Low rank updates in preconditioning the saddle point systems arising from data assimilation problems

M. Fisher[a], S. Gratton[b,c], S. Gürol[b*], Y. Trémolet[a] and X. Vasseur[d]

[a]European Centre for Medium-Range Weather Forecasts, Shinfield Park, Reading, RG2 9AX, UK; [b]CERFACS, 42 Avenue Gaspard Coriolis, 31057 Toulouse Cedex 01, France; [c]INP-ENSEEIHT, 31071 Toulouse, France; [d]ISAE-SUPAERO, 10 avenue Edouard Belin, BP 54032, 31055 Toulouse Cedex 4, France

The numerical solution of saddle point systems has received a lot of attention over the past few years in a wide variety of applications such as constrained optimization, computational fluid dynamics and optimal control, to name a few. In this paper, we focus on the saddle point formulation of a large-scale variational data assimilation problem, where the computations involving the constraint blocks are supposed to be much more expensive than those related to the $(1, 1)$ block of the saddle point matrix. New low-rank limited memory preconditioners exploiting the particular structure of the problem are proposed and analysed theoretically. Numerical experiments performed within the Object-Oriented Prediction System are presented to highlight the relevance of the proposed preconditioners.

Keywords: data assimilation; limited memory preconditioning; saddle point system; weak-constraint

1. Introduction

This paper investigates the saddle point representation of a large-scale quadratic minimization problem subject to linear equality constraints which is solved within a Gauss–Newton (GN) method. Therefore, in this study we deal with the solution of a sequence of saddle point systems particularly arising from data assimilation problems such as applications in Earth-system modelling. For instance, in meteorological applications the result of the data assimilation problem is the initial state of a dynamical system, which is then integrated forward in time to produce a weather forecast.

It has been shown that the saddle point formulation of the four-dimensional variational data assimilation (4D-Var) problem allows parallelization in the time dimension [34]. Therefore, this formulation represents a crucial step towards improved parallel scalability of the 4D-Var problem.

When the saddle point system is large, the problem is solved iteratively, usually by using a Krylov subspace method. When using Krylov methods, it is well-known that their convergence can be slow on indefinite systems unless a good preconditioner is used to accelerate the convergence. Therefore, finding an efficient preconditioner is crucial when solving the large-scale saddle point systems [1].

*Corresponding author. Email: gurol@cerfacs.fr

In recent years, a large amount of work has been devoted to find efficient preconditioners such as block diagonal, triangular and constrained preconditioners (see, e.g. [1,2,6,24, 25,28] and references therein). However, the efficiency of most of these preconditioners depends on the property that the computations involving the (1,1) primal variable block are computationally more expensive than the calculations involving the off-diagonal (constraint) blocks. This property does not hold for the saddle point representation of the variational data assimilation problem as we shall see. Therefore, many preconditioners in the literature become computationally expensive and not well-suited to our problem.

One preconditioner of interest for this study is the inexact constraint preconditioner proposed in [2]. It has been shown that this preconditioner can be used as a first-level preconditioner [34]. In this study, we investigate whether it is possible to further accelerate the convergence by improving the first-level preconditioner with low-rank updates generated from the inherited information.

The paper is structured as follows. In Section 2, we derive the variational data assimilation problem in which model errors are taken into account. We then focus on solution algorithms. The first standard solution approach includes the model constraints into the objective function and solves the resulting unconstrained optimization problem. By using this approach, the expensive matrix–vector products need to be performed sequentially in the time domain. As an alternative to the standard approach in terms of parallelization in the time domain, the second approach introduces additional constraints on the observation operator which leads to a quadratic problem with nonlinear equality constraints. This formulation is then solved by introducing a sequence of quadratic programs where second order derivatives of the constraints are ignored as in GN-like methods. The main computational work in this algorithm consists in solving saddle point linear system as discussed in [34].

In Section 3, we present new low-rank limited memory updates to precondition the saddle point matrix. These updates are obtained by solving the two-sided secant equations. According to our knowledge, two-sided equations are considered only in [19, 20] where authors used two-sided secant equations in order to approximate the Jacobian matrix. In this study, we use the approximation for preconditioning, which we believe is new in the literature. The performance of the derived preconditioners is shown in Section 4 by using a simple two-layer quasi-geostrophic model, and finally the conclusions are summarized in Section 5.

2. Problem definition and solution algorithms

We consider a set of states $\{x_i \in \mathbb{R}^n\}$ valid at times $\{t_i\}$, $i = 0, 1, \ldots, N$ during an analysis window $[t_0, t_N]$. We define the model error $\{q_i\}$, $i = 1, 2, \ldots, N$ associated with these states such that

$$x_i = \mathcal{M}_i(x_{i-1}) + q_i, \quad i \geq 1,$$

where $\mathcal{M}_i$ represents the physical model integration from time t_{i-1} to time t_i. We assume that a set of observations $y_i \in \mathbb{R}^{m_i}$ is available at time t_i, $i = 0, \ldots, N$. Our aim is then to find the best model fit to the observed data in the sense that the sum of square errors between the observed data and the model prediction is minimized. During this minimization, the model error will also be taken into account. This leads to a nonlinear least squares problem $\min_x f(x)$ with

$$f(x) = \frac{1}{2}\|x_0 - x_b\|^2_{B_b^{-1}} + \frac{1}{2}\sum_{i=0}^{N}\|\mathcal{H}_i(x_i) - y_i\|^2_{R_i^{-1}} + \frac{1}{2}\sum_{i=1}^{N}\|q_i\|^2_{Q_i^{-1}}, \tag{1}$$

where $x = [x_0, x_1, \ldots, x_N]^{\mathrm{T}}$ is a four-dimensional variable (i.e. time-distributed state variable), $x_b \in \mathbb{R}^n$ is the a priori information (background) for the state x_0, $\mathcal{H}_i$ is an observation operator

that represents a mapping from the state space to the observation space, B_b, R_i and Q_i are $n \times n$, $m_i \times m_i$ and $n \times n$ symmetric positive definite error covariance matrices corresponding to the background, observation and model errors, respectively. The first term in this nonlinear least squares problem is the Tikhonov regularization term [5, p. 101] which is often needed when the problem is under-determined, as it is the case in practical applications when there are fewer observations than variables in the model.

