Levenberg-Marquardt methods based on probabilistic gradient models and inexact subproblem solution, with application to data assimilation

El Houcine Bergou, Serge Gratton, Luis Vicente

Levenberg–Marquardt Methods Based on Probabilistic Gradient Models and Inexact Subproblem Solution, with Application to Data Assimilation

E. Bergou[†], S. Gratton[‡], and L. N. Vicente[§]

Abstract. The Levenberg–Marquardt algorithm is one of the most popular algorithms for the solution of nonlinear least squares problems. Motivated by the problem structure in data assimilation, we consider in this paper the extension of the classical Levenberg–Marquardt algorithm to the scenarios where the linearized least squares subproblems are solved inexactly and/or the gradient model is noisy and accurate only within a certain probability. Under appropriate assumptions, we show that the modified algorithm converges globally to a first order stationary point with probability one. Our proposed approach is first tested on simple problems where the exact gradient is perturbed with a Gaussian noise or only called with a certain probability. It is then applied to an instance in variational data assimilation where stochastic models of the gradient are computed by the so-called ensemble methods.

Key words. Levenberg–Marquardt method, nonlinear least squares, regularization, random models, inexactness, variational data assimilation, Kalman filter/smoother, ensemble Kalman filter/smoother

DOI. 10.1137/140974687

1. Introduction.

In this paper we are concerned with a class of nonlinear least squares problems for which the exact gradient is not available and replaced by a probabilistic or random model. Problems of this nature arise in several important practical contexts. One example is variational modeling for meteorology, such as 3DVAR and 4DVAR [7, 20], the dominant data assimilation least squares formulations used in numerical weather prediction centers worldwide. Here, ensemble methods, like those known by the abbreviations EnKF and EnKS (for ensemble Kalman filter and smoother, respectively) [10, 11], are used to approximate the data arising in the solution of the corresponding linearized least squares subproblems [22], in a way where the true gradient is replaced by an approximated stochastic gradient model. Other examples appear in the broad context of derivative-free optimization problems [6] where models of the objective function evaluation may result from, a possibly random, sampling procedure [1].

The Levenberg–Marquardt algorithm [12, 16] can be seen as a regularization of the Gauss–Newton method. A regularization parameter is updated at every iteration and indirectly

Funding: The research of the third author was supported by FCT under grants PTDC/MAT/116736/2010 and PEst-C/MAT/UI0324/2011.

[†]MaIAGE, INRA, Université Paris-Saclay, 78350 Jouy-en-Josas, France (el-houcine.bergou@jouy.inra.fr).
[‡]ENSEEIHT, INPT, B.P. 7122 31071, Toulouse Cedex 7, France (serge.gratton@enseeiht.fr).
[§]CMUC, Department of Mathematics, University of Coimbra, 3001-501 Coimbra, Portugal (lnv@mat.uc.pt).

controls the size of the step, making Gauss–Newton globally convergent (i.e., convergent to stationarity independently of the starting point). We found that the regularization term added to Gauss–Newton maintains the structure of the linearized least squares subproblems arising in data assimilation, enabling us to use techniques like ensemble methods while simultaneously providing a globally convergent approach.

However, the use of ensemble methods in data assimilation poses difficulties since it makes random approximations to the gradient. We thus propose and analyze a variant of the Levenberg–Marquardt method to deal with probabilistic gradient models. It is assumed that an approximation to the gradient is provided but only accurate with a certain probability. The knowledge of the probability of the error between the exact gradient and the model one, and in particular of its density function, can be used in our favor in the update of the regularization parameter.

Having in mind large-scale applications (as those arising from data assimilation), we then consider that the least squares subproblems formulated in the Levenberg–Marquardt method are only solved in some approximated way. The amount of inexactness in such approximated solutions (tolerated for global convergence) is rigorously quantified as a function of the regularization parameter, in a way that it can be used in practical implementations.

We organize this paper as follows. In section 2, a short introduction to the Levenberg–Marquardt method is provided. The new Levenberg–Marquardt method based on probabilistic gradient models is described in section 3. Section 4 addresses the inexact solution of the linearized least squares subproblems arising within Levenberg–Marquardt methods. We cover essentially two possibilities: conjugate gradients (CGs) and any generic inexact solution of the corresponding normal equations. The whole approach is shown to be globally convergent to first order critical points in section 5, in the sense that a subsequence of the true objective function gradients goes to zero with probability one. The proposed approach is numerically illustrated in section 6 with a simple problem, artificially modified to create (i) a scenario where the model gradient is a Gaussian perturbation of the exact gradient, and (ii) a scenario case where to compute the model gradient both exact/approximated gradient routines are available but the exact one (seen as expensive) is called only with a certain probability.

An application to data assimilation is presented in section 7 where the purpose is to solve the 4DVAR problem using the methodology described in this paper. For the less familiar reader, we start by describing the 4DVAR incremental approach (Gauss–Newton) and the ways to solve the resulting linearized least squares subproblems, in particular Kalman smoother and EnKS, the latter one leading to stochastic model gradients. We then show how our approach, namely, the Levenberg–Marquardt method based on probabilistic gradient models and an inexact subproblem solution, provides an appropriate framework for the application of the 4DVAR incremental approach using the EnKS method for the subproblems and finite differences for derivative approximation. Illustrative numerical results using the Lorenz–63 model as a forecast model are provided.

A discussion of conclusions and future improvements is given in section 8. Throughout this paper $\|\cdot\|$ will denote the vector or matrix ℓ_2-norm. The notation $[X;Y]$ will represent the concatenation of X and Y as in MATLAB syntax.

2. The Levenberg–Marquardt method. Let us consider the following general nonlinear least squares problem

$$\min_{x\in\mathbb{R}^n} \; f(x) = \frac{1}{2}\|F(x)\|^2, \tag{1}$$

where $F : \mathbb{R}^n \to \mathbb{R}^m$ is a (deterministic) vector-valued function, assumed continuously differentiable, and $m > n$. Our main probabilistic approach to deal with nonlinear least squares problems is derived having in mind a class of inverse problems arising from data assimilation, for which the function f to be minimized in (1) is of the form

$$\frac{1}{2}\left(\|x_0 - x_b\|^2_{B^{-1}} + \sum_{i=1}^{T}\|x_i - \mathcal{M}_i(x_{i-1})\|^2_{Q_i^{-1}} + \sum_{i=0}^{T}\|y_i - \mathcal{H}_i(x_i)\|^2_{R_i^{-1}}\right), \tag{2}$$

where $(x_0, \ldots, x_T)$ corresponds to x and where the operators $\mathcal{M}_i$ and $\mathcal{H}_i$ and the scaling matrices will be defined in section 7.

The Gauss–Newton method is an iterative procedure where at each point x_j a step is computed as a solution of the linearized least squares subproblem

$$\min_{s\in\mathbb{R}^n} \; \frac{1}{2}\|F_j + J_j s\|^2,$$

where $F_j = F(x_j)$ and $J_j = J(x_j)$ denotes the Jacobian of F at x_j. The subproblem has a unique solution if J_j has full column rank, and in that case the step is a descent direction for f. In the case of our target application problem (2), such a linearized least squares subproblem becomes

$$\begin{aligned}\min_{\delta x_0,\ldots,\delta x_T\in\mathbb{R}^n} \frac{1}{2}\Bigg(&\|x_0 + \delta x_0 - x_b\|^2_{B^{-1}} + \sum_{i=1}^{T}\|x_i + \delta x_i - \mathcal{M}_i(x_{i-1}) - \mathcal{M}_i'(x_{i-1})\delta x_{i-1}\|^2_{Q_i^{-1}} \\ &+ \sum_{i=0}^{T}\|y_i - \mathcal{H}_i(x_i) - \mathcal{H}_i'(x_i)\delta x_i\|^2_{R_i^{-1}}\Bigg),\end{aligned} \tag{3}$$

where $(\delta x_0, \ldots, \delta x_T)$ corresponds to s (and the other details are given in section 7).

The Levenberg–Marquardt method [12, 16] (see also [19]) was developed to deal with the rank deficiency of J_j and to provide a globalization strategy for Gauss–Newton. At each iteration it considers a step of the form $-(J_j^\top J_j + \gamma_j I)^{-1} J_j^\top F_j$, corresponding to the unique solution of

$$\min_{s\in\mathbb{R}^n} \; m_j(x_j + s) = \frac{1}{2}\|F_j + J_j s\|^2 + \frac{1}{2}\gamma_j^2\|s\|^2,$$

where γ_j is an appropriately chosen regularization parameter. See [18, Notes and References of Chapter 10] for a brief summary of theoretical and practical aspects regarding the Levenberg–Marquardt method.

The Levenberg–Marquardt method can be seen as a precursor of the trust-region method [5] in the sense that it seeks to determine when the Gauss–Newton step is applicable (in which

case the regularization parameter is set to zero) or when it should be replaced by a slower but safer gradient or steepest descent step (corresponding to a sufficiently large regularization parameter). The comparison with trust-region methods can also be drawn by looking at the square of the regularization parameter as the Lagrange multiplier of a trust-region subproblem of the form $\min_{s\in\mathbb{R}^n} (1/2)\|F_j + J_j s\|^2$ s.t. $\|s\| \leq \delta_j$, and in fact it was soon suggested in [17] to update the regularization parameter γ_j in the same form as the trust-region radius δ_j. For this purpose, one considers the ratio between the actual reduction $f(x_j) - f(x_j + s_j)$ attained in the objective function and the reduction $m_j(x_j) - m_j(x_j + s_j)$ predicted by the model, given by

$$\rho_j = \frac{f(x_j) - f(x_j + s_j)}{m_j(x_j) - m_j(x_j + s_j)}.$$

Then, if ρ_j is sufficiently greater than zero, the step is accepted and γ_j is possibly decreased (corresponding to "δ_j is possibly increased"). Otherwise the step is rejected and γ_j is increased (corresponding to "δ_j is decreased").

3. The Levenberg–Marquardt method based on probabilistic gradient models. We are interested in the case where we do not have exact values for the Jacobian J_j and the gradient $J_j^\top F_j$ (of the model $m_j(x_j + s)$ at $s = 0$), but rather approximations which we will denote by J_{m_j} and g_{m_j}. We are further interested in the case where these model approximations are built in some random fashion. We will then consider random models of the form M_j, where g_{M_j} and J_{M_j} are random variables, and use the notation $m_j = M_j(\omega_j)$, $g_{m_j} = g_{M_j}(\omega_j)$, and $J_{m_j} = J_{M_j}(\omega_j)$ for their realizations. Note that the randomness of the models implies the randomness of the current point $x_j = X_j(\omega_j)$ and the current regularization parameter $\gamma_j = \Gamma_j(\omega_j)$, generated by the corresponding optimization algorithm.

Thus, the model (where F_{m_j} represents an approximation to F_j)

$$\begin{aligned} m_j(x_j + s) - m_j(x_j) &= \frac{1}{2}\|F_{m_j} + J_{m_j} s\|^2 + \frac{1}{2}\gamma_j^2\|s\|^2 - \frac{1}{2}\|F_{m_j}\|^2 \\ &= g_{m_j}^\top s + \frac{1}{2} s^\top \left(J_{m_j}^\top J_{m_j} + \gamma_j^2 I\right) s \end{aligned}$$

is a realization of

$$M_j(X_j + s) - M_j(X_j) = g_{M_j}^\top s + \frac{1}{2} s^\top \left(J_{M_j}^\top J_{M_j} + \Gamma_j^2 I\right) s.$$

Note that we subtracted the order zero term to the model to avoid unnecessary terminology. Our subproblem then becomes

$$\min_{s\in\mathbb{R}^n} \; m_j(x_j + s) - m_j(x_j) = g_{m_j}^\top s + \frac{1}{2} s^\top \left(J_{m_j}^\top J_{m_j} + \gamma_j^2 I\right) s. \tag{4}$$

In our data assimilation applied problem (2), the randomness arises from the use of EnKS to approximately solve the linearized least squares subproblem (3). In fact, as we will see in section 7, quadratic models of the form $1/2(\|u\|^2_{(B^N)^{-1}} + \|Hu - \tilde{D}\|^2_{R^{-1}})$ will be realizations of random quadratic models $1/2(\|u\|^2_{\mathcal{B}^{-1}} + \|Hu - \tilde{\mathcal{D}}\|^2_{R^{-1}})$, where u corresponds to s, and where

it would be easy to see what are the realizations g_m and J_m and the corresponding random variables g_M and J_M.

