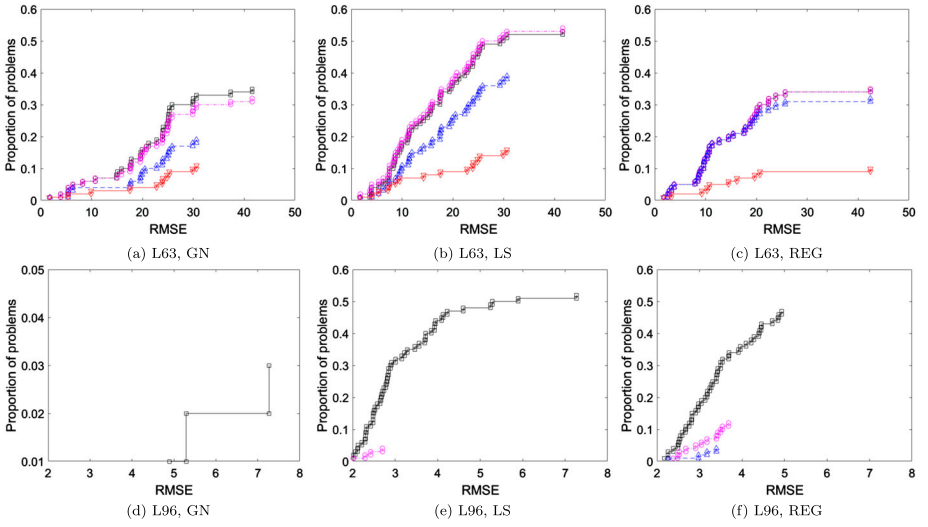
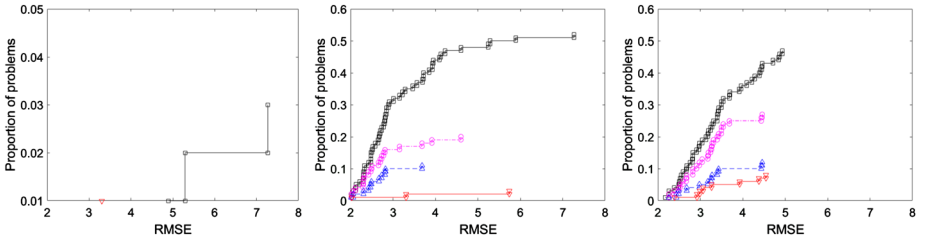


Figure 6. RMSE plots for the GN, LS and REG methods applied to the L63 problems in (a)–(c) and the L96 problems in (d)–(f) where $n_r = 100$, $\tau_e = 8$, $\tau_f = 10^{-3}$ and where there is one observation at the end of the time-window. The plots corresponding to a given problem and method (indicated in the plot captions) show the RMSE results where the inner loop is either solved by an exact method (black squares) or an inexact solver (CG) where $\tau_{cg} = 10^{-4}$ (magenta circles), $\tau_{cg} = 10^{-3}$ (blue triangles) or $\tau_{cg} = 10^{-2}$ (red downward-pointing triangles). The background error is 50% and the observation error is 10%. Note that a shorter range and finer scale is used for the y-axis in (d). (a) L63, GN (b) L63, LS (c) L63, REG (d) L96, GN (e) L96, LS (f) L96, REG.

solver is able to adjust the step such that it converges to a more accurate solution than when using an exact solver.

The results for the L96 problems given in Figure 6(d–f) differ from those of the L63 problems. More specifically, in Figure 6(d) we find that the minimum function value located by GN did not satisfy criterion (27) with $\tau_f = 10^{-3}$ for all choices of the CG tolerance, τ_{cg}, that we consider. This is to be expected as even when using an exact solver, GN was only able solve relatively few problems to the accuracy of $\tau_f = 10^{-3}$. Therefore, as we expect the number of problems solved to degrade as the level of inexactness increases, it is reasonable to find that zero problems are solved when using an inexact solver. This was also the case for LS where $\tau_{cg} = 10^{-3}$ and 10^{-2} as well as REG for $\tau_{cg} = 10^{-2}$. In addition we find that although LS was the preferred method when using an exact solver (see Figure 5(e)), Figure 6(d–f) indicate that it is REG that is the preferred method in the inexact case.