The formulation of the weighted nonlinear cost function (1) is known in the data assimilation community as the weak-constraint four-dimensional variational data assimilation (weak-4D-Var) problem [41,45]. This approach has its origin in the maximum likelihood estimation under the assumptions that all errors are unbiased, uncorrelated, and can be represented by zero mean Gaussian distributions [41]. In the nonlinear cost function (1), the model and observation errors are also assumed to be temporally uncorrelated. Time correlations can be included at the cost of using block-structured covariance matrices, provided that the required statistics is known [42]. We also note that, in practical applications, the Gaussian assumption may not be always realistic and can be partially relaxed [21,22,40]. The salient properties of this 4D-Var problem in real-life applications such as ocean or atmospheric data assimilation can be summarized as follows

- The dimension of the state variable n exceeds 10^6 in general. Hence, this problem can be really considered as a large-scale nonlinear least squares problem.
- The integrations of the physical model involving $\mathcal{M}_i$ are typically very costly. For this reason, exact second order derivative information is too expensive to compute.
- The error covariance matrices are not explicitly available. They are modelled or estimated using either smoothing functions or filters [9,44], and only their actions on a vector can be computed.

Taking into consideration these main properties, we review in Sections 2.1 and 2.2 practical algorithms for the solution of the large-scale nonlinear least squares problem given by (1).

2.1 *Solution via the GN method*

A widely used algorithm to minimize the large-scale nonlinear least squares cost function (1) is the GN method known in the data assimilation community as Incremental 4D-Var [10]. At each iteration (say the jth iteration) of the GN method, named as the outer loop, a step (an increment) $\delta x^{(j)}$ from a given $x^{(j)}$ is computed by minimizing a quadratic cost function which is the linearized least squares approximation of the nonlinear problem. This quadratic approximation in the neighbourhood of the iterate $x^{(j)}$ is given by

$$\mathcal{J}^{(j)} = \frac{1}{2}\|\delta x_0 - r^{(j)}\|^2_{B_b^{-1}} + \frac{1}{2}\sum_{i=0}^{N}\|H_i^{(j)}\delta x_i - d_i^{(j)}\|^2_{R_i^{-1}} + \frac{1}{2}\sum_{i=1}^{N}\|\delta q_i - c_i^{(j)}\|^2_{Q_i^{-1}}, \tag{2}$$

where

$$r^{(j)} = x_b - x_0^{(j)}, \quad c_i^{(j)} = -q_i^{(j)}, \quad d_i^{(j)} = y_i - \mathcal{H}_i(x_i^{(j)}).$$

In the quadratic cost function (2), $H_i^{(j)}$ represents the Jacobian matrix of the observation operator $\mathcal{H}_i$ at $x_i^{(j)}$, and δq_i is defined by $\delta x_i = M_i^{(j)}\delta x_{i-1} + \delta q_i$, where $M_i^{(j)}$ represents the Jacobian matrix of the physical model $\mathcal{M}_i$ at $x_{i-1}^{(j)}$. The state is then updated according to $x^{(j+1)} = x^{(j)} + \delta x$ where $\delta x = [\delta x_0, \delta x_1, \ldots, \delta x_N]^{\mathrm{T}}$. We can rewrite the quadratic subproblem (2) in a more compact

form as

$$\mathcal{J} = \tfrac{1}{2}\|\delta p - b\|^2_{D^{-1}} + \|HF\delta p - d\|^2_{R^{-1}}, \tag{3}$$

where we have dropped the outer loop index j for sake of clarity. In Equation (3), given $\ell = n \times (N+1)$ and $m = \sum_{i=0}^{N} m_i$, the vectors $\delta p \in \mathbb{R}^\ell$, $d \in \mathbb{R}^m$ and $b \in \mathbb{R}^\ell$ are defined as

$$\delta p = \begin{pmatrix} \delta x_0 \\ \delta q_1 \\ \vdots \\ \delta q_N \end{pmatrix}, \quad d = \begin{pmatrix} d_0 \\ d_1 \\ \vdots \\ d_N \end{pmatrix}, \quad b = \begin{pmatrix} r \\ c_1 \\ \vdots \\ c_N \end{pmatrix},$$

and the matrices $D \in \mathbb{R}^{\ell\times\ell}$, $R \in \mathbb{R}^{m\times m}$, $H \in \mathbb{R}^{m\times\ell}$ are given by

$$D = \mathrm{diag}(\mathrm{B_b}, \mathrm{Q}_1, \ldots, \mathrm{Q_N}), \quad \mathrm{R} = \mathrm{diag}(\mathrm{R}_0, \ldots, \mathrm{R_N}), \quad \mathrm{H} = \mathrm{diag}(\mathrm{H}_0, \ldots, \mathrm{H_N}).$$

The vectors δx and δp are related by $\delta x = F\delta p$ with F a ℓ by ℓ matrix defined as

$$F = \begin{pmatrix} I_n & & & & \\ M_{1,1} & I_n & & & \\ M_{1,2} & M_{2,2} & I_n & & \\ \vdots & \vdots & \ddots & \ddots & \\ M_{1,N} & M_{2,N} & \cdots & M_{N,N} & I_n \end{pmatrix},$$

where $M_{i,\bar{i}} = M_{\bar{i}} \cdots M_i$ represents an integration of the linear model from time t_{i-1} to time $t_{\bar{i}}$. The entries of this matrix are not available explicitly and the computation of matrix–vector products is performed through an operator. We stress that these matrix–vector products are very expensive since they require expensive model integrations.