We will now impose that the gradient models g_{M_j} are accurate with a certain probability regardless of the history $M_1, \ldots, M_{j-1}$. The accuracy is defined in terms of a multiple of the inverse of the square of the regularization parameter (as happens in [1] for trust-region methods based on probabilistic models where it is defined in terms of a multiple of the trust-region radius). As we will see later in the convergence analysis (since the regularization parameter is bounded from below), one can demand less here and consider just the inverse of a positive power of the regularization parameter.

Assumption 3.1. Given constants $\alpha \in (0, 2]$, $\kappa_{eg} > 0$, and $p \in (0, 1]$, the sequence of random gradient models $\{g_{M_j}\}$ is (p)-probabilistically κ_{eg}-first-order accurate, for corresponding sequences $\{X_j\}$, $\{\Gamma_j\}$, if the events

$$S_j = \left\{ \|g_{M_j} - J(X_j)^\top F(X_j)\| \leq \frac{\kappa_{eg}}{\Gamma_j^\alpha} \right\}$$

satisfy the following submartingale-like condition

$$p_j^* = P(S_j | \mathcal{F}_{j-1}^M) \geq p, \tag{5}$$

where $\mathcal{F}_j^M = \sigma(M_0, \ldots, M_{j-1})$ is the σ-algebra generated by $M_0, \ldots, M_{j-1}$.

Correspondingly, a gradient model realization g_{m_j} is said to be κ_{eg}-first-order accurate if

$$\|g_{m_j} - J(x_j)^\top F(x_j)\| \leq \frac{\kappa_{eg}}{\gamma_j^\alpha}.$$

The version of Levenberg–Marquart that we will analyze and implement takes a successful step if the ratio ρ_j between actual and predicted reductions is sufficiently positive (condition $\rho_j \geq \eta_1$ below). In such cases, and now deviating from classical Levenberg–Marquart methods and following [1], the regularization parameter γ_j is increased if the size of the gradient model is of the order of the inverse of γ_j squared (i.e., if $\|g_{m_j}\| < \eta_2/\gamma_j^2$ for some positive constant $\eta_2 > 0$). Another relevant distinction is that we necessarily decrease γ_j in successful iterations when $\|g_{m_j}\| \geq \eta_2/\gamma_j^2$. The algorithm is described below and generates a sequence of realizations for the above-mentioned random variables.

Algorithm 3.1 (Levenberg–Marquardt method based on probabilistic gradient models).

Initialization
Choose the constants $\eta_1 \in (0, 1)$, $\eta_2, \gamma_{\min} > 0$, $\lambda > 1$, and $0 < p_{\min} \leq p_{\max} < 1$. Select x_0 and $\gamma_0 \geq \gamma_{\min}$.
For $j = 0, 1, 2, \ldots$

1. Solve (or approximately solve) (4), and let s_j denote such a solution.
2. Compute $\rho_j = \frac{f(x_j) - f(x_j + s_j)}{m_j(x_j) - m_j(x_j + s_j)}$.
3. Make a guess p_j of the probability p_j^* given in (5) such that $p_{\min} \leq p_j \leq p_{\max}$.

If $\rho_j \geq \eta_1$, then set $x_{j+1} = x_j + s_j$ and

$$\gamma_{j+1} = \begin{cases} \lambda\gamma_j & \text{if } \|g_{m_j}\| < \eta_2/\gamma_j^2, \\ \max\left\{\frac{\gamma_j}{\lambda^{\frac{1-p_j}{p_j}}}, \gamma_{\min}\right\} & \text{if } \|g_{m_j}\| \geq \eta_2/\gamma_j^2. \end{cases}$$

Otherwise, set $x_{j+1} = x_j$ and $\gamma_{j+1} = \lambda\gamma_j$.

If exact gradients are used (in other words, if $g_{M_j} = J(X_j)^\top F(X_j)$), then one always has

$$p_j^* = P\left(0 \leq \frac{\kappa_{eg}}{\Gamma_j^\alpha} \middle| \mathcal{F}_{j-1}^M\right) = 1,$$

and the update of γ in successful iterations reduces to $\gamma_{j+1} = \max\{\gamma_j, \gamma_{\min}\}$ (when $\|g_{m_j}\| \geq \eta_2/\gamma_j^2$), as in the more classical deterministic-type Levenberg–Marquart methods. In general one should guess p_j based on the knowledge of the random error incurred in the application context. It is however pertinent to stress that the algorithm runs for any guess of $p_j \in (0, 1]$ such that $p_j \in [p_{\min}, p_{\max}]$.

4. Inexact solution of the linearized least squares subproblems. Step 1 of Algorithm 3.1 requires the approximate solution of subproblem (4). As in trust-region methods, there are different techniques to approximate the solution of this subproblem yielding a globally convergent step, and we will discuss three of them in this section. For the purposes of global convergence it is sufficient to compute a step s_j that provides a reduction in the model as good as the one produced by the so-called Cauchy step (defined as the minimizer of the model along the negative gradient or steepest descent direction $-g_{m_j}$).

4.1. A Cauchy step. The Cauchy step is defined by minimizing $m_j(x_j - tg_{m_j})$ when $t > 0$ and is given by

$$s_j^c = -\frac{\|g_{m_j}\|^2}{g_{m_j}^\top (J_{m_j}^\top J_{m_j} + \gamma_j^2 I) g_{m_j}} g_{m_j}. \tag{6}$$

The corresponding Cauchy decrease (on the model) is

$$m_j(x_j) - m_j(x_j + s_j^c) = \frac{1}{2} \frac{\|g_{m_j}\|^4}{g_{m_j}^\top (J_{m_j}^\top J_{m_j} + \gamma_j^2 I) g_{m_j}}.$$

Since $g_{m_j}^\top (J_{m_j}^\top J_{m_j} + \gamma_j^2 I) g_{m_j} \leq \|g_{m_j}\|^2 (\|J_{m_j}\|^2 + \gamma_j^2)$, we conclude that

$$m_j(x_j) - m_j(x_j + s_j^c) \geq \frac{1}{2} \frac{\|g_{m_j}\|^2}{\|J_{m_j}\|^2 + \gamma_j^2}.$$

The Cauchy step (6) is cheap to calculate as it does not require solving any system of linear equations. Moreover, the Levenberg–Marquart method will be globally convergent if it uses a step that attains a reduction in the model as good as a multiple of the Cauchy decrease. Thus we will impose the following assumption on the step calculation.

Assumption 4.1. For every step j and for all realizations m_j of M_j,

$$m_j(x_j) - m_j(x_j + s_j) \geq \frac{\theta_{fcd}}{2} \frac{\|g_{m_j}\|^2}{\|J_{m_j}\|^2 + \gamma_j^2}$$

for some constant $\theta_{fcd} > 0$.

Such an assumption asks from the step a very mild reduction on the model (a fraction of what a step along the negative gradient would achieve) and it can thus be seen as a sort of minimum first order requirement.

4.2. A truncated-CG step. Despite providing a sufficient reduction in the model and being cheap to compute, the Cauchy step is a particular form of steepest descent, which can perform poorly regardless of the step length. One can see that the Cauchy step depends on $J_{m_j}^\top J_{m_j}$ only in the step length. Faster convergence can be expected if the matrix $J_{m_j}^\top J_{m_j}$ also influences the step direction.

Since the Cauchy step is the first step of the CG method when applied to the minimization of the quadratic $m_j(x_j + s) - m_j(x_j)$, it is natural to propose to run CG further and stop only when the residual becomes relatively small. Since CG generates iterates by minimizing the quadratic over nested Krylov subspaces, and the first subspace is the one generated by g_{m_j} (see, e.g., [18, Theorem 5.2]), the decrease attained at the first CG iteration (i.e., by the Cauchy step) is kept by the remaining.

4.3. A step from inexact solution of normal equations. Another possibility to approximately solve subproblem (4) is to apply some iterative solver (not necessarily CG) to the solution of the normal equations

$$\left(J_{m_j}^\top J_{m_j} + \gamma_j^2 I\right) s_j = -g_{m_j}.$$

An inexact solution s_j^{in} is then computed such that

$$\left(J_{m_j}^\top J_{m_j} + \gamma_j^2 I\right) s_j^{in} = -g_{m_j} + r_j \tag{7}$$

for a relatively small residual r_j satisfying $\|r_j\| \leq \epsilon_j \|g_{m_j}\|$. For such sufficiently small residuals we can guarantee a Cauchy decrease.

Assumption 4.2. For some constants $\beta_{in} \in (0, 1)$ and $\theta_{in} > 0$, suppose that $\|r_j\| \leq \epsilon_j \|g_{m_j}\|$ and

$$\epsilon_j \leq \min\left\{\frac{\theta_{in}}{\gamma_j^\alpha}, \sqrt{\beta_{in} \frac{\gamma_j^2}{\|J_{m_j}\|^2 + \gamma_j^2}}\right\}.$$

Note that we only need the second bound on ϵ_j (see the above inequality) to prove the desired Cauchy decrease. The first bound on ϵ_j will be used later in the convergence analysis. The following result is proved in the appendix.

Lemma 4.1. *Under Assumption* 4.2, *an inexact step* s_j^{in} *of the form* (7) *achieves Cauchy decrease if it satisfies Assumption* 4.1 *with* $\theta_{fcd} = 2(1 - \beta_{in})$.

5. Global convergence to first order critical points. We start by stating that two terms, that later will appear in the difference between the actual and predicted decreases, have the right order accuracy in terms of γ_j. The proof is given in the appendix.

Lemma 5.1. *For the three steps proposed (Cauchy, truncated CG, and inexact normal equations), one has that*

$$\|s_j\| \leq \frac{2\|g_{m_j}\|}{\gamma_j^2}$$

and

$$|s_j^\top(\gamma_j^2 s_j + g_{m_j})| \leq \frac{4\|J_{m_j}\|^2\|g_{m_j}\|^2 + 2\theta_{in}\|g_{m_j}\|^2}{\min\{1, \gamma_{\min}^{2-\alpha}\}\gamma_j^{2+\alpha}}.$$

(Assumption 4.2 is assumed for the inexact normal equations step $s_j = s_j^{in}$.)

We proceed by describing the conditions required for global convergence.

Assumption 5.1. The function f is continuously differentiable in an open set containing $L(x_0) = \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}$ with Lipschitz continuous gradient on $L(x_0)$ and corresponding constant $\nu > 0$.

The Jacobian model is uniformly bounded, i.e., there exists $\kappa_{Jm} > 0$ such that $\|J_{m_j}\| \leq \kappa_{Jm}$ for all j.

The next result is a classical one and essentially says that the actual and predicted reductions match each other well for a value of the regularization parameter γ_j sufficiently large relatively to the size of the gradient model (which would correspond to a sufficiently small trust-region radius in trust-region methods).

Lemma 5.2. *Let Assumption 5.1 hold. Let also Assumption 4.2 hold for the inexact normal equations step $s_j = s_j^{in}$. If x_j is not a critical point of f and the gradient model g_{m_j} is κ_{eg}-first-order accurate, and if*

$$\gamma_j \geq \left(\frac{\kappa_j}{1-\eta_1}\right)^{\frac{1}{\alpha}} \quad \textit{with} \quad \kappa_j = \left(1 + \frac{\kappa_{Jm}^2}{\gamma_{\min}^2}\right)\frac{2\nu + \frac{2\kappa_{eg}}{\|g_{m_j}\|} + 2\theta_{in} + 8\kappa_{Jm}^2}{\min\{1, \gamma_{\min}^{2-\alpha}\}\theta_{fcd}},$$

then $\rho_j \geq \eta_1$.