As it was difficult to compare the L96 results when $\tau_f = 10^{-3}$, we repeat the L96 experiments for when we increase τ_f to 10^{-2} to better understand the effect of using an inexact solver in the inner loop on the accuracy of the three methods. These results are given in Figure 7. Figure 7(a) shows GN performing poorly for all levels of inexactness. However, one problem appears to be solved when $\tau_{cg} = 10^{-2}$, although this is still relatively poor compared to the performance of



(a) L96, GN (b) L96, LS (c) L96, REG

Figure 7. RMSE plots for the GN, LS and REG methods applied to the L96 problems in where $n_r = 100$, $\tau_e = 8$, $\tau_f = 10^{-2}$ and where there is one observation at the end of the time-window. The plots corresponding to a given problem and method (indicated in the plot captions) show the RMSE results where the inner loop is either solved by an exact method (black squares) or an inexact solver (CG) where $\tau_{cg} = 10^{-4}$ (magenta circles), $\tau_{cg} = 10^{-3}$ (blue triangles) or $\tau_{cg} = 10^{-2}$ (red downward-pointing triangles). The background error is 50% and the observation error is 10%. Note that a shorter range and finer scale is used for the y-axis in (a). (a) L96, GN (b) L96, LS (c) L96, REG.

LS and REG shown in Figure 7(b-c), respectively. In these figures we find that, as with the L63 results, as the accuracy with which the inner problem is solved reduces, the proportion of problems solved by LS and REG also reduces. The LS method is preferred in the exact case, but it is REG that is the preferred method in the inexact case, as indicated by the L96 results where $\tau_f = 10^{-3}$.

In this section, we studied how the quality of the analysis is affected when using GN, LS and REG and also considered the effects of an inexact solver on the performance of the three methods. Including more observations constrains the solution to be closer to the reference state when the observation error is small. We next show the effect on the performance of the methods as we include more observations and see if this gives any improvement in the performance of the methods when the background error is much larger than the observation error.

5.5. Effect of observations

Within this section, we show how the use of more observation locations in time affects the performance of the three methods. We take the worst case for the three methods when there is a 50% error in the background and see if including more observations in time with a 10% error affects the performance of the methods. For both models, we consider only equally spaced observations in time, one set of observations at time N (Nobs1), times $N/2$ and N (Nobs2), times $N/4, N/2, 3N/4$ and N (Nobs3) and the even time points (Nobs4), where $N = 40$. For the Nobs1 case, observations are based on the reference state at the end of the time-window and more observations are included over time in the Nobs2, Nobs3 and Nobs4 cases. This not only increases the condition number of the problem but also constrains the estimate more tightly to the reference state.

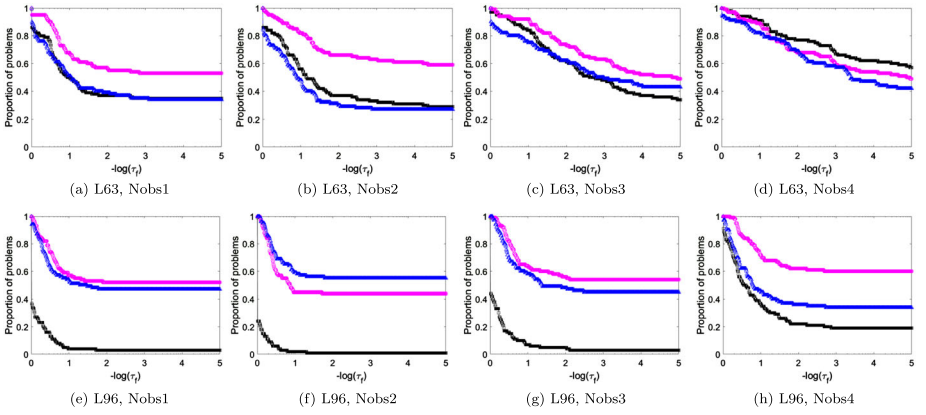


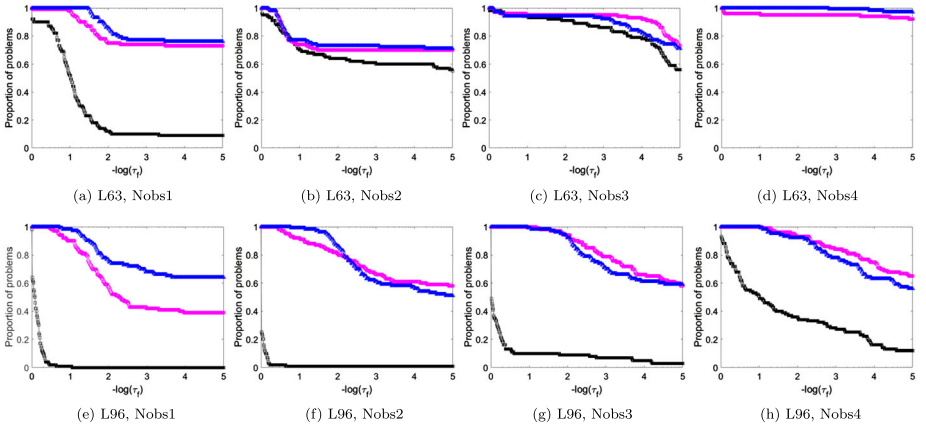
Figure 8. Data profiles where $n_r = 100$ and $\tau_e = 8$ for the L63 problems in (a)–(d) and the L96 problems in (e)–(h) for different observation locations in time, as indicated in the plot captions, where the background error is 50% and the observation error is 10%. (a) L63, Nobs1 (b) L63, Nobs2 (c) L63, Nobs3 (d) L63, Nobs4 (e) L96, Nobs1 (f) L96, Nobs2 (g) L96, Nobs3 (h) L96, Nobs4.