Minimizing $\mathcal{J}$ defined in Equation (3) using iterative methods involves the computation of matrix–vector products with F, which requires *sequential* model integrations. Hence, since the matrix–vector products with $M_{i,\bar{i}}$ are known to be expensive, the sequential integrations will be very costly in this approach. An alternative method referred to as the 'saddle point approach' in [34] is briefly explained next, since it will be the basis for further developments.

2.2 *Solution via a sequence of quadratic programming problems*

The saddle point formulation of the weak-constrained 4D-Var provides a framework for allowing parallelization in the time domain. This is a key issue when tackling large-scale problems in data assimilation on modern architectures. In this section, we briefly present this algorithm and refer the reader to [34] for a complete description.

We can simply reformulate the nonlinear problem (1) as a the minimum of a convex quadratic function under nonlinear equality constraints as

$$f(p, w) = \tfrac{1}{2}\|p\|^2_{D^{-1}} + \tfrac{1}{2}\|w - y\|^2_{R^{-1}} \tag{4}$$

subject to

$$p_i = x_i - \mathcal{M}_i(x_{i-1}), (i = 1, \ldots, N),$$
$$w_i = \mathcal{H}_i(x_i), (i = 0, 1, \ldots, N),$$

where $p = [x_0 - x_b, p_1, \ldots, p_N]^{\mathrm{T}}$, $w = [w_0, w_1, \ldots, w_N]^{\mathrm{T}}$, and $y = [y_0, y_1, \ldots, y_N]^{\mathrm{T}}$. This constrained minimization problem can be solved by using a sequential quadratic programming

that represents a mapping from the state space to the observation space, B_b, R_i and Q_i are $n \times n$, $m_i \times m_i$ and $n \times n$ symmetric positive definite error covariance matrices corresponding to the background, observation and model errors, respectively. The first term in this nonlinear least squares problem is the Tikhonov regularization term [5, p. 101] which is often needed when the problem is under-determined, as it is the case in practical applications when there are fewer observations than variables in the model.

The formulation of the weighted nonlinear cost function (1) is known in the data assimilation community as the weak-constraint four-dimensional variational data assimilation (weak-4D-Var) problem [41,45]. This approach has its origin in the maximum likelihood estimation under the assumptions that all errors are unbiased, uncorrelated, and can be represented by zero mean Gaussian distributions [41]. In the nonlinear cost function (1), the model and observation errors are also assumed to be temporally uncorrelated. Time correlations can be included at the cost of using block-structured covariance matrices, provided that the required statistics is known [42]. We also note that, in practical applications, the Gaussian assumption may not be always realistic and can be partially relaxed [21,22,40]. The salient properties of this 4D-Var problem in real-life applications such as ocean or atmospheric data assimilation can be summarized as follows

- The dimension of the state variable n exceeds 10^6 in general. Hence, this problem can be really considered as a large-scale nonlinear least squares problem.
- The integrations of the physical model involving $\mathcal{M}_i$ are typically very costly. For this reason, exact second order derivative information is too expensive to compute.
- The error covariance matrices are not explicitly available. They are modelled or estimated using either smoothing functions or filters [9,44], and only their actions on a vector can be computed.

Taking into consideration these main properties, we review in Sections 2.1 and 2.2 practical algorithms for the solution of the large-scale nonlinear least squares problem given by (1).

2.1 *Solution via the GN method*

A widely used algorithm to minimize the large-scale nonlinear least squares cost function (1) is the GN method known in the data assimilation community as Incremental 4D-Var [10]. At each iteration (say the jth iteration) of the GN method, named as the outer loop, a step (an increment) $\delta x^{(j)}$ from a given $x^{(j)}$ is computed by minimizing a quadratic cost function which is the linearized least squares approximation of the nonlinear problem. This quadratic approximation in the neighbourhood of the iterate $x^{(j)}$ is given by

$$\mathcal{J}^{(j)} = \frac{1}{2}\|\delta x_0 - r^{(j)}\|^2_{B_b^{-1}} + \frac{1}{2}\sum_{i=0}^{N}\|H_i^{(j)}\delta x_i - d_i^{(j)}\|^2_{R_i^{-1}} + \frac{1}{2}\sum_{i=1}^{N}\|\delta q_i - c_i^{(j)}\|^2_{Q_i^{-1}}, \tag{2}$$

where

$$r^{(j)} = x_b - x_0^{(j)}, \quad c_i^{(j)} = -q_i^{(j)}, \quad d_i^{(j)} = y_i - \mathcal{H}_i(x_i^{(j)}).$$

In the quadratic cost function (2), $H_i^{(j)}$ represents the Jacobian matrix of the observation operator $\mathcal{H}_i$ at $x_i^{(j)}$, and δq_i is defined by $\delta x_i = M_i^{(j)}\delta x_{i-1} + \delta q_i$, where $M_i^{(j)}$ represents the Jacobian matrix of the physical model $\mathcal{M}_i$ at $x_{i-1}^{(j)}$. The state is then updated according to $x^{(j+1)} = x^{(j)} + \delta x$ where $\delta x = [\delta x_0, \delta x_1, \ldots, \delta x_N]^{\mathrm{T}}$. We can rewrite the quadratic subproblem (2) in a more compact