Proof. Again we omit the indices j in the proof. Applying a Taylor expansion,

$$\begin{aligned}
1 - \frac{\rho}{2} &= \frac{m(x) - f(x) + f(x+s) - m(x+s) + m(x) - m(x+s)}{2[m(x) - m(x+s)]} \\
&= \frac{s^\top J(x)^\top F(x) + R - s^\top g_m - s^\top(J_m^\top J_m + \gamma^2 I)s - s^\top g_m}{2[m(x) - m(x+s)]} \\
&= \frac{R + (J(x)^\top F(x) - g_m)^\top s - s^\top(J_m^\top J_m)s - s^\top(\gamma^2 s + g_m)}{2[m(x) - m(x+s)]},
\end{aligned}$$

where $R \leq \nu\|s\|^2/2$.

Now, using Lemma 5.1, Assumptions 4.1 and 5.1, and $\gamma \geq \gamma_{\min}$,

$$
\begin{aligned}
1 - \frac{\rho}{2} &\leq \frac{\frac{\nu}{2}\|s\|^2 + \frac{\kappa_{eg}}{\gamma^\alpha}\|s\| + \|J_m\|^2\|s\|^2 - s^\top(\gamma^2 s + g)}{\frac{\theta_{fcd}\|g_m\|^2}{\|J_m\|^2+\gamma^2}} \\
&\leq \frac{\frac{2\nu\|g_m\|^2}{\gamma^4} + \frac{2\kappa_{eg}\|g_m\|}{\gamma^{2+\alpha}} + \frac{4\kappa_{Jm}^2\|g_m\|^2}{\gamma^4} + \frac{4\kappa_{Jm}^2\|g_m\|^2 + 2\|g_m\|^2\theta_{in}}{\min\{1,\gamma_{\min}^{2-\alpha}\}\gamma^{2+\alpha}}}{\frac{\theta_{fcd}\|g_m\|^2}{\gamma^2(\|J_m\|^2/\gamma_{\min}^2+1)}} \\
&\leq \frac{\left(1 + \frac{\kappa_{Jm}}{\gamma_{\min}^2}\right)\left(2\nu + \frac{2\kappa_{eg}}{\|g_m\|} + 2\theta_{in} + 8\kappa_{Jm}^2\right)}{\min\{1,\gamma_{\min}^{2-\alpha}\}\theta_{fcd}\gamma^\alpha} \leq \frac{\kappa}{\gamma^\alpha} \leq 1 - \eta_1.
\end{aligned}
$$

We have thus proved that $\rho \geq 2\eta_1 > \eta_1$. ∎

One now establishes that the regularization parameter goes to infinity, which corresponds to the trust-region radius going to zero in [1].

Lemma 5.3. *Let the second part of Assumption 5.1 hold (the uniform bound on J_{m_j}). For every realization of the Algorithm 3.1, $\lim_{j\to\infty}\gamma_j = \infty$.*

Proof. If the result is not true, then there exists a bound $B > 0$ such that the number of times that $\gamma_j < B$ happens is infinite. Because of the way γ_j is updated one must have an infinity of iterations such that $\gamma_{j+1} \leq \gamma_j$, and for these iterations one has $\rho_j \geq \eta_1$ and $\|g_{m_j}\| \geq \eta_2/B^2$. Thus,

$$
\begin{aligned}
f(x_j) - f(x_j + s_j) &\geq \eta_1[m_j(x_j) - m_j(x_j + s_j)] \\
&\geq \eta_1\left(\frac{\theta_{fcd}}{2}\frac{1}{\|J_m\|^2 + \gamma^2}\right)\|g_{m_j}\|^2 \\
&\geq \frac{\eta_1\theta_{fcd}}{2(\kappa_{Jm}^2 + B^2)}\left(\frac{\eta_2}{B^2}\right)^2.
\end{aligned}
$$

Since f is bounded from below by zero, the number of such iterations cannot be infinite, and hence we arrive at a contradiction. ∎

Now, if we assume that the gradient models are (p_j)-probabilistically κ_{eg}-first-order accurate, we can show our main global convergence result. First we will state an auxiliary result from the literature that will be useful for the analysis (see [8, Theorem 5.3.1] and [8, Exercise 5.3.1]).

Lemma 5.4. *Let G_j be a submartingale, in other words, a set of random variables which are integrable ($E(|G_j|) < \infty$) and satisfy $E(G_j|\mathcal{F}_{j-1}) \geq G_{j-1}$ for every j, where $\mathcal{F}_{j-1} = \sigma(G_0, \ldots, G_{j-1})$ is the σ-algebra generated by $G_0, \ldots, G_{j-1}$ and $E(G_j|\mathcal{F}_{j-1})$ denotes the conditional expectation of G_j given the past history of events $\mathcal{F}_{j-1}$.*

Assume further that there exists $M > 0$ such that $|G_j - G_{j-1}| \leq M < \infty$ for every j. Consider the random events $C = \{\lim_{j\to\infty} G_j$ exists and is finite$\}$ and $D = \{\limsup_{j\to\infty} G_j = \infty\}$. Then $P(C \cup D) = 1$.

Theorem 5.1. *Let Assumption 5.1 hold. Let also Assumption 4.2 hold for the inexact normal equations step $s_j = s_j^{in}$.*

Suppose that the gradient model sequence $\{g_{M_j}\}$ is (p_j)-probabilistically κ_{eg}-first-order accurate for some positive constant κ_{eg} (Assumption 3.1). Let $\{X_j\}$ be a sequence of random iterates generated by Algorithm 3.1. Then almost surely,

$$\liminf_{j\to\infty} \|\nabla f(X_j)\| = 0.$$

Proof. The proof follows the same lines as [1, Theorem 4.2]. Let

$$W_j = \sum_{i=0}^{j} \left(\frac{1}{p_i} 1_{S_i} - 1 \right),$$

where S_i is as in Assumption 3.1. Recalling $p_j^* = P(S_j|\mathcal{F}_{j-1}^M) \geq p_j$, we start by showing that $\{W_j\}$ is a submartingale:

$$E(W_j|\mathcal{F}_{j-1}^M) = W_{j-1} + \frac{1}{p_j} P(S_j|\mathcal{F}_{j-1}^M) - 1 \geq W_{j-1}.$$

Moreover, $\min\{1, 1/p_j - 1\} \leq |W_j - W_{j-1}| \leq \max\{(1-p_j)/p_j, 1\} \leq \max\{1/p_j, 1\} = 1/p_j$. Since $0 < p_{\min} \leq p_j \leq p_{\max} < 1$, one has $0 < \min\{1, 1/p_{\max} - 1\} \leq |W_j - W_{j-1}| \leq 1/p_{\min}$. Thus, from $0 < \min\{1, 1/p_{\max} - 1\} \leq |W_j - W_{j-1}|$, the event $\{\lim_{j\to\infty} W_j \text{ exists and is finite}\}$ has probability zero, and using Lemma 5.4 and $|W_j - W_{j-1}| \leq 1/p_{\min}$, one concludes that $P(\limsup_{j\to\infty} W_j = \infty) = 1$.

Suppose there exist $\epsilon > 0$ and j_1 such that, with positive probability, $\|\nabla f(X_j)\| \geq \epsilon$ for all $j \geq j_1$. Let now $\{x_j\}$ and $\{\gamma_j\}$ be any realization of $\{X_j\}$ and $\{\Gamma_j\}$, respectively, built by Algorithm 3.1. By Lemma 5.3, there exists j_2 such that $\forall j \geq j_2$

$$\gamma_j > b_\epsilon = \max\left\{ \left(\frac{2\kappa_{eg}}{\epsilon}\right)^{\frac{1}{\alpha}}, \left(\frac{2\eta_2}{\epsilon}\right)^{\frac{1}{2}}, \lambda^{\frac{p-1}{p}}\gamma_{\min}, \left(\frac{\kappa_\epsilon}{1-\eta_1}\right)^{\frac{1}{\alpha}} \right\}, \tag{8}$$

where

$$\kappa_\epsilon = \left(1 + \frac{\kappa_{Jm}^2}{\gamma_{\min}^2}\right) \frac{2\nu + \frac{4\kappa_{eg}}{\epsilon} + 2\theta_{in} + 8\kappa_{Jm}^2}{\min\{1, \gamma_{\min}^{2-\alpha}\}\theta_{fcd}}.$$

For any $j \geq j_0 = \max\{j_1, j_2\}$ two cases are possible.

If $1_{S_j} = 1$, then, from (8),

$$\|g_{m_j} - J(x_j)^\top F(x_j)\| \leq \frac{\kappa_{eg}}{\gamma_j^\alpha} < \frac{\epsilon}{2},$$

yielding $\|g_{m_j}\| \geq \epsilon/2$. From (8) we also have that $\|g_{m_j}\| \geq \epsilon/2 \geq \eta_2/\gamma_j^2$. On the other hand, Lemma 5.2, (8), and $\|g_{m_j}\| \geq \epsilon/2$ together imply that $\rho_j \geq \eta_1$. Hence, from this and step 3 of Algorithm 3.1, the iteration is successful. Also, from $\|g_{m_j}\| \geq \eta_2/\gamma_j^2$ and (8) (note that $(1-x)/x$ is decreasing in $(0,1]$), γ is updated in step 3 as

$$\gamma_{j+1} = \frac{\gamma_j}{\lambda^{\frac{1-p_j}{p_j}}}.$$

Let now B_j be a random variable with realization $b_j = \log_\lambda(b_\epsilon/\gamma_j)$. In the case $1_{S_j} = 1$,

$$b_{j+1} = b_j + \frac{1-p_j}{p_j}.$$

If $1_{S_j} = 0$, then $b_{j+1} \geq b_j - 1$, because either $\gamma_{j+1} \leq \gamma_j$ and therefore $b_{j+1} \geq b_j$, or $\gamma_{j+1} = \lambda\gamma_j$ and therefore $b_{j+1} \geq b_j - 1$. Hence $B_j - B_{j_0} \geq W_j - W_{j_0}$, and from $P(\limsup_{j\to\infty} W_j = \infty) = 1$ one obtains $P(\limsup_{j\to\infty} B_j = \infty) = 1$ which leads to a contradiction with the fact that $B_j < 0$ happens for all $j \geq j_0$ with positive probability. ∎

6. A numerical illustration. The main concern in the application of Algorithm 3.1 is to ensure that the gradient model is (p_j)-probabilistically accurate (i.e., $p_j^* \geq p_j$; see Assumption 3.1) or at least to find a lower bound $p_{\min} > 0$ such that $p_j^* \geq p_{\min}$. However, one can, in some situations, overcome these difficulties such as in the cases where the model gradient (i) is a Gaussian perturbation of the exact one, or (ii) results from using either the exact one (seen as expensive) or an approximation. In the former case we will consider a run of the algorithm under a stopping criterion of the form $\gamma_j > \gamma_{\max}$.