For the L63 problems from Figure 8(a–d), we see that as the number of observation locations in time is increased, all three methods are solving more problems at a higher level of accuracy. This is more apparent when more evaluations are allowed as shown in Figure 9(a–d). Here, the performance of GN improves drastically between the Nobs1 and Nobs2 cases (Figure 8(a-b)) while there is less significant change in the behavior of LS and REG. In Figure 8(d), we see that GN is able to solve more problems than LS and REG. Again, this could be because the LS and REG methods require more iterations to converge when GN is performing well due to the need to adjust their parameters. This argument coincides with Figure 9(d) where more evaluations are allowed and the LS and REG methods are able to perform as well as or better than GN. For the L96 problems, we see a different result. From Figure 8, we only see a significant improvement in the performance of GN in the Nobs4 case (Figure 8(h)). Otherwise, there is little effect. This conclusion can also be drawn from Figure 9(g-h) where more evaluations are allowed.

Similar studies were carried out on the performance of GN, LS and REG when applied to the preconditioned 4D-Var problem where we instead choose $\mathbf{B} = \sigma_b^2 \mathbf{C}_B$, where $\mathbf{C}_B$ is a correlation matrix; similar conclusions are drawn but due to space constraints, are not included within this paper.

6. Conclusion

We have shown that the globally convergent methods, LS and REG, have the capacity to improve current estimates of the DA analysis within the limited time and cost available in DA, through the use of safeguards within GN which guarantee the convergence of the method from any initial guesses.



(a) L63, Nobs1 (b) L63, Nobs2 (c) L63, Nobs3 (d) L63, Nobs4

(e) L96, Nobs1 (f) L96, Nobs2 (g) L96, Nobs3 (h) L96, Nobs4

Figure 9. Data profiles where $n_r = 100$ for the L63 problems where $\tau_e = 1000$ in (a)–(d) and the L96 problems where $\tau_e = 100$ in (e)–(h) for different observation locations in time, as indicated in the plot captions, where the background error is 50% and the observation error is 10%. The GN line is below the LS line in (d). (a) L63, Nobs1 (b) L63, Nobs2 (c) L63, Nobs3 (d) L63, Nobs4 (e) L96, Nobs1 (f) L96, Nobs2 (g) L96, Nobs3 (h) L96, Nobs4.

Using the L63 and L96 models in the preconditioned 4D-Var framework, we have shown that when there is more uncertainty in the background information compared to the observations, the GN method may fail to converge in the long time-window case yet the globally convergent methods LS and REG are able to improve the estimate of the initial state. We compare the quality of the estimate obtained using the RMSE of the analysis and show that even in the case where the background is highly inaccurate compared to the observations, the globally convergent methods find estimates with an RMSE less than or equal to the RMSE of the estimates GN obtains. We take the case where the background is highly inaccurate compared to the observations and find that, even when using an inexact solver, the globally convergent methods outperform GN. Furthermore, we find that the convergence of all three methods is improved when more observations are included along the time-window. In addition to the numerical results, the assumptions made in the global convergence theorems of both LS and REG when applied to a general nonlinear least-squares problem and a discussion as to whether these assumptions are satisfied in DA is presented in the appendix. We note that preconditioning the second derivative matrix is not necessary for these results to hold, although this is the case we have focused on within our work.

Our findings are important in DA as they show that in cases where the accuracy of the prior information is poor and when there is limited computational budget, the globally convergent methods are able to minimize the 4D-Var objective function, unlike GN. We recommend that these methods are tested on DA problems with realistic models and for different applications to understand if these conclusions continue to hold. In particular, one should consider such problems where an accurate initial guess for the algorithms is unavailable and a long

assimilation time-window is used, as we found that it is in this case that LS and REG have an advantage over GN.

Within this paper, the 4D-Var inner loop problem is solved exactly, except in Section 5.4, where we consider the effect of using an inexact solver. In practice it must be solved inexactly, due to the size of the control vector, and by the use of approximations to meet the computational and time constraints. This is a common area of research in the DA community in order to improve the quality of the assimilation analysis as well as the speed of convergence of the algorithms. Furthermore, in the case where GN performs better than LS and REG, further research is needed on updating the globalization parameters (stepsize $\alpha^{(k)}$ and regularization parameter $\gamma^{(k)}$) to speed up convergence.