As mentioned in [15], this approach is a particular case of the simultaneous analysis and design method (SAND) [4]. We note that the SAND formulation differs from the one introduced in [15] by the presence of an additional constraint in the observation space. The crucial property of the SAND approach is that it already offers a first level of parallelism in the time domain. Indeed, matrix–vector products with

$$F^{-1} = \begin{pmatrix} I_n & & & & \\ -M_1 & I_n & & & \\ & -M_2 & I_n & & \\ & & \ddots & \ddots & \\ & & & -M_N & I_n \end{pmatrix}$$

can be obtained by performing the integrations of the linear model in *parallel*. Since the model integrations are known to be expensive, this approach represents a crucial step towards higher computational efficiency. In the SAND formulation (5), by defining additional constraints on the observation operator, we introduce additional parallelism in H and H^{T} operators. Nevertheless, the dimension of the global problem $(3\ell + 2m)$ has significantly increased. Hence, we consider the application of Schur complement reduction techniques to find the solution in a space of reduced dimension. Using the following partitioning of unknowns in (6)

$$\left(\begin{array}{ccccc} D^{-1} & 0 & 0 & I & 0 \\ 0 & R^{-1} & 0 & 0 & I \\ \hline 0 & 0 & 0 & -F^{-T} & -H^{\mathrm{T}} \\ I & 0 & -F^{-1} & 0 & 0 \\ 0 & I & -H & 0 & 0 \end{array}\right) \left(\begin{array}{c} \delta p \\ \delta w \\ \hline \lambda \\ \mu \\ \delta x \end{array}\right) = \left(\begin{array}{c} D^{-1}b \\ R^{-1}d \\ \hline 0 \\ 0 \\ 0 \end{array}\right),$$

a reduced system of order $(2\ell + m)$ involving the negative Schur complement is obtained as

$$\begin{pmatrix} D & 0 & F^{-1} \\ 0 & R & H \\ F^{-T} & H^{\mathrm{T}} & 0 \end{pmatrix} \begin{pmatrix} \lambda \\ \mu \\ \delta x \end{pmatrix} = \begin{pmatrix} b \\ d \\ 0 \end{pmatrix}. \tag{7}$$

The order of this reduced system is still large. Nevertheless, this formulation preserves the parallelization in the time domain. As a summary, the solution of the minimization problem (5) can be obtained by solving the saddle point system (7), which is then used to update the state vector x. In the remainder of this paper, we focus on efficient preconditioned iterative methods for the solution of the saddle point system given by (7).

2.2.1 *Solution of the linearized subproblem.*

We consider preconditioned Krylov subspace methods for the solution of (7) and refer the reader to [1] for a survey of numerical methods related to the solution of indefinite linear systems. First, we point out an important feature of our problem that we need to take into consideration when choosing an appropriate preconditioner.

We write the saddle point matrix in (7) in a compact form as

$$\mathcal{A} = \begin{pmatrix} D & 0 & F^{-1} \\ 0 & R & H \\ F^{-T} & H^{\mathrm{T}} & 0 \end{pmatrix} = \begin{pmatrix} A & B^{\mathrm{T}} \\ B & 0 \end{pmatrix}, \tag{8}$$

with $A \in \mathbb{R}^{(\ell+m)\times(\ell+m)}$ and $B \in \mathbb{R}^{\ell\times(\ell+m)}$, respectively. The computational costs related to the matrix–vector products with $\mathcal{A}$ in Equation (8) differ from most of the situations studied in the literature, in the sense that performing matrix–vectors with B is expensive, while computations involving A are relatively cheap. Therefore, many of the well-established preconditioners [1,24,28] are not appropriate in our particular setting. Hence, we focus our attention on the *inexact constraint preconditioner* $\mathcal{P}$ [2,3], that is,

$$\mathcal{P} = \begin{pmatrix} A & \tilde{B}^{\mathrm{T}} \\ \tilde{B} & 0 \end{pmatrix} = \begin{pmatrix} D & 0 & \tilde{F}^{-1} \\ 0 & R & \tilde{H} \\ \tilde{F}^{-\mathrm{T}} & \tilde{H}^{\mathrm{T}} & 0 \end{pmatrix}, \tag{9}$$

where $\tilde{B} \in \mathbb{R}^{\ell\times(\ell+m)}$ is a full row rank approximation of B, $\tilde{F} \in \mathbb{R}^{\ell\times\ell}$, $\tilde{H} \in \mathbb{R}^{m\times\ell}$, approximations of F and H, respectively. The simple choice

$$\mathcal{P}_1 = \begin{pmatrix} D & 0 & \tilde{F}^{-1} \\ 0 & R & 0 \\ \tilde{F}^{-\mathrm{T}} & 0 & 0 \end{pmatrix} \quad \text{with} \quad \tilde{F}^{-1} = \begin{pmatrix} I_n & & & \\ -I_n & I_n & & \\ & -I_n & I_n & \\ & & \ddots & \ddots \\ & & & -I_n \quad I_n \end{pmatrix} \tag{10}$$

has been found in [14] to provide good results in terms of rate of convergence of preconditioned Krylov subspace method.

Therefore, this inexact constraint preconditioner can be used as a preconditioner that we will call a first-level preconditioner. Our next goal will be to improve it by updating techniques, that will be called second-level preconditioning techniques in what follows.

3. Second-level preconditioning for the saddle point approach

In this section, we are interested in further improving an initial constraint preconditioner $\mathcal{P}_1$ so that the linear system at the $(j+1)$-th nonlinear iteration reads

$$\mathcal{P}_{j+1}^{-1}\mathcal{A}_{j+1}u_{j+1} = \mathcal{P}_{j+1}^{-1}f_{j+1}, \quad \text{with} \quad \mathcal{A}_{j+1} = \begin{pmatrix} A & B_{j+1}^{\mathrm{T}} \\ B_{j+1} & 0 \end{pmatrix}, \quad \mathcal{P}_{j+1} = \begin{pmatrix} A & \tilde{B}_{j+1}^{\mathrm{T}} \\ \tilde{B}_{j+1} & 0 \end{pmatrix}, \tag{11}$$

with

$$\delta y_{j+1} = \begin{pmatrix} \lambda_{j+1} \\ \mu_{j+1} \end{pmatrix}, \quad u_{j+1} = \begin{pmatrix} \delta y_{j+1} \\ \delta x_{j+1} \end{pmatrix}, \quad f_{j+1} = \begin{pmatrix} b_{j+1} \\ d_{j+1} \\ 0 \end{pmatrix}.$$

$\mathcal{P}_{j+1}$ is built from pairs of vectors collected during the iterative solution of the linear systems arising in the previous nonlinear iterations. This idea has been well explored on sequences of *symmetric positive definite* linear systems coming from GN iterations and has proved effective on large-scale applications [18,27]. While constructing a new preconditioner for the $(j+1)$th iteration, we must take into consideration the features of the saddle point system (8) as explained above, that is, computations involving the matrix B_{j+1} are expensive, while computations involving A are relatively cheap. Therefore, in Sections 3.2 and 3.3, we consider the class of inexact constraint preconditioners and focus on specific updates of the form

$$\mathcal{P}_{j+1} = \mathcal{P}_j + \begin{pmatrix} 0 & \Delta\tilde{B}^{\mathrm{T}} \\ \Delta\tilde{B} & 0 \end{pmatrix},$$

with $\Delta\tilde{B} \in \mathbb{R}^{\ell\times(\ell+m)}$.