6.1. Gaussian noise. At each iteration of the algorithm, we consider an artificial random gradient model, by adding to the exact gradient an independent Gaussian noise, more precisely we have $g_{M_j} = J(X_j)^\top \nabla F(X_j) + \varepsilon_j$, where $(\varepsilon_j)_i \sim N(0, \sigma_{j,i}^2)$ for $i = 1, \ldots, n$. Let Σ_j be a diagonal matrix with diagonal elements $\sigma_{j,i}$, $i = 1, \ldots, n$. It is known that

$$\|\Sigma_j \varepsilon_j\|^2 = \sum_{i=1}^{n} \left(\frac{(\varepsilon_j)_i}{\sigma_{j,i}}\right)^2 \sim \chi_2(n),$$

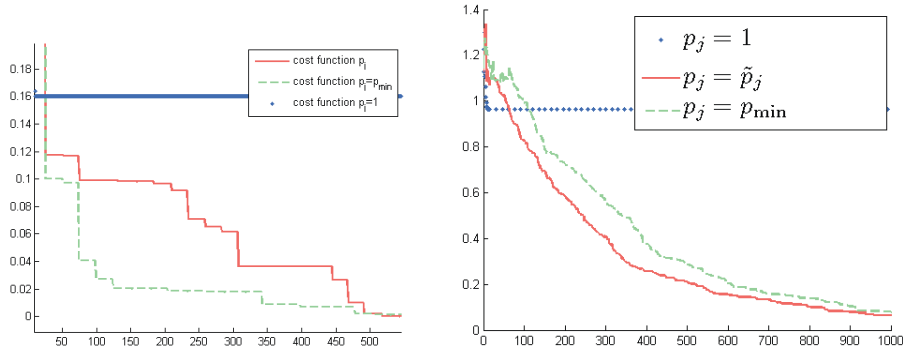
where $\chi_2(n)$ is the chi-squared distribution with n degrees of freedom. To be able to give an explicit form of the probability of the model being κ_{eg}-first-order accurate, for a chosen $\kappa_{eg} > 0$, we assume also that the components of the noise are identically distributed, that is, $\sigma_{j,i} = \sigma_j \ \forall i \in \{1, \ldots, n\}$. Because of the way in which γ_j is updated in Algorithm 3.1, it is bounded by $\lambda^j\gamma_0$ and, thus, $\Gamma_j \leq \min\{\lambda^j\gamma_0, \gamma_{\max}\}$, where $\gamma_{\max}$ is the constant used in the stopping criterion. One therefore has

$$\begin{aligned} p_j^* &= P\left(\|g_{M_j} - J(X_j)^\top F(X_j)\| \leq \frac{\kappa_{eg}}{\Gamma_j^\alpha} \,\middle|\, F_{j-1}^M\right) \\ &\geq P\left(\|\Sigma_j\varepsilon_j\|^2 \leq \left(\frac{\kappa_{eg}}{\sigma_j \min\{\lambda^j\gamma_0, \gamma_{\max}\}^\alpha}\right)^2 \,\middle|\, F_{j-1}^M\right). \end{aligned}$$

Using the Gaussian nature of the noise ε_j and the fact that it is independent from the filtration F_{j-1}^M, we obtain

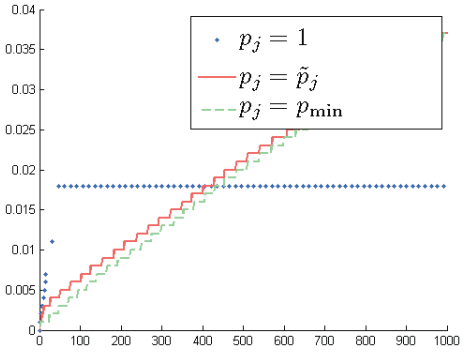
$$p_j^* \geq CDF_{\chi_2(n)}^{-1}\left(\left(\frac{\kappa_{eg}}{\sigma_j \min\{\lambda^j\gamma_0, \gamma_{\max}\}^\alpha}\right)^2\right) \stackrel{\text{def}}{=} \tilde{p}_j, \tag{9}$$

where $CDF_{\chi_2(n)}$ is the cumulative density function of a chi-squared distribution with n degrees of freedom.



(a) Average of function values.

(b) Average of absolute error of iterates.



(c) Average percentage of successful iterations.

Figure 1. *Average results of Algorithm 3.1 for 60 runs when using probabilities $p_j = 1$ (dotted line), $p_j = \tilde{p}_j$ (solid line), and $p_j = p_{\min}$ (dashed line). The x-axis represents number of iterations.*

The numerical illustration was done with the following nonlinear least squares problem defined using the well-known Rosenbrock function

$$f(x, y) = \frac{1}{2}\left(\|x - 1\|^2 + 100\|y - x^2\|^2\right) = \frac{1}{2}\|F(x, y)\|^2.$$

The minimizer of this problem is $(x^*, y^*)^\top = (1, 1)^\top$.

Algorithm 3.1 was initialized with $x_0 = (1.2, 0)^\top$ and $\gamma_0 = 1$. The algorithmic parameters were set to $\eta_1 = \eta_2 = 10^{-3}$, $\gamma_{\min} = 10^{-6}$, and $\lambda = 2$. The stopping criterion used is $\gamma_j > \gamma_{\max}$, where $\gamma_{\max} = 10^6$. We used $\alpha = 1/2$, $\sigma_j = \sigma = 10\ \forall j$, and $\kappa_{eg} = 100$ for the random gradient model.

Figure 1 depicts the average, over 60 runs of Algorithm 3.1, of the objective function values, the absolute errors of the iterates, and the percentages of successful iterations, using, across all iterations, the three choices $p_j = 1$, $p_j = \tilde{p}_j$, and $p_j = p_{\min}$. In the last case, $p_{\min}$ is an underestimation of p_j^* given by

$$p_{\min} = CDF^{-1}_{\chi^2(n)}\left(\left(\frac{\kappa_{eg}}{\sigma\gamma_{\max}^{\alpha}}\right)^2\right) = 5 \cdot 10^{-3}.$$

Table 1

For three different runs of Algorithm 3.1, the table shows the values of the objective function and relative error of the solution found for the three choices $p_j = 1$, $p_j = \tilde{p}_j$, and $p_j = p_{\min} = 5 \cdot 10^{-3}$.

Run number	1	2	3
$\|(x,y) - (x^*,y^*)\|/\|(x^*,y^*)\|$ $(p_j = 1)$	1.0168	0.3833	0.7521
$f(x,y)$ $(p_j = 1)$	0.5295	0.0368	1.47
$\|(x,y) - (x^*,y^*)\|/\|(x^*,y^*)\|$ $(p_j = \tilde{p}_j)$	0.0033	0.0028	0.0147
$f(x,y)$ $(p_j = \tilde{p}_j)$	2.6474e-006	1.9778e-006	4.3548e-005
$\|(x,y) - (x^*,y^*)\|/\|(x^*,y^*)\|$ $(p_j = p_{\min})$	0.1290	0.1567	0.0068
$f(x,y)$ $(p_j = p_{\min})$	0.0036	0.0059	9.1426e-006

The final objective function values and the relative final errors are shown in Table 1 for the first three runs of the algorithm. One can see that the use of $p_j = \tilde{p}_j$ leads to a better performance than $p_j = p_{\min}$ (because $\tilde{p}_j \geq p_{\min}$ is a better bound for p_j^* than $p_{\min}$ is).

In the case where $p_j = 1$, Algorithm 3.1 exhibits a performance worse than for the two other choices of p_j. The algorithm stagnated after some iterations, and could not approximate the minimizer with a decent accuracy. In this case, γ_j is increasing along the iterations, and thus it becomes very large after some iterations while the step $s_j \sim 1/\gamma_j^2$ becomes very small.

Other numerical experiments (not reported here) have shown that, when the error on the gradient is small ($\sigma \ll 1$), the two versions $p_j = \tilde{p}_j$ and $p_j = 1$ give almost the same results, and this is consistent with the theory because when $\sigma \to 0$, from (9),

$$\tilde{p}_j \to CDF^{-1}_{\chi_2(n)}(\infty) = 1.$$

Note that, on the other extreme, when the error on the gradient is big ($\sigma \gg 1$), version $p_j = \tilde{p}_j$ approaches version $p_j = p_{\min}$ since $\tilde{p}_j \simeq p_{\min}$.

6.2. Expensive gradient case. Let us assume that, in practice, for a given problem, one has two routines for gradient calculation. The first routine computes the exact gradient and is expensive. The second routine is less expensive but computes only an approximation of the gradient. The model gradient results from a call to either routine. In this section, we propose a technique to choose the probability of calling the exact gradient which makes our approach applicable.

Algorithm 6.1 (Algorithm to determine when to call the exact gradient g_{M_j}).

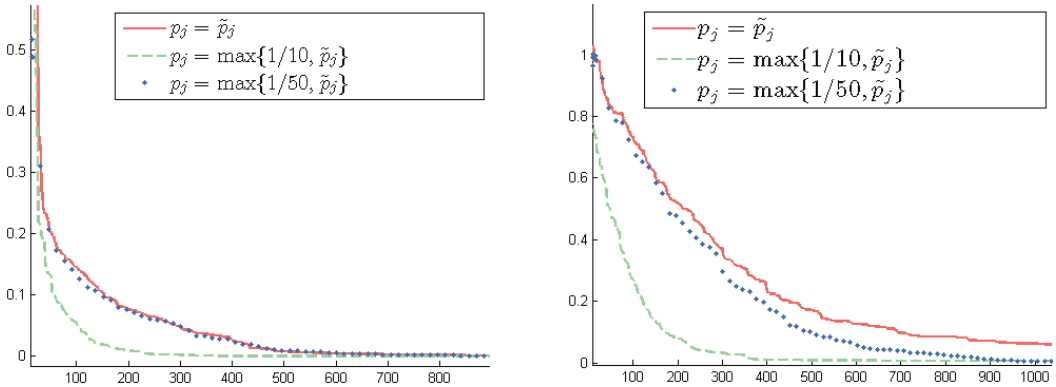
Initialization
Choose the constant $p_{\min} \in (0,1)$ ($p_{\min}$ is the lower bound of all the probabilities p_j^*).
For a chosen probability $\bar{p}_j$ such that $\bar{p}_j \geq p_{\min}$

1. Sample a random variable $U \sim \mathcal{U}([0, 1/\bar{p}_j])$, independently from F_{j-1}^M, and $\mathcal{U}([0, 1/\bar{p}_j])$ is the uniform distribution on the interval $[0, 1/\bar{p}_j]$.

1.1 If $U \leq 1$, compute g_{M_j} using the routine which gives the exact gradient.

1.2 Otherwise, compute g_{M_j} using the routine which gives an approximation of the exact gradient.

Lemma 6.1. *If we use Algorithm 6.1 to compute the model gradient at the jth iteration of Algorithm 3.1, then we have $p_j^* \geq \bar{p}_j \geq p_{\min}$.*



(a) Average of function values.

(b) Average of absolute error of iterates.

Figure 2. *Average results of Algorithm 3.1 for 60 runs when using probabilities $p_j = \tilde{p}_j$ (solid line), $p_j = \max\{1/10, \tilde{p}_j\}$ (dotted line), and $p_j = \max\{1/50, \tilde{p}_j\}$ (dashed line). The x-axis represents number of iterations.*

Proof. By using inclusion of events, we have that

$$p_j^* = P\left(\|g_{M_j} - J(X_j)^\top F(X_j)\| \leq \frac{\kappa_{eg}}{\Gamma_j^\alpha} \middle| F_{j-1}^M \right)$$
$$\geq P\left(\|g_{M_j} - J(X_j)^\top F(X_j)\| = 0 \middle| F_{j-1}^M \right)$$

and from Algorithm 6.1 we conclude that

$$P\left(\|g_{M_j} - J(X_j)^\top F(X_j)\| = 0 \middle| F_{j-1}^M \right) \geq P(U \leq 1) = \frac{1}{1/\bar{p}_j},$$

and thus $p_j^* \geq \bar{p}_j$. The other inequality, $\bar{p}_j \geq p_{\min}$, is imposed in the algorithm. ∎

For the experiments we use the same test function and the same parameters as in section 6.1. In step 1.2 of Algorithm 6.1, we set the model gradient g_{M_j} to the exact gradient of the function plus a Gaussian noise sampled from $N(0, 10I)$. Across all iterations, we use Algorithm 6.1 to compute g_{M_j} with the three following choices of $\bar{p}_j$:

- $\bar{p}_j = 1/10$, i.e., at iteration j the model gradient coincides with the exact gradient with probability at least $\bar{p}_j = 1/10$. Moreover, we have $p_j^* \geq \tilde{p}_j$, where $\tilde{p}_j$ is the same as in (9), and thus one can choose $p_j = \max\{1/10, \tilde{p}_j\}$.
- $\bar{p}_j = 1/50$, with the same analysis as before and one can choose $p_j = \max\{1/50, \tilde{p}_j\}$.
- $\bar{p}_j \simeq 0$ ($\bar{p}_j = 10^{-10}$ in the experiment below), i.e., at iteration j the probability that the model gradient coincides with the exact gradient is very small. Thus one can choose $p_j = \tilde{p}_j$.