Acknowledgments

We acknowledge in this work that the code for the Lorenz 1996 model was developed by Adam El-Said.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work has been funded in part by the UK Engineering and Physical Sciences Research Council Centre for Doctoral Training in Mathematics of Planet Earth, the University of Reading EPSRC studentship (Award Number: EP/N509723/1) and by the NERC National Centre for Earth Observation.

References

[1] Le Dimet F-X, Talagrand O. Variational algorithms for analysis and assimilation of meteorological observations: theoretical aspects. Tellus A: Dynamic Meteorology and Oceanography. 1986;38(2):97–110. doi: 10.3402/tellusa.v38i2.11706

[2] Nichols NK. Mathematical concepts of data assimilation. In: Lahoz W, Swinbank R, Khattatov B, editors. Data assimilation: making sense of observations. Berlin, Heildelberg: Springer; 2010. p. 13–39.

[3] Rabier F. Overview of global data assimilation developments in numerical weather-prediction centres. Q J R Meteorolog Soc. 2005;131(613):3215–3233. doi: 10.1256/qj.05.129

[4] Gauthier P, Tanguay M, Laroche S, et al. Extension of 3DVAR to 4DVAR: implementation of 4DVAR at the meteorological service of Canada. Monthly Weather Rev. 2007;135(6):2339–2354. doi: 10.1175/MWR3394.1

[5] Courtier P, Thépaut J-N, Hollingsworth A. A strategy for operational implementation of 4D-Var, using an incremental approach. Q J Royal Meteorolog Soc. 1994;120(519):1367–1387.

[6] Rabier F, Thépaut J-N, Courtier P. Extended assimilation and forecast experiments with a four-dimensional variational assimilation system. Q J Royal Meteorolog Soc. 1998;124(550):1861–1887.

[7] Rawlins F, Ballard S, Bovis K, et al. The met office global four-dimensional variational data assimilation scheme. Q J Royal Meteorolog Soc. 2007;133(623):347–362. doi: 10.1002/qj.v133:623
[8] Gilbert JC, Lemaréchal C. Some numerical experiments with variable-storage Quasi-Newton algorithms. Math Program. 1989;45(1-3):407–435. doi: 10.1007/BF01589113
[9] Liu DC, Nocedal J. On the limited memory BFGS method for large scale optimization. Math Program. 1989;45(1-3):503–528. doi: 10.1007/BF01589116
[10] Shanno DF, Phua K-H. Remark on 'Algorithm 500: minimization of unconstrained multivariate functions [e4]'. ACM Trans Math Softw (TOMS). 1980;6(4):618–622. doi: 10.1145/355921.355933
[11] Zou X, Navon IM, Berger M, et al. Numerical experience with limited-memory quasi-Newton and truncated Newton methods. SIAM J Optim. 1993;3(3):582–608. doi: 10.1137/0803029
[12] Dembo RS, Eisenstat SC, Steihaug T. Inexact Newton methods. SIAM J Numer Anal. 1982;19(2):400–408. doi: 10.1137/0719025
[13] Le Dimet F-X, Navon IM, Daescu DN. Second-order information in data assimilation. Monthly Weather Rev. 2002;130(3):629–648. doi: 10.1175/1520-0493(2002)130 < 0629:SOIIDA > 2.0.CO;2
[14] Wang Z, Navon I, Zou X, et al. A truncated Newton optimization algorithm in meteorology applications with analytic Hessian/vector products. Comput Optim Appl. 1995;4(3):241–262. doi: 10.1007/BF01300873
[15] Wang Z, Droegemeier K, White L. The adjoint Newton algorithm for large-scale unconstrained optimization in meteorology applications. Comput Optim Appl. 1998;10(3):283–320. doi: 10.1023/A:1018321307393
[16] Daescu DN, Navon IM. An analysis of a hybrid optimization method for variational data assimilation. Int J Comut Fluid Dyn. 2003;17(4):299–306. doi: 10.1080/1061856031000120510
[17] Dennis Jr JE, Schnabel RB. Numerical methods for unconstrained optimization and nonlinear equations. Philadelphia (PA): SIAM; 1996.
[18] Nocedal J, Wright SJ. Numerical optimization. 2nd ed. New York (NY): Springer; 2006.
[19] Gratton S, Lawless AS, Nichols NK. Approximate Gauss–Newton methods for nonlinear least squares problems. SIAM J Optim. 2007;18(1):106–132. doi: 10.1137/050624935
[20] Gratton S, Laloyaux P, Sartenaer A. Derivative-free optimization for large-scale nonlinear data assimilation problems. Q J Royal Meteorolog Soc. 2014;140(680):943–957. doi: 10.1002/qj.2014.140.issue-680
[21] Bannister R. A review of operational methods of variational and ensemble-variational data assimilation. Q J R Meteorolog Soc. 2017;143(703):607–633. doi: 10.1002/qj.2017.143.issue-703
[22] Liu C, Xiao Q, Wang B. An ensemble-based four-dimensional variational data assimilation scheme. Part I: technical formulation and preliminary test. Monthly Weather Rev. 2008;136(9):3363–3373. doi: 10.1175/2008MWR2312.1
[23] Lawless AS, Gratton S, Nichols NK. An investigation of incremental 4D-Var using non-tangent linear models. Q J R Meteorolog Soc. 2005;131(606):459–476. doi: 10.1256/qj.04.20
[24] Laroche S, Gauthier P. A validation of the incremental formulation of 4D variational data assimilation in a nonlinear barotropic flow. Tellus A: Dynamic Meteorology and Oceanography. 1998;50(5):557–572. doi: 10.3402/tellusa.v50i5.14558
[25] Lawless AS, Nichols NK. Inner-loop stopping criteria for incremental four-dimensional variational data assimilation. Monthly Weather Rev. 2006;134(11):3425–3435. doi: 10.1175/MWR3242.1