3.1 *The direct and adjoint secant equations*

Given a positive integer k and a set of vectors $(\delta y_i, \delta x_i)^{\mathrm{T}}$ $(i = 1, \ldots, k)$, respectively, we define

$$\begin{pmatrix} A & B_{j+1}^{\mathrm{T}} \\ B_{j+1} & 0 \end{pmatrix} \begin{pmatrix} \delta y_i \\ \delta x_i \end{pmatrix} = \begin{pmatrix} g_i \\ h_i \end{pmatrix}, \quad (i = 1, \ldots, k). \tag{12}$$

We impose that $\mathcal{P}_{j+1}$ satisfies the k relations

$$\begin{pmatrix} A & \tilde{B}_{j+1}^{\mathrm{T}} \\ \tilde{B}_{j+1} & 0 \end{pmatrix} \begin{pmatrix} \delta y_i \\ \delta x_i \end{pmatrix} = \begin{pmatrix} g_i \\ h_i \end{pmatrix}, \quad (i = 1, \ldots, k). \tag{13}$$

Relations (12) and (13) simply lead to

$$\tilde{B}_{j+1}^{\mathrm{T}} \delta x_i = g_i - A\delta y_i, \quad (i = 1, \ldots, k), \tag{14}$$

$$\tilde{B}_{j+1} \delta y_i = h_i, \quad (i = 1, \ldots, k). \tag{15}$$

We call Equations (14) and (15) the *adjoint secant equation* and the *direct secant equation*, respectively, in analogy with the corresponding notions in quasi-Newton methods. Since $\tilde{B}_{j+1} = \tilde{B}_j + \Delta\tilde{B}$, we obtain from Equations (14) and (15) the set of $2k$ relations

$$\Delta\tilde{B}^{\mathrm{T}} \delta x_i = g_i - A\delta y_i - \tilde{B}_j{}^{\mathrm{T}} \delta x_i := \tilde{p}_i, \tag{16}$$

$$\Delta\tilde{B}\delta y_i = h_i - \tilde{B}_j \delta y_i := \tilde{q}_i. \tag{17}$$

To satisfy all these relations at once, we consider the following multiple secant equations

$$\Delta\tilde{B}^{\mathrm{T}} X = P, \tag{18}$$

$$\Delta\tilde{B} Y = Q, \tag{19}$$

where $X \in \mathbb{R}^{\ell \times k}$, $P \in \mathbb{R}^{(\ell+m)\times k}$, $Y \in \mathbb{R}^{(\ell+m)\times k}$ and $Q \in \mathbb{R}^{\ell \times k}$ defined as

$$\begin{aligned} X &= [\delta x_1, \ldots, \delta x_k], \\ P &= [\tilde{p}_1, \ldots, \tilde{p}_k], \\ Y &= [\delta y_1, \ldots, \delta y_k], \\ Q &= [\tilde{q}_1, \ldots, \tilde{q}_k], \end{aligned} \tag{20}$$

are supposed to be of full column rank. The consistency of these multiple secant equations depends on a condition relating X,Y,P and Q, as stated in the next lemma.

LEMMA 3.1 *Suppose that $X \in \mathbb{R}^{\ell \times k}$, $P \in \mathbb{R}^{(\ell+m)\times k}$, $Y \in \mathbb{R}^{(\ell+m)\times k}$ and $Q \in \mathbb{R}^{\ell \times k}$ defined in* (20) *are of full column rank. A necessary and sufficient condition for the linear systems defined in Equations* (18) *and* (19) *to be consistent is*

$$P^{\mathrm{T}} Y = X^{\mathrm{T}} Q. \tag{21}$$

Proof For sake of readability, the proof is postponed to the appendix. ■

Finding the update $\Delta\tilde{B}$, that is, solving the secant equations given by (18) and (19) simultaneously, is not a very well covered area in the literature. Most of the studies focus on the case

of square matrices and use only one secant equation at a time. To the best of our knowledge, two-sided equations are only considered in [19,20], to obtain an approximation of the Jacobian matrix. In this study, we rather use the approximation for preconditioning, which we believe is new. Throughout the paper, we aim at finding certain solutions of the two-sided equations given by (18) and (19). We especially focus on the rank-k and rank-$2k$ cases, giving a strong emphasis on the following three properties:

- *Hereditary*: When the new update is repeatedly used in a sequence of consistent secant equations, the new update still satisfies the previous secant equations.
- *Transformation invariance*: The update is invariant with respect to linear transformations. For instance, in data assimilation, this can be a transformation of model variables to a set of control variables that are assumed to be uncorrelated [23]. When such a transformation occurs, this property ensures that the update maintains all the relations of the transformed problem that are initially satisfied.
- *Least-change*: The update has a least change characterization in terms of a particular matrix norm. In this study, we focus on the least change characterization in terms of a (possibly weighted) Frobenius norm.