Figure 2 depicts the average of the function values and the absolute error of the iterates over 60 runs of Algorithm 3.1 when using the three choices of the probability p_j. As expected, the better the quality of the model is the more efficient Algorithm 3.1 is (fewer iterations are needed to "converge" in the sense of sufficiently reducing the objective function value

and absolute error). We can clearly see that Algorithm 3.1 using the models for which $p_j = \max\{1/10, \tilde{p}_j\}$ provides a better approximation to the minimizer of the objective function than using the models for which $p_j = \max\{1/50, \tilde{p}_j\}$, and this latter one is better than the case when $p_j = \tilde{p}_j$.

7. Application to data assimilation. Data assimilation is the process by which observations of a real system are incorporated into a computer model (the forecast) to produce an estimate (the analysis) of the state of the system. 4DVAR is the data assimilation method mostly used in numerical weather prediction centers worldwide. 4DVAR attempts to reconcile a numerical model and the observations, by solving a very large weighted nonlinear least squares problem. The unknown is a vector of system states over discrete points in time. The objective function to be minimized is the sum of the squares of the differences between the initial state and a known background state at the initial time and the differences between the actual observations and ones predicted by the model.

7.1. 4DVAR problem. We want to determine $x_0, \ldots, x_T$, where x_i is an estimator of the state X_i at time i, from the background state $X_0 = x_b + W_b$, $W_b \sim N(0, B)$. The observations are denoted by $y_i = \mathcal{H}_i(X_i) + V_i$, $V_i \sim N(0, R_i)$, $i = 0, \ldots, T$, and the numerical model by $X_i = \mathcal{M}_i(X_{i-1}) + W_i$, $W_i \sim N(0, Q_i)$, $i = 1, \ldots, T$, where $\mathcal{M}_i$ is the model operator at time i and $\mathcal{H}_i$ is the observation operator at time i (both not necessarily linear). The random vectors W_b, V_i, W_i are the noises on the background, on the observation at time i, and on the model at time i, respectively, and are supposed to be Gaussian distributed with mean zero and covariance matrices B, R_i, and Q_i, respectively. Assuming that the errors (the background, the observation, and the model errors) are independent from each other and uncorrelated in time [9], the posterior probability function of this system (in other words, the pdf of $X_0, \ldots, X_T$ knowing $y_0, \ldots, y_T$) is proportional to

$$\exp^{-\frac{1}{2}\left(\|x_0 - x_b\|^2_{B^{-1}} + \sum_{i=1}^{T} \|x_i - \mathcal{M}_i(x_{i-1})\|^2_{Q_i^{-1}} + \sum_{i=0}^{T} \|y_i - \mathcal{H}_i(x_i)\|^2_{R_i^{-1}}\right)} \tag{10}$$

and therefore the maximizer of the posterior probability function estimator is defined to be the minimizer of the weak constraint 4DVAR problem [20] defined as the minimizer of the function defined in (2), which is the negative logarithm of (10).

7.2. Incremental 4DVAR. To find the solution of the nonlinear least squares problem (2), one proceeds iteratively by linearization. At each iteration, one solves the auxiliary linear least squares subproblem defined in (3) for the increments $\delta x_0, \ldots, \delta x_T$. Such an iterative process is nothing else than the Gauss–Newton method [2] for nonlinear least squares, known in the data assimilation community as the incremental approach [7].

Denote $z_i = \delta x_i$, $z_b = x_b - x_0$, $z = [z_0; \ldots; z_T]$, $m_i = \mathcal{M}_i(x_{i-1}) - x_i$, $d_i = y_i - \mathcal{H}_i(x_i)$, $M_i = \mathcal{M}_i'(x_{i-1})$, and $H_i = \mathcal{H}_i'(x_i)$. Then (3) becomes

$$\min_{z \in \mathbb{R}^{n(T+1)}} \frac{1}{2} \left(\|z_0 - z_b\|^2_{B^{-1}} + \sum_{i=1}^{T} \|z_i - M_i z_{i-1} - m_i\|^2_{Q_i^{-1}} + \sum_{i=0}^{T} \|d_i - H_i z_i\|^2_{R_i^{-1}} \right). \tag{11}$$

It is known that the solution of the linear least squares problem (11) is exactly the same as

the Kalman smoother estimator for the following linear system (see [21])

$$Z_0 = z_b + W_b, \quad W_b \sim N(0, B), \tag{12}$$

$$Z_i = M_i Z_{i-1} + m_i + W_i, \quad W_i \sim N(0, Q_i), \quad i = 1, \dots, T, \tag{13}$$

$$d_i = H_i Z_i + V_i, \quad V_i \sim N(0, R_i), \quad i = 0, \dots, T. \tag{14}$$

For simplicity, we now rewrite the linear system (12)–(14) as

$$Z = Z_b + W, \quad W \sim N(0, B_W), \tag{15}$$

$$D = HZ + V, \quad V \sim N(0, R), \tag{16}$$

where

$$\begin{aligned}
Z &= [Z_0; \dots; Z_T] \text{ is the joint state of the states } Z_0, \dots, Z_T,\\
D &= [d_0; d_1; \dots; d_T],\\
Z_b &= [z_b; M_1 z_b + m_1; M_2(M_1 z_b + m_1) + m_2; \dots; M_T(\cdots M_1 z_b + m_1 \cdots) + m_T],\\
H &= \operatorname{diag}(H_0, \dots, H_T) \text{ is the joint observation operator},\\
W &= [W_b; M_1 W_b + W_1; M_2(M_1 W_b + W_1) + W_2; \dots; M_T(\cdots M_1 W_b + W_1 \cdots) + W_T],\\
B_W &= \operatorname{cov}(W), \quad V = [V_0; V_1; \dots; V_T], \quad \text{and} \quad R = \operatorname{cov}(V).
\end{aligned}$$

To simplify it even more, we make the change of variables $U = Z - Z_b$, and then (15)–(16) becomes

$$\begin{aligned}
U &\sim N(0, B_W),\\
D - HZ_b = HU + V, &\quad V \sim N(0, R),
\end{aligned}$$

and the linear least squares problem (11) becomes (with z replaced by $u + Z_b$)

$$\min_{u \in \mathbb{R}^{n(T+1)}} \frac{1}{2} \left(\|u\|^2_{B_W^{-1}} + \|D - HZ_b - Hu\|^2_{R^{-1}} \right). \tag{17}$$

To solve problem (17), we propose to use the EnKS as a linear least squares solver instead of the Kalman smoother. The ensemble approach is naturally parallelizable over the ensemble members. Moreover, the proposed approach uses finite differences from the ensemble, and no tangent or adjoint operators are needed (i.e., the method is free of derivatives).

7.3. Kalman and EnKS. The Kalman smoother gives the expectation and the covariance of the state U (equivalently Z) knowing the data D, in other words it calculates $U^a = E(U|D)$ and $P^a = \operatorname{cov}(U|D)$, and is described by

$$\begin{aligned}
U^a &= K(D - HZ_b),\\
P^a &= (I - KH)B_W,\\
K &= B_W H^\top (H B_W H^\top + R)^{-1}.
\end{aligned}$$

For Z one has $Z^a = E(Z|D) = Z_b + K(D - HZ_b)$. In the data assimilation community, the vector U^a (equivalently Z^a) is called the analysis and the matrix K is called the Kalman gain.

The EnKS [9, 10] consists of applying Monte Carlo to generate an ensemble following $N(0, B_W)$ and then use its corresponding empirical covariance matrix instead of B_W to approximate U^a. Let us denote by k the ensemble members index, running over $k = 1, \dots, N$, where N is the ensemble size. We sample an ensemble $\tilde{U}^k$ from $N(0, B_W)$ by first sampling w_b^k according to $N(0, B)$, w_1^k according to $N(0, Q_1), \dots, w_T^k$ according to $N(0, Q_T)$, and then by setting $\tilde{U}^k$ as follows: $\tilde{U}_0^k = w_b^k$, $\tilde{U}_1^k = M_1 w_b^k + w_1^k, \dots, \tilde{U}_T^k = M_T(\cdots M_1 w_b^k + w_1^k \cdots) + w_T^k$. Let $\tilde{U}^k = [\tilde{U}_0^k; \tilde{U}_1^k; \dots; \tilde{U}_T^k]$ and

$$\bar{\tilde{U}} = \frac{1}{N}\sum_{k=1}^{N} \tilde{U}^k \quad \text{and} \quad B^N = \frac{1}{N-1}\sum_{k=1}^{N} (\tilde{U}^k - \bar{\tilde{U}})(\tilde{U}^k - \bar{\tilde{U}})^\top$$

be the empirical mean and covariance of the ensemble $\tilde{U}^k$, respectively. One has

$$B^N = CC^\top, \quad \text{where} \quad C = \frac{1}{\sqrt{N-1}}[\tilde{U}^1 - \bar{\tilde{U}}, \tilde{U}^2 - \bar{\tilde{U}}, \dots, \tilde{U}^N - \bar{\tilde{U}}].$$

We then build the centered ensemble $U^k = \tilde{U}^k - \bar{\tilde{U}}$. Note that the empirical mean of the ensemble U^k is equal to zero and that its empirical covariance matrix is B^N.

Now one generates the ensemble $U^{k,a}$ as follows:

$$U^{k,a} = U^k + K^N(D - HZ_b - V^k), \tag{18}$$

where V^k is sampled from $N(0, R)$, and

$$K^N = B^N H^\top (HB^N H^\top + R)^{-1}.$$

In practice, the empirical covariance matrix B^N is never computed or stored since to compute the matrix products $B^N H^\top$ and $HB^N H^\top$ only matrix-vector products are needed:

$$B^N H^\top = \frac{1}{N-1}\sum_{k=1}^{N} U^k {U^k}^\top H^\top = \frac{1}{N-1}\sum_{k=1}^{N} U^k h_k^\top,$$

$$HB^N H^\top = H\frac{1}{N-1}\sum_{k=1}^{N} U^k {U^k}^\top H^\top = \frac{1}{N-1}\sum_{k=1}^{N} h_k h_k^\top,$$

$$K^N = \frac{1}{N-1}\sum_{k=1}^{N} U^k h_k^\top \left(\frac{1}{N-1}\sum_{k=1}^{N} h_k h_k^\top + R\right)^{-1},$$

where $h_k = HU^k = [H_0 U_0^k; \dots; H_T U_T^k]$.

We denote by $\bar{U}^a$ and $\bar{V}$ the empirical mean of the ensembles $U^{k,a}$ and V^k, respectively. One has from (18)

$$\bar{U}^a = K^N(D - HZ_b - \bar{V}). \tag{19}$$

It is known that when $N \to \infty$, $\bar{U}^a \to U^a$ in L^p (see [13, 15]) and, thus, asymptotically, $\bar{U}^a$ is the solution of the linearized subproblem (17) (and $\bar{U}^a + Z_b$ is the solution of the linearized subproblem (11)).

7.4. The linearized least squares subproblems arising in EnKS. From (19) we conclude that $\bar{U}^a$ is the Kalman smoother estimator for the following system,

$$\begin{aligned} &\tilde{U} \sim N(0, B^N), \\ &\tilde{D} = H\tilde{U} + \tilde{V}, \quad \tilde{V} \sim N(0, R), \quad \text{where } \tilde{D} = D - HZ_b - \bar{V}. \end{aligned} \tag{20}$$

Hence, for a large N (such that B^N is invertible), $\bar{U}^a$ is the solution of the following linear least squares problem

$$\min_{u \in \mathbb{R}^{n(T+1)}} \frac{1}{2}\left(\|u\|^2_{(B^N)^{-1}} + \|Hu - \tilde{D}\|^2_{R^{-1}} \right). \tag{21}$$

From the above derivation, we conclude that when we use the EnKS (until now with exact derivatives) to approximate the solution of the linearized subproblem (11), what is obtained is the solution of the linear least squares problem (21). The least squares model in (21) can be seen, in turn, as a realization of the following stochastic model,

$$\frac{1}{2}\left(\|u\|^2_{\mathcal{B}^{-1}} + \|Hu - \tilde{\mathcal{D}}\|^2_{R^{-1}} \right), \tag{22}$$

where $\mathcal{B}^{-1}$ and $\tilde{\mathcal{D}}$ are random variables, with realizations $(B^N)^{-1}$ and $\tilde{D}$, respectively.