[26] Bannister RN. A review of forecast error covariance statistics in atmospheric variational data assimilation. II: modelling the forecast error covariance statistics. Q J R Meteorolog Soc. 2008;134(637):1971–1996. doi: 10.1002/qj.v134:637
[27] Conn AR, Gould NI, Toint PL. Trust region methods. Philadelphia (PA): SIAM; 2000.
[28] Levenberg K. A method for the solution of certain non-linear problems in least squares. Q Appl Math. 1944;2(2):164–168. doi: 10.1090/qam/1944-02-02
[29] Marquardt DW. An algorithm for least-squares estimation of nonlinear parameters. J Soc Ind Appl Math. 1963;11(2):431–441. doi: 10.1137/0111030
[30] Cartis C, Gould NI, Toint PL. Adaptive cubic regularisation methods for unconstrained optimization. Part I: motivation, convergence and numerical results. Math Program. 2011;127(2):245–295. doi: 10.1007/s10107-009-0286-5
[31] Cartis C, Gould NI, Toint PL. Adaptive cubic regularisation methods for unconstrained optimization. Part II: worst-case function-and derivative-evaluation complexity. Math Program. 2011;130(2):295–319. doi: 10.1007/s10107-009-0337-y
[32] Trémolet Y. Incremental 4D-Var convergence study. Tellus A: Dynamic Meteorology and Oceanography. 2007;59(5):706–718. doi: 10.1111/j.1600-0870.2007.00271.x
[33] Lorenc A, Ballard S, Bell R, et al. The met. Office global three-dimensional variational data assimilation scheme. Q J R Meteorolog Soc. 2000;126(570):2991–3012.
[34] Buehner M, Morneau J, Charette C. Four-dimensional ensemble-variational data assimilation for global deterministic weather prediction. Nonlinear Process Geophys. 2013;20(5):669–682. doi: 10.5194/npg-20-669-2013
[35] Wolfe P. Convergence conditions for ascent methods. SIAM Rev. 1969;11(2):226–235. doi: 10.1137/1011036
[36] Armijo L. Minimization of functions having Lipschitz continuous first partial derivatives. Pacific J Math. 1966;16(1):1–3. doi: 10.2140/pjm
[37] Gratton S, Gürol S, Simon E, et al. Guaranteeing the convergence of the saddle formulation for weakly constrained 4D-Var data assimilation. Q J R Meteorolog Soc. 2018;144(717):2592–2602. doi: 10.1002/qj.2018.144.issue-717
[38] Bergou E, Gratton S, Vicente L. Levenberg–Marquardt methods based on probabilistic gradient models and inexact subproblem solution, with application to data assimilation. SIAM/ASA J Uncertainty Quantif. 2016;4(1):924–951. doi: 10.1137/140974687
[39] Bergou E, Diouane Y, Kungurtsev V, et al. A stochastic Levenberg-Marquardt method using random models with complexity results. Preprint arXiv:1807.02176, 2021.
[40] Mandel J, Bergou E, Gürol S, et al. Hybrid Levenberg-Marquardt and weak-constraint ensemble Kalman smoother method. Nonlinear Process Geophys. 2016;23(2):59. doi: 10.5194/npg-23-59-2016
[41] Bergou EH, Diouane Y, Kungurtsev V. Convergence and complexity analysis of a Levenberg–Marquardt algorithm for inverse problems. J Optim Theory Appl. 2020;185:927–944. doi: 10.1007/s10957-020-01666-1
[42] Gürol S. Solving regularized nonlinear least-squares problem in dual space [dissertation]. France: Université de Toulouse; 2013.
[43] Haben SA, Lawless AS, Nichols NK. Conditioning and preconditioning of the variational data assimilation problem. Comput Fluids. 2011;46(1):252–256. doi: 10.1016/j.compfluid.2010.11.025
[44] Haben SA, Lawless AS, Nichols NK. Conditioning of incremental variational data assimilation, with application to the met office system. Tellus A: Dynamic Meteorology and Oceanography. 2011;63(4):782–792. doi: 10.1111/j.1600-0870.2011.00527.x
[45] Golub GH, Van Loan CF. Matrix computations. Vol. 3. Baltimore: Johns Hopkins University Press; 2012.