3.2 *A rank-k update*

The proof in the appendix has derived a general form of the update $\Delta\tilde{B}$. Now, as in [19,20], we specifically analyse the case of a rank-k update. Indeed, provided that $P^{\mathrm{T}}Y$ is non-singular and $P^{\mathrm{T}}Y = X^{\mathrm{T}}Q$, a solution of both the adjoint and direct secant equations can be written as a rank-k update $\Delta\tilde{B}$ of the form

$$\Delta\tilde{B} = QZP^{\mathrm{T}}, \tag{22}$$

with $Z \in \mathbb{R}^{k\times k}$ being non-singular. This is stated in the next theorem.

THEOREM 3.2 *Given $X \in \mathbb{R}^{\ell\times k}$, $P \in \mathbb{R}^{(\ell+m)\times k}$, $Y \in \mathbb{R}^{(\ell+m)\times k}$ and $Q \in \mathbb{R}^{\ell\times k}$ of full column rank defined in* (20)*, suppose that relationship* (21) *holds and that $P^{\mathrm{T}}Y$ is non-singular . Then the rank-k matrix of the form* (22) *given by*

$$\Delta\tilde{B} = Q(P^{\mathrm{T}}Y)^{-1}P^{\mathrm{T}} \tag{23}$$

or

$$\Delta\tilde{B} = Q(X^{\mathrm{T}}Q)^{-1}P^{\mathrm{T}} \tag{24}$$

is a solution of the multiple secant equations (18) *and* (19)*.*

Proof The necessary and sufficient condition for the direct secant equation in Z

$$QZP^{\mathrm{T}}Y = Q \tag{25}$$

to have a solution [26, Ch. 2, Theorem 13] reads

$$QQ^{\dagger}Q(P^{\mathrm{T}}Y)^{\dagger}(P^{\mathrm{T}}Y) = Q,$$

where † denotes the Moore–Penrose pseudo-inverse. This relation effectively holds, since Q is supposed to be of full column rank (the relation $Q^{\dagger}Q = I_k$ is then satisfied) and $P^{\mathrm{T}}Y$ is non-singular . Hence a possible solution of (25) can be obtained as [26, Ch. 2, Theorem 13]

$$Z = Q^{\dagger}Q(P^{\mathrm{T}}Y)^{-1},$$

that is, $Z = (P^{\mathrm{T}}Y)^{-1}$, which completes the proof. We note that the adjoint secant equation $\Delta\tilde{B}^{\mathrm{T}}X = P$ holds due to Equation (21). ■

Remark 1 When $k = 1$, we recover the rank-one update derived in [19,20].

Remark 2 If the equality $P^{\mathrm{T}}Y = X^{\mathrm{T}}Q$ does not hold, the update (23) obviously solves only the direct multiple secant equations (19) exactly, whereas the update (24) solves only the adjoint multiple secant equations (18) exactly.

We have derived the rank-k block formula, when k pairs satisfy the multiple secant equations. Block formulas ensure that the secant equations do hold at once. Therefore, by construction, these updates satisfy the hereditary property. As an alternative to the block formula, one can recursively use the rank-1 update[1] as

$$\tilde{B}_{i+1} = \tilde{B}_i + \frac{(h_i - \tilde{B}_i \delta y_i)(g_i - A\delta y_i - \tilde{B}_i^{\mathrm{T}} \delta x_i)^{\mathrm{T}}}{(h_i - \tilde{B}_i \delta y_i)^{\mathrm{T}} \delta x_i}, \quad \tilde{B}_1 = \tilde{B}_{(j=0)} \tag{26}$$

to obtain $\tilde{B}_{j+1} = \tilde{B}_{k+1}$, where we have assumed that $(h_i - \tilde{B}_i \delta y_i)^{\mathrm{T}} \delta x_i \neq 0$ for $i = 1, \ldots, k$. This rank-1 update is actually used in [19] to update the Jacobian matrix in the framework of the full quasi-Newton method. It is referred as to the two-sided rank one update (TR1) update. A similar rank-1 update is also used in [20]. The properties of the TR1 update can be summarized as follows:

- The TR1 update generalizes the classical symmetric rank one (SR1) update [7, 11], in the sense that this update reduces to the SR1 update, when $\tilde{B}_{j+1}$ is symmetric.
- When the secant equations are consistent, it is shown in [19] that the TR1 recursive formula satisfies the *hereditary property*, (i.e. it maintains the validity of all previous secant equations), and is *invariant under linear transformations*.
- The TR1 update has no obvious least change characterization.

We note that, in general, the updates obtained from the recursive formula and the block formula are not necessarily equivalent [8]. However, concerning the TR1 update, when the secant equations are consistent, both block and recursive formulas become equivalent. In practice, since it is easier to monitor the values of the denominator (to avoid blow up of the update) from the recursive formula involving vectors rather than matrices as in the block formula, recursive formula are often preferred to the block ones.

3.3 *A rank-2k update*

3.3.1 *Least-Frobenius norm update.*

In this section, as an alternative to the TR1 update proposed in Section 3.2, we present a new update $\Delta\tilde{B}$, which satisfies a least change characterization in terms of the Frobenius norm.

THEOREM 3.3 *Given $X \in \mathbb{R}^{\ell \times k}$, $P \in \mathbb{R}^{(\ell+m)\times k}$, $Y \in \mathbb{R}^{(\ell+m)\times k}$ and $Q \in \mathbb{R}^{\ell \times k}$ of full column rank defined in* (20) *, suppose that relationship* (21) *holds. Then the solution of the minimization problem*

$$\begin{aligned} \min_{\Delta\tilde{B} \in \mathbb{R}^{\ell \times (\ell+m)}} \quad & \|\Delta\tilde{B}\|_F \\ \text{subject to} \quad & \Delta\tilde{B}^{\mathrm{T}} X = P, \\ & \Delta\tilde{B} Y = Q, \end{aligned} \tag{27}$$

is given by

$$\Delta\tilde{B}_* = X^{T\dagger}P^{\mathrm{T}} + (I - XX^{\dagger})QY^{\dagger}, \tag{28}$$

or equivalently

$$\Delta\tilde{B}_* = (Y^{T\dagger}Q^{\mathrm{T}} + (I - YY^{\dagger})PX^{\dagger})^{\mathrm{T}}. \tag{29}$$