Both the incremental method and the method which approximates the solution of the linearized subproblem (11) using EnKS may diverge. Convergence to a stationary point of (2) can be recovered by controlling the size of the step, and one possibility to do so is to consider the application of the Levenberg–Marquardt method as in Algorithm 3.1. As in [14], at each step, a regularization term is then added to the model in (21),

$$m(x+u) = \frac{1}{2}\left(\|u\|^2_{(B^N)^{-1}} + \|Hu - \tilde{D}\|^2_{R^{-1}} + \gamma^2\|u\|^2 \right), \tag{23}$$

which corresponds to adding a regularization term to the model (22)

$$M(x+u) = \frac{1}{2}\left(\|u\|^2_{\mathcal{B}^{-1}} + \|Hu - \tilde{\mathcal{D}}\|^2_{R^{-1}} + \Gamma^2\|u\|^2 \right). \tag{24}$$

We now provide the details about the solution of (23). For this purpose let

$$P^N = (I - K^N H)B^N. \tag{25}$$

Note that by using the Sherman–Morrison–Woodbury formula one has

$$P^N = \left((B^N)^{-1} + H^T R^{-1} H \right)^{-1}, \tag{26}$$

in other words, P^N is the inverse of the Hessian of model in (21).

Proposition 7.1. *The minimizer of the model* (23) *is* $u^* = \bar{U}^a - P^N(P^N + (1/\gamma^2)I_n)^{-1}\bar{U}^a$.

Proof. Since $\bar{U}^a$ is the solution of problem (21), a Taylor expansion around $\bar{U}^a$ of the model in (21) gives

$$\frac{1}{2}\left(\|u\|^2_{(B^N)^{-1}} + \|Hu - \tilde{D}\|^2_{R^{-1}}\right) = \frac{1}{2}\left(\|\bar{U}^a\|^2_{(B^N)^{-1}} + \|H\bar{U}^a - \tilde{D}\|^2_{R^{-1}} + \|u - \bar{U}^a\|^2_{(P^N)^{-1}}\right).$$

Hence, the minimizer of the model (23) is the same as the minimizer of

$$\frac{1}{2}\left(\|\bar{U}^a\|^2_{(B^N)^{-1}} + \|H\bar{U}^a - \tilde{D}\|^2_{R^{-1}} + \|u - \bar{U}^a\|^2_{(P^N)^{-1}} + \gamma^2\|u\|^2\right)$$

and thus given by

$$u^* = \left((P^N)^{-1} + \gamma^2 I\right)^{-1}(P^N)^{-1}\bar{U}^a. \tag{27}$$

By using the Sherman–Morrison–Woodbury formula, one has

$$\left((P^N)^{-1} + \gamma^2 I\right)^{-1} = P^N - P^N\left(P^N + (1/\gamma^2)I_n\right)^{-1}P^N,$$

which together with (27) concludes the proof. ∎

7.5. Derivative-free LM-EnKS. The linearized model (LM) and observation operators appear only when acting on a given vector, and therefore they could be efficiently approximated by finite differences. The linearized observation operator $H_i = \mathcal{H}_i'(x_i)$ appears in the action on the ensemble members and can be approximated by

$$H_i\delta x_i = \mathcal{H}_i'(x_i)\delta x_i \simeq \frac{\mathcal{H}_i\left(x_i + \tau\delta x_i\right) - \mathcal{H}_i\left(x_i\right)}{\tau},$$

where $\tau > 0$ is a finite differences parameter. Originally, in EnKS, to avoid the derivatives of $\mathcal{H}_i$, the quantity $H_i\delta x_i = H_i(x_i + \delta x_i - x_i)$ is approximated by $\mathcal{H}_i(x_i + \delta x_i) - \mathcal{H}_i(x_i)$, which is equivalent to using finite differences with the parameter $\tau = 1$. The LM $M_1 = \mathcal{M}_1'(x_0)$ appears in the action on a given vector (in this case z_b), and so do the remaining ones $M_3, \ldots, M_T$. Such actions can be approximated by finite differences in the following way:

$$\begin{aligned}
M_1 z_b = \mathcal{M}_1'(x_0)z_b &\simeq \frac{\mathcal{M}_1(x_0 + \tau z_b) - \mathcal{M}_1(x_0)}{\tau},\\
M_2(M_1 z_b + m_1) = \mathcal{M}_2'(x_1)(M_1 z_b + m_1) &\simeq \frac{\mathcal{M}_2\left(x_1 + \tau(M_1 z_b + m_1)\right) - \mathcal{M}_2(x_1)}{\tau}\\
&\simeq \frac{\mathcal{M}_2\left(x_1 + \mathcal{M}_1(x_0 + \tau z_b) - \mathcal{M}_1(x_0) + \tau m_1\right) - \mathcal{M}_2(x_1)}{\tau}.
\end{aligned}$$

Since our approach is derivative free, we replace all the derivatives of the model and of the observation operators by approximation by finite differences. The quantities using derivatives

then become

$$\hat{h}_k = \left[\frac{\mathcal{H}_0\left(x_0 + \tau U_0^k\right) - \mathcal{H}_0\left(x_0\right)}{\tau}; \ldots; \frac{\mathcal{H}_T\left(x_T + \tau U_T^k\right) - \mathcal{H}_T\left(x_T\right)}{\tau} \right] \simeq h_k,$$

$$\hat{K}^N = \frac{1}{N-1} \sum_{k=1}^{N} U^k \hat{h}_k^\top \left(\frac{1}{N-1} \sum_{k=1}^{N} \hat{h}_k \hat{h}_k^\top + R \right)^{-1} \simeq K^N, \tag{28}$$

$$\hat{Z}_b = \left[z_b; \frac{\mathcal{M}_1\left(x_0 + \tau z_b\right) - \mathcal{M}_1\left(x_0\right)}{\tau} + m_1; \ldots \right] \simeq Z_b, \tag{29}$$

$$\hat{H}Z_b = \left[\frac{\mathcal{H}_0\left(x_0 + \tau z_b\right) - \mathcal{H}_0\left(x_0\right)}{\tau}; \ldots \right] \simeq HZ_b,$$

$$\hat{U}^a = \hat{K}^N (D - \hat{H}Z_b - \bar{V}) \simeq \bar{U}^a,$$

$$\hat{P}^N = B^N - \hat{K}^N \frac{1}{N-1} \sum_{k=1}^{N} \hat{h}_k {U^k}^\top \simeq P^N, \tag{30}$$

$$\hat{u}^* = \hat{U}^a - \hat{P}^N \left(\hat{P}^N + (1/\gamma^2) I_n \right)^{-1} \hat{U}^a \simeq u^*. \tag{31}$$

Since $\hat{u}^*$ is an approximation to u^* using finite differences for derivatives, there exists a constant $M > 0$, which depends on the second derivatives of the model and observation operators, such that $\|e\| \leq M\tau$, where $e = u^* - \hat{u}^*$. Moreover, the minimizer u^* of the weighted least squares model (23) is the solution of the normal equations

$$\left(\left(B^N\right)^{-1} + H^T R^{-1} H + \gamma^2 I \right) u^* = H^T R^{-1} \tilde{D},$$

where $H^T R^{-1} \tilde{D} = \nabla m(x) = g_m$, and thus

$$\left(\left(B^N\right)^{-1} + H^T R^{-1} H + \gamma^2 I \right) \hat{u}^* = g_m - \left(\left(B^N\right)^{-1} + H^T R^{-1} H + \gamma^2 I \right) e,$$

and so $\hat{u}^*$ can be seen as an inexact solution of the normal equations, with a residual equal to

$$r = -\left((B^N)^{-1} + H^T R^{-1} H + \gamma^2 I \right) e.$$

We have seen that the solution of the normal equations can be inexact as long as Assumption 4.2 is met. The residual r is then required to satisfy $\|r\| \leq \epsilon \|g_m\|$ for some $\epsilon > 0$, to fulfill the global convergence requirements of our Levenberg–Marquardt approach, and for this purpose we need the following assumption.

Assumption 7.1. The approximation $\hat{u}^*$ of u^* satisfies $\|e\| \leq M\tau$, where $e = u^* - \hat{u}^*$, for some constant $M > 0$.

The Jacobian of the observation operator $\mathcal{H}$ is uniformly bounded, i.e., there exists $\kappa_H > 0$ such that $\|\mathcal{H}_i'(x_i)\| \leq \kappa_H$ for all $i \in \{0, \ldots, T\}$ and for all iterations j.

We note that the iteration index j has been omitted from the notation of this section until now. In fact, the point x has been denoting the iterate x_j.

Proposition 7.2. *Under Assumption 7.1, if the finite differences parameter τ is such that*

$$\tau \leq \frac{\epsilon \|g_m\|}{M\left(\|(B^N)^{-1}\| + \kappa_H^2 \|R^{-1}\| + \gamma^2\right)}, \tag{32}$$

then $\|r\| \leq \epsilon \|g_m\|$.

Proof. One has

$$\begin{aligned}\|r\| &\leq \left\|(B^N)^{-1} + H^T R^{-1} H + \gamma^2 I\right\| \|e\| \\ &\leq \left(\|(B^N)^{-1}\| + \kappa_H^2 \|R^{-1}\| + \gamma^2\right) M\tau \leq \epsilon \|g_m\|.\end{aligned}$$

■

Now, from (24) the gradient of the stochastic model is $g_{M_j} = -H^\top R^{-1}\tilde{\mathcal{D}}$ and from (17) the exact gradient of the function to be minimized in problem (2) is $-H^\top R^{-1}(D - HZ_b)$. Thus,

$$p_j^* = P\left(\|H^\top R^{-1}(D - HZ_b - \tilde{\mathcal{D}})\| \leq \frac{\kappa_{eg}}{\Gamma_j^\alpha} \middle| \mathcal{F}_{j-1}^{\tilde{M}}\right).$$

But we know that $D - HZ_b - \tilde{\mathcal{D}} = \bar{V} = (1/N)\sum_{i=1}^N V_i$, where V_i are independently and identically distributed and follow $N(0, R)$, and thus $D - HZ_b - \tilde{\mathcal{D}} \sim N(0, R/N)$ and $R^{-1}(D - HZ_b - \tilde{\mathcal{D}}) \sim \frac{R^{-1/2}}{\sqrt{N}} N(0, I)$. Thus

$$\begin{aligned}p_j^* &\geq P\left(\frac{\kappa_H \|R^{-1/2}\|}{\sqrt{N}} \|N(0, I)\| \leq \frac{\kappa_{eg}}{\Gamma_j^\alpha} \middle| \mathcal{F}_{j-1}^{\tilde{M}}\right) \\ &= P\left(\|N(0, I)\| \leq \frac{\kappa\sqrt{N}}{\Gamma_j^\alpha} \middle| \mathcal{F}_{j-1}^{\tilde{M}}\right),\end{aligned}$$

where $\kappa = \frac{\kappa_{eg}}{\kappa_H \|R^{-1/2}\|}$. Since $\Gamma_j \leq \min\{\lambda^j \gamma_0, \gamma_{\max}\}$,

$$p_j^* \geq CDF_{\chi^2(m)}^{-1}\left(\left(\frac{\kappa\sqrt{N}}{\min\{\lambda^j\gamma_0, \gamma_{\max}\}^\alpha}\right)^2\right) \stackrel{\text{def}}{=} \tilde{p}_j, \tag{33}$$

where $m = \sum_{i=0}^T m_i$, m_i is the size of y_i, and $\gamma_{\max}$ is the tolerance used in the stopping criterion. Note that $\lim_{N\to\infty} \tilde{p}_j = 1$, thus $\lim_{N\to\infty} p_j^* = 1$, and hence when $N \to \infty$ the gradient approximation using ensemble converges almost surely to the exact gradient.