[46] Moré JJ. The Levenberg-Marquardt algorithm: implementation and theory. In: Watson GA, editor. Numerical analysis. Berlin, Heidelberg: Springer; 1978. p. 105–116.
[47] Lorenz EN. Deterministic nonperiodic flow. J Atmospheric Sci. 1963;20(2):130–141. doi: 10.1175/1520-0469(1963)020 < 0130:DNF > 2.0.CO;2
[48] Lorenz EN. Predictability: A problem partly solved. In: Proceedings of the Seminar on Predictability, Vol. I. Reading: ECMWF; 1996.
[49] Beiranvand V, Hare W, Lucet Y. Best practices for comparing optimization algorithms. Optim Eng. 2017;18(4):815–848. doi: 10.1007/s11081-017-9366-1
[50] Fisher M, Nocedal J, Trémolet Y, et al. Data assimilation in weather forecasting: a case study in PDE-constrained optimization. Optim Eng. 2009;10(3):409–426. doi: 10.1007/s11081-008-9051-5
[51] MathWorks: mldivide documentation [accessed 2021 May 30]. Available at https://www.mathworks.com/help/matlab/ref/mldivide.html.
[52] MathWorks: pcg documentation [accessed 2022 Jan 10]. Available at https://uk.mathworks.com/help/matlab/ref/pcg.html.
[53] ECMWF. ECMWF Newsletter No.158 Winter 2018/19. (158):21–26, 2019.
[54] Moré JJ, Wild SM. Benchmarking derivative-free optimization algorithms. SIAM J Optim. 2009;20(1):172–191. doi: 10.1137/080724083
[55] Moodey AJ, Lawless AS, Potthast RW, et al. Nonlinear error dynamics for cycled data assimilation methods. Inverse Probl. 2013;29(2):025002. doi: 10.1088/0266-5611/29/2/025002

Appendix: Convergence theorems

In this section, we outline some existing global convergence results for the LS and REG methods and discuss whether the assumptions made hold in DA. We first state the definitions of a local and global minimum of an optimization problem $\min_{\mathbf{v}\in\mathbb{R}^n} f(\mathbf{v})$ where $f : \mathbb{R}^n \to \mathbb{R}$ and $\mathbf{v} \in \mathbb{R}^n$.

Definition A.1 (Local minimizer [18]): A point $\mathbf{v}^*$ is a local minimizer of f if there is a neighborhood $\mathcal{N}$ of $\mathbf{v}^*$ such that $f(\mathbf{v}^*) \leq f(\mathbf{v})$ for all $\mathbf{v} \in \mathcal{N}$.

Definition A.2 (Global minimizer [18]): A point $\mathbf{v}^*$ is a global minimizer of $f : \mathbb{R}^n \to \mathbb{R}$ if $f(\mathbf{v}^*) \leq f(\mathbf{v})$ for all $\mathbf{v} \in \mathbb{R}^n$.

A global solution is difficult to locate in most cases due to the nonlinearity of the problems. Therefore, a local solution is often sought by algorithms for nonlinear optimization.

We focus on nonlinear least-squares optimization problems of the form (5) for the remainder of this section. The GN method can only guarantee local convergence under certain conditions and not necessarily global convergence. This is dependent on how close the initial guess is from the local minimum the algorithm locates and whether or not the residual vector $\mathbf{r}$ of (5) is a zero vector at a solution $\mathbf{v}^*$. Furthermore, the region of local convergence depends on problem constants not known a priori, such as Lipschitz constants of the gradient.

A local convergence result for the GN method can be found in Theorem 10.2.1 of [17] where the performance of GN is shown to be dependent on whether or not the second-order terms in (9) evaluated at the solution $\mathbf{v}^*$ are close to zero. Another local convergence result can be found in Theorem 4 of [19] where GN is treated as an inexact Newton method. The theorem guarantees convergence of the GN method if for each iteration $k = 0, 1, \ldots$, the norm of the ratio of $\mathbf{Q}(\mathbf{v}^{(k)})$ and $\mathbf{J}(\mathbf{v}^{(k)})^T\mathbf{J}(\mathbf{v}^{(k)})$, the second and first terms of (9) respectively, is less than or equal to some constant $\hat{\eta}$ where $0 \leq \hat{\eta} \leq 1$.