Proof Any rank-2k solution $\Delta\tilde{B}$ satisfying Equation (18) (the first constraint of the minimization problem (27)) can be written as [38, Lemma 2.1]

$$\Delta\tilde{B} = X^{\mathrm{T}\dagger}P^{\mathrm{T}} + (I - XX^{\dagger})C, \tag{30}$$

where $C \in \mathbb{R}^{\ell\times(\ell+m)}$. Inserting the relation (30) into the second constraint (Equation (19)) yields

$$X^{T\dagger}P^{\mathrm{T}}Y + (I - XX^{\dagger})CY = Q.$$

The least Frobenius norm solution of the minimization problem

$$\min_{C\in\mathbb{R}^{\ell\times(\ell+m)}} \|(Q - X^{T\dagger}P^{\mathrm{T}}Y) - (I - XX^{\dagger})CY\|_F$$

is then given by [38, Lemma 2.3]

$$C_* = (I - XX^{\dagger})^{\dagger}(Q - X^{T\dagger}P^{\mathrm{T}}Y)Y^{\dagger},$$

or equivalently

$$C_* = (I - XX^{\dagger})^{\dagger}QY^{\dagger}. \tag{31}$$

Since $(I - XX^{\dagger})$ defines an orthogonal projection, substituting Equation (31) into Equation (30) yields the solution of the minimization problem as

$$\Delta\tilde{B}_* = X^{T\dagger}P^{\mathrm{T}} + (I - XX^{\dagger})QY^{\dagger}.$$

This rank-2k update can be also written as

$$\Delta\tilde{B}_* = X^{T\dagger}P^{\mathrm{T}} + QY^{\dagger} - X(X^{\mathrm{T}}X)^{-1}X^{\mathrm{T}}QY^{\dagger}.$$

Using Equation (21), we deduce

$$\begin{aligned}
\Delta\tilde{B}_* &= X^{T\dagger}P^{\mathrm{T}} + QY^{\dagger} - X(X^{\mathrm{T}}X)^{-1}P^{\mathrm{T}}YY^{\dagger},\\
&= X^{T\dagger}P^{\mathrm{T}} + QY^{\dagger} - X^{T\dagger}P^{\mathrm{T}}YY^{\dagger},\\
&= QY^{\dagger} + X^{T\dagger}P^{\mathrm{T}}(I - YY^{\dagger}),\\
&= (Y^{T\dagger}Q^{\mathrm{T}} + (I - YY^{\dagger})PX^{\dagger})^{\mathrm{T}},
\end{aligned}$$

which completes the proof. ■

Remark 3 In the case of a single secant equation, Equation (30) simplifies into the following rank-2 update

$$\Delta\tilde{B}_* = \delta x^{T\dagger}p^{\mathrm{T}} + (I - \delta x\delta x^{\dagger})C, \tag{32}$$

where C is a rank-2 matrix. Let us now investigate the role of C in the geometrical properties of the update $\Delta\tilde{B}$. To do so, we denote $u = \delta x/\|\delta x\|_2$ and choose $U_\perp \in \mathbb{R}^{\ell\times(\ell-1)}$ such that

$U = [u, U_\perp]$ is an orthogonal matrix. Then Equation (32) can be written as

$$\Delta\tilde{B}_* = \frac{up^{\mathrm{T}}}{\|\delta x\|_2} + U_\perp U_\perp{}^{\mathrm{T}} C.$$

Therefore, a matrix–vector multiplication with $\Delta\tilde{B}_*$ leads to a vector in the column space of U. When C is chosen as the zero matrix, the resulting vector has no components in the direction of the column vectors of $U_\perp$, that is, $\Delta\tilde{B}_*^{\mathrm{T}} v = 0$ for any vector v orthogonal to u and the rank-one *Broyden update* is recovered. When $C = (I - \delta x \delta x^\dagger)\alpha q p^{\mathrm{T}}$, this yields the TR1 update. As shown in Theorem 3.3, choosing $C = (I - \delta x \delta x^\dagger)^\dagger q \delta y^\dagger$ results in a rank-two least-change update. Therefore, different choices of C define the information to be included from the directions orthogonal to the vector u.

We name the new update given in Theorem 3.3 as the *least-Frobenius two-sided rank-2k update* (FTR2). As an alternative to the block formulas, the FTR2 update can be recursively obtained from Equation (28) (with $k = 1$ and $\tilde{B}_1 = \tilde{B}_{(j=0)}$) as

$$\tilde{B}_{i+1} = \tilde{B}_i + \frac{\delta x_i (g_i - A\delta y_i - \tilde{B}_i^{\mathrm{T}} \delta y_i)^{\mathrm{T}}}{\delta x_i^{\mathrm{T}} \delta x_i} + \frac{(h_i - \tilde{B}_i \delta y_i)\delta y_i^{\mathrm{T}}}{\delta y_i^{\mathrm{T}} \delta y_i} - \frac{\delta x_i \delta x_i^{\mathrm{T}} (h_i - \tilde{B}_i \delta y_i)\delta y_i^{\mathrm{T}}}{\delta x_i^{\mathrm{T}} \delta x_i \delta y_i^{\mathrm{T}} \delta y_i}. \tag{33}$$

The properties of the FTR2 update can be summarized as follows:

- The FTR2 update generalizes the Powell-symmetric-Broyden (PSB) update [33], in the sense that the PSB update is recovered, when $\tilde{B}_{j+1}$ is symmetric.
- As outlined before, the block FTR2 update satisfies the hereditary property. However, the *recursive FTR2* (*similar to the PSB update*) *does not satisfy the hereditary property*. Hence, it satisfies only the last direct and adjoint secant equations. When using new pairs, the previous secant equations do not hold any longer.
- The FTR2 update is affected by linear transformations on the variable domain.
- It is shown in Theorem 3.3 that the FTR2 update satisfies a least-change property with respect to the Frobenius norm.