We are now ready to propose a version of Algorithm 3.1 for the solution of the 4DVAR problem (2) when using EnKS as the linear solver.

Algorithm 7.1 (Levenberg–Marquardt method based on probabilistic gradient models for data assimilation).

Initialization

Choose the constants $\eta_1 \in (0, 1)$, $\eta_2, \gamma_{\min}, \gamma_{\max} > 0$, and $\lambda > 1$. Select x_0 and $\gamma_0 \in [\gamma_{\min}, \gamma_{\max}]$. Choose all the parameters related to solving the 4DVAR problem (2) using EnKS as the linear solver.

For $j = 0, 1, 2, \ldots$ **and while** $\gamma_j \leq \gamma_{\max}$

1. Let $x = x_j$. Choose τ satisfying (32). Compute the increment $\hat{u}^*$ using (31) and set $z^* = \hat{u}^* + \hat{Z}_b$, where $\hat{Z}_b$ is computed as in (29). Let $s_j = z^*$.
2. Compute $\rho_j = \frac{f(x_j) - f(x_j + s_j)}{m_j(x_j) - m_j(x_j + s_j)}$, where f is the nonlinear least squares model in (2) and m_j is the model (23).
3. If $\rho_j \geq \eta_1$, then set $x_{j+1} = x_j + s_j$ and

$$\gamma_{j+1} = \begin{cases} \lambda \gamma_j & \text{if } \|g_{m_j}\| < \eta_2 / \gamma_j^2, \\ \max\left\{ \frac{\gamma_j}{\lambda \sqrt{\frac{1-p_j}{p_j}}}, \gamma_{\min} \right\} & \text{if } \|g_{m_j}\| \geq \eta_2 / \gamma_j^2, \end{cases}$$

where $p_j = \tilde{p}_j$ is computed as in (33).
Otherwise, set $x_{j+1} = x_j$ and $\gamma_{j+1} = \lambda \gamma_j$.

7.6. Derivative-free LM-EnKS in practice. In sections 7.4–7.5, we have assumed that the ensemble size N was large enough for the empirical covariance matrix B^N to be invertible (holding Proposition 7.1). However, for the EnKS to be relevant the ensemble size has to be smaller than the dimension of the state space. In this subsection, we explain how we circumvent this problem in practice. The theoretical extension of our method to small ensemble sizes as well as its performance for large and realistic problems is the subject of future research.

For small values of the ensemble size ($N < n$), the matrix B^N is no longer invertible. In particular, the Sherman–Morrisson–Woodbury formula is no longer applicable, as it was before to establish (25)–(26) in terms of $(B^N)^{-1}$. However, following the spirit of (26), we could think of using pseudoinverses instead, and approximating the matrix P^N defined in (25) by

$$P^N = \left((B^N)^\dagger + H^T R^{-1} H \right)^\dagger .$$

In practice what we do is simply to replace inverses by pseudoinverses in all calculations, namely, in (28) and in (31).

Another concern when using a small ensemble size is how to ensure Assumption 3.1 (gradient model being (p_j)-probabilistically accurate). When N is sufficiently large, we have shown that the formula (33) provides a value of p_j that satisfies the assumption. This formula can, however, still be used in practice for small values of N.

7.7. Computational experiments with Lorenz–63 model. To evaluate the performance of Algorithm 7.1 for data assimilation, we will test it using the classical twin experiment technique used in the data assimilation community. This technique consists on fixing an initial true state (denoted by truth_0) and then to integrate it over time using the model to obtain the true state at each time i (denoted by truth_i). We then build the data y_i by applying the observation operator $\mathcal{H}_i$ to the truth at time i and by adding a Gaussian perturbation $N(0, R_i)$. Similarly, the background x_b is sampled from the Gaussian distribution with mean truth_0 and covariance matrix B. Then we try to recover the truth using the observations and the background.

For the 4DVAR problem (2), we consider the Lorenz–63 model, a simple dynamical system with chaotic behavior. The Lorenz equations are given by the nonlinear system

$$\frac{dx}{dt} = -\sigma(x-y), \quad \frac{dy}{dt} = \rho x - y - xz, \quad \text{and} \quad \frac{dz}{dt} = xy - \beta z,$$

where $x = x(t)$, $y = y(t)$, $z = z(t)$, and σ, ρ, β are parameters. The state at time t is $X_t = (x(t), y(t), z(t))^\top \in \mathbb{R}^3$. This nonlinear system is discretized using a fourth order Runge–Kutta method. The parameters σ, ρ, β are chosen as 10, 28, and 8/3, respectively. The initial truth is set to $(1,1,1)^\top$ and the truth at time i to $\text{truth}_i = \mathcal{M}(\text{truth}_{i-1}) + W_i$, where W_i is sampled from $N(0, Q_i)$ and $\mathcal{M}$ is the model obtained by discretization of the Lorenz–63 model. The model error covariance is given by $Q_i = \sigma_q^2 I$, where $\sigma_q = 10^{-4}$. The background mean x_b is sampled from $N(\text{truth}_0, B)$. The background covariance is $B = \sigma_b^2 I$, where $\sigma_b = 1$. The time step is chosen as $dt = 0.11 > 0.01$. (Note here that the model at time $t+1$, as a function of the model at time t, becomes more nonlinear as dt increases, and this justifies having chosen dt larger than in [3].) The time windows length is $T = 40$. The observation operator is $\mathcal{H}_i = 10I$. At each time i, the observations are constructed as follows: $y_i = \mathcal{H}_i(\text{truth}_i) + V_i$, where V_i is sampled from $N(0, R)$, $R = \sigma_r^2 I$, and $\sigma_r = 1$.

Following the spirit of Assumption 4.2, the finite difference parameter is set as

$$\tau_j = \min\left\{10^{-3}, \frac{\epsilon_j \|g_{m_j}\|}{M\left(\|(B^N)^\dagger\| + \kappa_H^2 \|R^{-1}\| + \gamma_j^2 \|(B^N)^\dagger\|\right)}\right\},$$

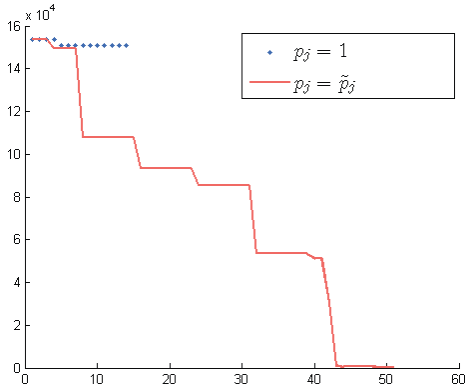
where the value of 1 is chosen for the unknown constants M and κ_H (see Assumption 7.1). In this experimental framework, the model gradient is given by $g_{m_j} = -H^\top R^{-1} \tilde{D} = 10\tilde{D}$, where $\tilde{D}$ is computed according to (20). Then, following the spirit of Assumption 4.2, ϵ_j is chosen as

$$\epsilon_j = \min\left\{\frac{\theta_{in}}{\gamma_j^\alpha}, \sqrt{\beta_{in}\frac{\gamma_j^2}{\kappa_{Jm}^2 + \gamma_j^2}}\right\},$$

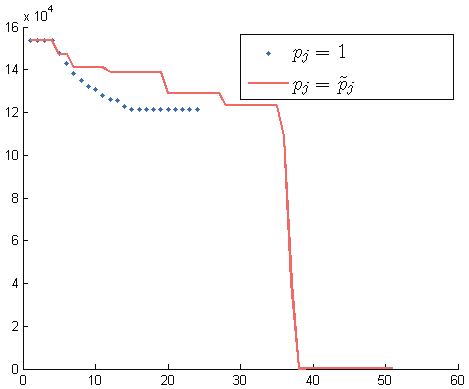
where $\beta_{in} = 1/2$, $\theta_{in} = 1$, and $\alpha = 0.5$. The unknown constant κ_{Jm} (see Assumption 5.1) is set to 1.

The basic algorithmic parameters are set to $\eta_1 = \eta_2 = 10^{-6}$, $\gamma_{\min} = 10^{-5}$, $\gamma_{\max} = 10^6$, and $\lambda = 8$. The initial regularization parameter is $\gamma_0 = 1$. Finally, we set $\kappa = 1$ in the calculation of $\tilde{p}_j$ given in (33).

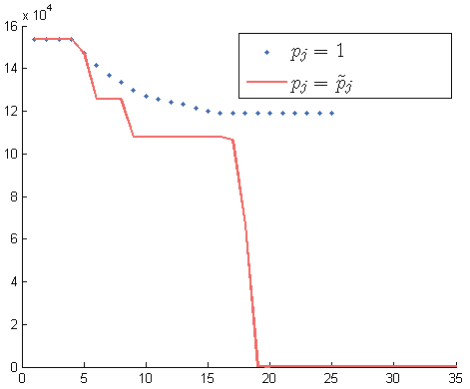
Figure 3 depicts the plots of the objective function values for one run of Algorithm 7.1, using the choices $p_j = \tilde{p}_j$ and $p_j = 1$ and four ensemble sizes $N = 4, 40, 80, 400$. A single run shows well the behavior of the algorithm on this problem, thus there is no need to take averages over several runs. For all ensemble sizes used, the version using $p_j = 1$ stagnated after some iterations, and could not approximate the minimizer with a decent accuracy. One can see that the version with $p_j = \tilde{p}_j$ performs much better than the one with $p_j = 1$, regardless of the size of the ensemble. As expected, the larger this size, the better is the accuracy of the final solution found (which can be further confirmed in Table 2). These results illustrate the importance of using probability $p_j = \tilde{p}_j$ to update the regularization parameter γ.



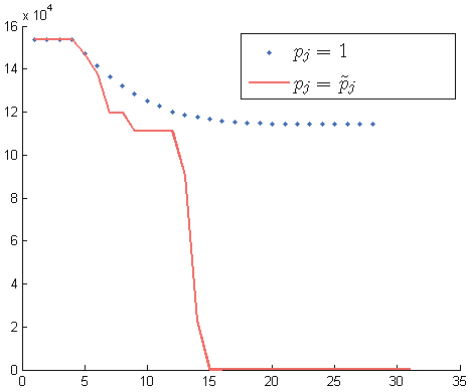
(a) Objective function values (size $N = 4$).



(b) Objective function values (size $N = 40$).



(c) Objective function values (size $N = 80$).



(d) Objective function values (size $N = 400$).

Figure 3. *Results of one run of Algorithm 7.1, using probabilities $p_j = 1$ (dotted line) and $p_j = \tilde{p}_j$ (solid line), for different ensemble sizes. The x-axis represents number of iterations.*

Table 2

The table shows the final values of the objective function found for the two versions $p_j = 1$ and $p_j = \tilde{p}_j$ and the four ensemble sizes.

Ensemble size	4	40	80	400
Final f $(p_j = 1)$	1.5e5	1.2e5	1.2e5	1.1e5
Final f $(p_j = \tilde{p}_j)$	304.7	65.7	62.1	63.1

8. Conclusions. In this paper we have adapted the Levenberg–Marquardt method for nonlinear least squares problems to handle the cases where the gradient of the objective function is subject to noise or only computed accurately within a certain probability. The gradient model was then considered random in the sense of being a realization of a random variable, and assumed first order accurate under some probability p_j^* (see (5)). Given the knowledge of a lower bound p_j for this probability (see Assumption 3.1), we have shown how to update the regularization parameter of the method in such a way that the whole approach

is almost surely globally convergent. The analysis followed similar steps as in the theory in [1]. The main difficulty in the application of Algorithm 3.1 is to ensure that the models are indeed (p_j)-probabilistically accurate, but we presented a number of practical situations where this is achievable.