It is important to note here that the globally convergent methods we are concerned with, namely LS and REG, can only guarantee global convergence to a local minimum under certain conditions and not necessarily to a global minimum.

Before we list the assumptions for the global convergence theorems, we first state the definition of the Lipschitz continuity property of a general function g as this is widely used in the theorems.

Definition A.3 (Lipschitz continuous function (see [18] A.42)): Let g be a function where $g : \mathbb{R}^n \to \mathbb{R}^m$ for general n and m. The function g is said to be Lipschitz continuous on some set $\mathcal{N} \subset \mathbb{R}^n$ if there exists a constant $L > 0$ such that,

$$\|g(\mathbf{v}) - g(\mathbf{w})\| \leq L\|\mathbf{v} - \mathbf{w}\|, \quad \forall\, \mathbf{v}, \mathbf{w} \in \mathcal{N}. \tag{A1}$$

The following assumptions are used to prove global convergence of both the LS and REG methods.

(A1) $\mathbf{r}$ is uniformly bounded above by $\omega > 0$ such that $\|\mathbf{r}(\mathbf{v})\| \leq \omega$.
(A2) $\mathbf{r} \in \mathcal{C}^1(\mathbb{R}^n)$ is Lipschitz continuous on $\mathbb{R}^n$ with Lipschitz constant $L_r > 0$.
(A3) $\mathbf{J}$ is Lipschitz continuous on $\mathbb{R}^n$ with Lipschitz constant $L_J > 0$.

We remark that for the LS method, we can weaken assumptions (A2) and (A3) using an open set $\mathcal{N}$ containing the level set

$$\mathcal{L} = \left\{\mathbf{v} \in \mathbb{R}^n \,|\, \mathcal{J}(\mathbf{v}) \leq \mathcal{J}(\mathbf{v}^{(0)})\right\}. \tag{A2}$$

In order to achieve the sufficient decrease property of the LS method, the following assumption is needed.

(A4) $\mathbf{J}(\mathbf{v})$ in (6) is uniformly full rank for all $\mathbf{v} \in \mathbb{R}^n$, that is, the singular values of $\mathbf{J}(\mathbf{v})$ are uniformly bounded away from zero, so there exists a constant ν such that $\|\mathbf{J}(\mathbf{v})\mathbf{z}\| \geq \nu\|\mathbf{z}\|$ for all $\mathbf{z} \in \mathbb{R}^n$, and all $\mathbf{v}$ in an open neighborhood $\mathcal{N}$ containing the level set $\mathcal{L}$.

In 4D-Var practice, it is reasonable to assume that the physical quantities are bounded. Therefore, we can say that both $\mathbf{x}_0 - \mathbf{x}^b$ and the innovation vector $\mathbf{y} - \mathcal{H}(\mathbf{x})$ are bounded in practice, thus satisfying assumption (A1). In 4D-Var, we must assume that the nonlinear model $\mathcal{M}_{0,i}$ is Lipschitz continuous in order for (A2) to hold. As discussed in [55], this is a common assumption made in the meteorological applications, where attractors are known to exist. However, we cannot say that this is necessarily the case in 4D-Var practice, although it is reasonable to assume that $\mathbf{M}_{0,i}$ is bounded, and hence that $\mathcal{M}_{0,i}$ is Lipschitz continuous.

In order for the Jacobian $\mathbf{J}$ to be Lipschitz continuous, we require its derivative to be bounded above by its Lipschitz constant. Therefore, for assumption (A3) to hold, we require $\mathbf{r}$ to be twice continuously differentiable in practice, which is a common assumption made in 4D-Var, and also, that these derivatives of $\mathbf{r}$ are bounded above.

As mentioned in Section 2, the preconditioned 4D-Var Hessian (10) is full rank by construction as it consists of the identity matrix and a non-negative definite term. Therefore, the Jacobian of the residual of the preconditioned problem in (6) is full rank and assumption (A4) holds. This is also the case for the standard 4D-Var problem (1), because of the presence of $\mathbf{B}^{1/2}$ in its Jacobian.

We now outline the global convergence theorems for the LS and REG methods, using these assumptions.

A.1 Global convergence of the LS method

A proof for the global convergence of GN method with Wolfe line search conditions can be found in [18], which uses the Zoutendijk condition. This proof can be adapted to prove the global convergence theorem of the LS method, Algorithm 2, given as follows.