The last section of the paper was devoted to the well-known 4DVAR problem in data assimilation. We have shown that a lower bound for the probability of first order accuracy can also be provided here (to be used in our Levenberg–Marquardt framework) when using the EnKS method for the formulation and solution of the corresponding linearized least squares subproblems.

We have also covered the situation where the linearized least squares problems arising in the Levenberg–Marquardt method are solved inexactly, which then encompasses a range of practical situations, from inexactness in linear algebra to inexactness in derivatives. This is particularly useful in the 4DVAR application to accommodate finite differences of the nonlinear operators involved.

A number of issues need further and deeper investigation, in particular, the study of the performance of our approach when applied to large and realistic data assimilation problems.

After we had submitted our paper, Bocquet and Sakov [4] extended their previous approach [3] to 4DVAR and used finite difference approximations for the tangent operators, similarly to our paper. Bocquet and Sakov [3, 4] nest the minimization loop for the 4DVAR objective function inside the EnKS and minimize over the span of the ensemble, rather than nesting EnKS as a linear solver inside the 4DVAR minimization loop over the full state space, as we do. Moreover, they use a classical version of the Levenberg–Marquardt method to perform their minimization without any control or assumption on the derivative approximations arising from the use of ensembles. Their method was designed for strong-constraint 4DVAR, i.e., for the case $Q_i = 0 \, \forall i$.

Appendix.

Proof of Lemma 4.1. In the proof we will omit the indices j. One has

$$\begin{aligned} m(x) - m(x + s^{in}) &= -g_m^\top s^{in} - \frac{1}{2}(-g_m + r)^\top s^{in} = -\frac{1}{2}(g_m + r)^\top s^{in} \\ &= \frac{1}{2}(g_m - r)^\top (J_m^\top J_m + \gamma^2 I)^{-1}(g_m + r). \end{aligned}$$

Since $J_m^\top J_m$ is positive semidefinite,

$$r^\top (J_m^\top J_m + \gamma^2 I)^{-1} r \leq \frac{\|r\|^2}{\gamma^2} \leq \frac{\epsilon^2 \|g_m\|^2}{\gamma^2}$$

and

$$(g_m)^\top (J_m^\top J_m + \gamma^2 I)^{-1} g_m \geq \frac{\|g_m\|^2}{\|J_m\|^2 + \gamma^2}.$$

Thus, using Assumption 4.2, we conclude that

$$\begin{aligned} m(x) - m(x + s^{in}) &\geq \left(\frac{1}{\|J_m\|^2 + \gamma^2} - \frac{\epsilon^2}{\gamma^2} \right) \|g_m\|^2 \\ &\geq \frac{2(1-\beta_{in})}{2} \frac{\|g_m\|^2}{\|J_m\|^2 + \gamma^2}. \end{aligned}$$

■

Proof of Lemma 5.1. We will again omit the indices j in the proof.

If $s = s^c$ is the Cauchy point, since $J_m^\top J_m$ is positive semidefinite, $\|g_m^\top (J_m^\top J_m + \gamma^2 I) g_m\| \geq \gamma^2 \|g_m\|^2$ and we have that $\|s^c\| \leq \|g_m\|/\gamma^2$. To prove the second inequality,

$$\begin{aligned} (s^c)^\top (\gamma^2 (s^c) + g_m) &= \frac{\gamma^2 \|g_m\|^6}{((g_m)^\top (J_m^\top J_m + \gamma^2 I) g_m)^2} - \frac{\|g_m\|^4}{(g_m)^\top (J_m^\top J_m + \gamma^2 I) g_m} \\ &= -\frac{\|g_m\|^4 (g_m)^\top J_m^\top J_m g_m}{((g_m)^\top (J_m^\top J_m + \gamma^2 I) g_m)^2}, \end{aligned}$$

and then using a similar argument and $\gamma \geq \gamma_{\min}$,

$$|(s^c)^\top (\gamma^2 (s^c) + g_m)| \leq \frac{\|J_m\|^2 \|g_m\|^2}{\gamma^4} \leq \frac{4\|J_m\|^2 \|g_m\|^2 + 2\theta_{in} \|g_m\|^2}{\min\{1, \gamma_{\min}^{2-\alpha}\} \gamma^{2+\alpha}}.$$

If $s = s^{cg}$ is obtained by truncated CG, then there exists an orthogonal matrix V with a first column given by $-g_m/\|g_m\|$ and such that

$$s^{cg} = V \left(V^\top (J_m^\top J_m + \gamma^2 I) V \right)^{-1} V^T g_m = V \left(V^\top J_m^\top J_m V + \gamma^2 I \right)^{-1} \|g_m\| e_1,$$

where e_1 is the first vector of the canonical basis of $\mathbb{R}^n$. From the positive semidefiniteness of $V^\top J_m^\top J_m V$, we immediately obtain $\|s^{cg}\| \leq \|g_m\|/\gamma^2$. To prove the second inequality we apply the Sherman–Morrisson–Woodbury formula, to obtain

$$s^{cg} = V \left(\frac{1}{\gamma^2} I - \frac{1}{\gamma^4} (J_m V)^\top \left(I + \frac{(J_m V)(J_m V)^\top}{\gamma^2} \right)^{-1} (J_m V) \right) \|g_m\| e_1.$$

Since $V e_1 = -g_m/\|g_m\|$,

$$\gamma^2 s^{cg} + g_m = -\frac{1}{\gamma^2} V (J_m V)^\top \left(I + \frac{(J_m V)(J_m V)^\top}{\gamma^2} \right)^{-1} (J_m V) \|g_m\| e_1.$$

Now, from the fact that $(J_m V)(J_m V)^\top/\gamma^2$ is positive semidefinite, the norm of the inverse of $I + (J_m V)(J_m V)^\top/\gamma^2$ is no greater than one, and thus (since V is orthogonal)

$$\|\gamma^2 s^{cg} + g_m\| \leq \frac{\|J_m\|^2 \|g_m\|}{\gamma^2}.$$

Finally (recalling $\gamma \geq \gamma_{\min}$),

$$|(s^{cg})^\top(\gamma^2(s^{cg}) + g_m)| \leq \|s^{cg}\|\|\gamma^2 s^{cg} + g_m\| \leq \frac{\|J_m\|^2\|g_m\|^2}{\gamma^4}$$
$$\leq \frac{4\|J_m\|^2\|g_m\|^2 + 2\theta_{in}\|g_m\|^2}{\min\{1, \gamma_{\min}^{2-\alpha}\}\gamma^{2+\alpha}}.$$

If $s = s^{in}$ is an inexact solution of the normal equations, and the residual satisfies Assumption 4.2, $\|s^{in}\| \leq (\|g_m\| + \|r\|)/\gamma^2 \leq 2\|g_m\|/\gamma^2$. Applying the Sherman–Morrisson–Woodbury formula,

$$s^{in} = \left(\frac{1}{\gamma^2}I - \frac{1}{\gamma^4}J_m^\top\left(I + \frac{J_m J_m^\top}{\gamma^2}\right)^{-1} J_m\right)(-g_m + r).$$

Thus,

$$\gamma^2 s^{in} + g_m = -\frac{1}{\gamma^2}J_m^\top\left(I + \frac{J_m J_m^\top}{\gamma^2}\right)^{-1} J_m(-g_m + r) + r.$$

Using the fact that the norm of the inverse above is no greater than one, Assumption 4.2, and $\gamma \geq \gamma_{\min}$,

$$|(s^{in})^\top(\gamma^2(s^{in}) + g_m)| \leq \|s^{in}\|\|\gamma^2 s^{in} + g_m\|$$
$$\leq \frac{4\|J_m\|^2\|g_m\|^2}{\gamma^4} + \frac{2\theta_{in}\|g_m\|^2}{\gamma^{2+\alpha}}$$
$$\leq \frac{4\|J_m\|^2\|g_m\|^2 + 2\theta_{in}\|g_m\|^2}{\min\{1, \gamma_{\min}^{2-\alpha}\}\gamma^{2+\alpha}}.$$

■

REFERENCES

[1] A. S. Bandeira, K. Scheinberg, and L. N. Vicente, *Convergence of trust-region methods based on probabilistic models*, SIAM J. Optim., 24 (2014), pp. 1238–1264.

[2] B. M. Bell, *The iterated Kalman smoother as a Gauss-Newton method*, SIAM J. Optim., 4 (1994), pp. 626–636.

[3] M. Bocquet and P. Sakov, *Combining inflation-free and iterative ensemble Kalman filters for strongly nonlinear systems*, Nonlinear Process. Geophys., 19 (2012), pp. 383–399.

[4] M. Bocquet and P. Sakov, *An iterative ensemble Kalman smoother*, Quart. J. Roy. Meteor. Soc., 140 (2014), pp. 1521–1535.

[5] A. R. Conn, N. I. M. Gould, and Ph. L. Toint, *Trust-Region Methods*, MPS/SIAM Ser. Optim., SIAM, Philadelphia, 2000.

[6] A. R. Conn, K. Scheinberg, and L. N. Vicente, *Introduction to Derivative-Free Optimization*, MPS/SIAM Ser. Optim. 8, SIAM, Philadelphia, 2009.

[7] P. Courtier, J. N. Thyepaut, and A. Hollingsworth, *A strategy for operational implementation of 4D-VAR, using an incremental approach*, Quart. J. Roy. Meteor. Soc., 120 (1994), pp. 1367–1387.

[8] R. Durrett, *Probability: Theory and Examples*, 4th ed., Camb. Ser. Stat. Probab. Math., Cambridge University Press, Cambridge, 2010.

[9] G. Evensen, *Data Assimilation: The Ensemble Kalman Filter*, Springer, Berlin, 2007.

[10] G. Evensen, *Data Assimilation: The Ensemble Kalman Filter*, 2nd ed., Springer, Dordrecht, The Netherlands, 2009.

[11] S. P. Khare, J. L. Anderson, T. J. Hoar, and D. Nychka, *An investigation into the application of an ensemble Kalman smoother to high-dimensional geophysical systems*, Tellus A, 60 (2008), pp. 97–112.
[12] K. Levenberg, *A method for the solution of certain problems in least squares*, Quart. Appl. Math., 2 (1944), pp. 164–168.
[13] F. Le Gland, V. Monbet, and V. C. Tran, *Large sample asymptotics for the ensemble Kalman filter*, in The Oxford Handbook of Nonlinear Filtering, D. Crisan and B. Rozovskii, eds., Oxford University Press, Oxford, 2011.
[14] J. Mandel, E. Bergou, S. Gürol, and S. Gratton, *Hybrid Levenberg Marquardt and weak constraint ensemble Kalman smoother method*, Nonlinear Process. Geophys. Discuss., 2 (2015), pp. 865–902.
[15] J. Mandel, L. Cobb, and J. B. Beezley, *On the convergence of the ensemble Kalman filter*, Appl. Math., 56 (2011), pp. 533–541.
[16] D. W. Marquardt, *An algorithm for least-squares estimation of nonlinear parameters*, J. Soc. Ind. Appl. Math., 11 (1963), pp. 431–441.
[17] J. J. Moré, *The Levenberg-Marquardt algorithm: Implementations and theory*, in Numerical Analysis, Lecture Notes in Math. 630, G. A. Watson, ed., Springer-Verlag, Berlin, 1977, pp. 105–116.
[18] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed., Springer-Verlag, Berlin, 2006.
[19] M. R. Osborne, *Nonlinear least squares–the Levenberg algorithm revisited*, J. Austr. Math. Soc. Ser. B, 19 (1976), pp. 343–357.
[20] Y. Trémolet, *Model-error estimation in 4D-Var*, Quart. J. Roy. Meteor. Soc., 133 (2007), pp. 1267–1280.
[21] J. Tshimanga, S. Gratton, A. T. Weaver, and A. Sartenaer, *Limited-memory preconditioners, with application to incremental four-dimensional variational data assimilation*, Quart. J. Roy. Meteor. Soc., 134 (2008), pp. 751–769.
[22] M. Zupanski, *Maximum likelihood ensemble filter: Theoretical aspects*, Monthly Weather Rev., 133 (2006), pp. 1710–1726.