Theorem A.1 (Global convergence for the Gauss-Newton with bArmijo line search method, Algorithm 2): *Suppose we have a function* $\mathcal{J} = \frac{1}{2}\mathbf{r}^T\mathbf{r}$ *and its gradient* $\nabla\mathcal{J} = \mathbf{J}^T\mathbf{r}$ *where* $\mathbf{r} \in \mathcal{C}^1(\mathbb{R}^n)$ *and* $\mathbf{J}$ *is the Jacobian of* $\mathbf{r}$. *Assume (A1)–(A4) hold. Then if the iterates* $\{\mathbf{v}^{(k)}\}$ *are generated by the GN method with stepsizes* $\alpha^{(k)}$ *that satisfy the Armijo condition (13), we have*

$$\lim_{k\to\infty} \mathbf{J}(\mathbf{v}^{(k)})^T\mathbf{r}(\mathbf{v}^{(k)}) = 0. \tag{A3}$$

That is, the gradient norms converge to zero, and so the Gauss-Newton method with bArmijo line search is globally convergent.

The proof of Theorem A.1 requires the bArmijo chosen stepsizes $\alpha^{(k)}$ to be bounded below, which can be derived using assumptions (A1)–(A3). Using this lower bound, as well as assumption (A4), we are able to prove the Zoutendijk condition (as in [18]) and its variant

$$\sum_{k\geq 0} \cos(\theta^{(k)})\|\nabla\mathcal{J}(\mathbf{v}^{(k)})\|_2\|\mathbf{s}^{(k)}\|_2 < \infty \tag{A4}$$

hold. Both the Zoutendijk condition and its variant (A4) use the angle between $\mathbf{s}^{(k)}$ (the GN search direction) and $-\nabla\mathcal{J}(\mathbf{v}^{(k)})$ (the steepest descent direction), $\theta^{(k)}$, which is given by

$$\cos(\theta^{(k)}) = \frac{(-\nabla\mathcal{J}(\mathbf{v}^{(k)}))^T\mathbf{s}^{(k)}}{\|\nabla\mathcal{J}(\mathbf{v}^{(k)})\|_2\|\mathbf{s}^{(k)}\|_2}. \tag{A5}$$

By showing that the angle is uniformly bounded away from zero with k, one can show that GN with line search is a globally convergent method.

We will next present the global convergence theorem for the REG method. The REG method has no sufficient decrease condition as in the LS method. Therefore, the use of the level set (A2) is not required. The assumptions for convergence are similar to the LS method aside from the requirement of $\mathbf{J}(\mathbf{v})$ being full rank.

A.2 Global convergence of the REG method

The global convergence theorem for the GN with quadratic regularization method, Algorithm 3, is given as follows.

Theorem A.2 (Global convergence for the Gauss-Newton with regularization method, Algorithm 3): *Suppose we have a function* $\mathcal{J} = \frac{1}{2}\mathbf{r}^T\mathbf{r}$ *and its gradient* $\nabla\mathcal{J} = \mathbf{J}^T\mathbf{r}$ *where* $\mathbf{r} \in \mathcal{C}^1(\mathbb{R}^n)$ *and* $\mathbf{J}$ *is the Jacobian of* $\mathbf{r}$. *Assume (A1)–(A3) hold. Then if the iterates* $\{\mathbf{v}^{(k)}\}$ *are generated by the Gauss-Newton with regularization method, we have that*

$$\lim_{k\to\infty} \mathbf{J}(\mathbf{v}^{(k)})^T\mathbf{r}(\mathbf{v}^{(k)}) = 0. \tag{A6}$$

That is, the gradient norms converge to zero, and so the Gauss-Newton method with regularization is globally convergent.

We first note that some adaptations of the lemmas from the global convergence proof of the Adaptive Regularisation algorithm using Cubics (ARC method) are used to prove

Theorem A.2, see [30,31]. We begin the proof by deriving an expression for the predicted model decrease in terms of the gradient. We require the use of an upper bound on $\gamma^{(k)}$, denoted as $\gamma_{\max}$, which is derived using a property of Lipschitz continuous gradients. We show that $\gamma^{(k)} \leq \gamma_{\max}$ for all $k \geq 0$ by first showing that if $\gamma^{(k)}$ is large enough, then we have a successful step so that $\gamma^{(k)}$ can stop increasing due to unsuccessful steps in Algorithm 3. We use the expression for $\gamma_{\max}$ to prove global convergence of the REG method under assumptions (A1)–(A3) by showing that the gradient norms converge to zero as we iterate.

Note that for both the LS and REG, if $\mathbf{r}(\mathbf{v}^{(k)}) \to 0$, i.e. (5) is a zero residual problem, then we have that (A3) and (A6) hold as $|\mathcal{J}(\mathbf{v}^{(k)})|$ is uniformly bounded. However, in practice the variational problem is not usually a zero residual problem.