3.3.2 *The weighted least-Frobenius norm update.*

In this section, we are interested in solving the minimization problem (27) in a weighted Frobenius norm, such that the update enjoys a transformation invariance property. The next theorem provides such an update.

THEOREM 3.4 *Given $X \in \mathbb{R}^{\ell\times k}$, $P \in \mathbb{R}^{(\ell+m)\times k}$, $Y \in \mathbb{R}^{(\ell+m)\times k}$ and $Q \in \mathbb{R}^{\ell\times k}$ of full column rank defined in* (20) *suppose that relationship* (21) *holds. Let $S \in \mathbb{R}^{\ell\times k}$ and $T \in \mathbb{R}^{(\ell+m)\times k}$ be such that $X^{\mathrm{T}}S$ and $Y^{\mathrm{T}}T$ are symmetric and positive definite matrices, respectively. Let $W_1 \in \mathbb{R}^{\ell\times\ell}$ and $W_2 \in \mathbb{R}^{(\ell+m)\times(\ell+m)}$ be non-singular matrices such that the relations $W_1 W_1^{\mathrm{T}} X = S$ and $W_2^{\mathrm{T}} W_2 Y = T$ hold. Then the solution of the minimization problem*

$$\begin{aligned} \min_{\Delta\tilde{B}\in\mathbb{R}^{\ell\times(\ell+m)}} \quad & \|W_1^{-1}\Delta\tilde{B}W_2^{-1}\|_F \\ \text{subject to} \quad & \Delta\tilde{B}^{\mathrm{T}} X = P, \\ & \Delta\tilde{B}Y = Q, \end{aligned} \tag{34}$$

is given by

$$\Delta\tilde{B}_* = S(X^{\mathrm{T}}S)^{-1}P^{\mathrm{T}} + (Q - S(X^{\mathrm{T}}S)^{-1}X^{\mathrm{T}}Q)(Y^{\mathrm{T}}T)^{-1}T^{\mathrm{T}}, \tag{35}$$

or equivalently

$$\Delta\tilde{B}_* = (T(Y^{\mathrm{T}}T)^{-1}Q^{\mathrm{T}} + (P - T(Y^{\mathrm{T}}T)^{-1}Y^{\mathrm{T}}P)(X^{\mathrm{T}}S)^{-1}S^{\mathrm{T}})^{\mathrm{T}}. \tag{36}$$

Proof We define $\hat{\Delta}\tilde{B} \in \mathbb{R}^{\ell\times(\ell+m)}$ as $\hat{\Delta}\tilde{B} = W_1^{-1}\Delta\tilde{B}W_2^{-1}$. Then the minimization problem (34) can be written as

$$\begin{aligned} \min_{\hat{\Delta}\tilde{B}\in\mathbb{R}^{\ell\times(\ell+m)}} \quad & \|\hat{\Delta}\tilde{B}\|_F \\ \text{subjectto} \quad & \hat{\Delta}\tilde{B}^{\mathrm{T}}W_1{}^{\mathrm{T}}X = W_2^{-T}P, \\ & \hat{\Delta}\tilde{B}W_2Y = W_1^{-1}Q. \end{aligned}$$

This problem is structurally identical to the minimization problem (27). Hence, using the update formula (28) in this context yields

$$\begin{aligned} \hat{\Delta}\tilde{B}_* &= (X^{\mathrm{T}}W_1)^{\dagger}(W_2^{-T}P)^{\mathrm{T}} + (I - W_1{}^{\mathrm{T}}X(W_1{}^{\mathrm{T}}X)^{\dagger})W_1^{-1}Q(W_2Y)^{\dagger}, \\ &= W_1{}^{\mathrm{T}}X(X^{\mathrm{T}}W_1W_1{}^{\mathrm{T}}X)^{-1}P^{\mathrm{T}}W_2^{-1} \\ &\quad + (I - W_1{}^{\mathrm{T}}X(X^{\mathrm{T}}W_1W_1{}^{\mathrm{T}}X)^{-1}X^{\mathrm{T}}W_1)W_1^{-1}Q(Y^{\mathrm{T}}W_2^{\mathrm{T}}W_2Y)^{-1}Y^{\mathrm{T}}W_2^{\mathrm{T}}. \end{aligned} \tag{37}$$

Substituting the relations

$$W_1{}^{\mathrm{T}}X = W_1^{-1}S, \tag{38}$$

$$W_2Y = W_2^{-T}T, \tag{39}$$

into Equation (37), we obtain

$$\begin{aligned} \hat{\Delta}\tilde{B}_* &= W_1^{-1}S(X^{\mathrm{T}}S)^{-1}P^{\mathrm{T}}W_2^{-1} + (W_1^{-1}Q - W_1^{-1}S(X^{\mathrm{T}}S)^{-1}X^{\mathrm{T}}Q)(Y^{\mathrm{T}}T)^{-1}T^{\mathrm{T}}W_2^{-1}, \\ &= W_1^{-1}[S(X^{\mathrm{T}}S)^{-1}P^{\mathrm{T}} + (Q - S(X^{\mathrm{T}}S)^{-1}X^{\mathrm{T}}Q)(Y^{\mathrm{T}}T)^{-1}T^{\mathrm{T}}]W_2^{-1}. \end{aligned}$$

From Equation (29), the solution of the minimization problem (34) can be written as

$$\begin{aligned} \hat{\Delta}\tilde{B}_* &= ((Y^{\mathrm{T}}W_2^{\mathrm{T}})^{\dagger}(W_1^{-1}Q)^{\mathrm{T}} + (I - W_2Y(W_2Y)^{\dagger})W_2^{-T}P(W_1{}^{\mathrm{T}}X)^{\dagger})^{\mathrm{T}}, \\ &= (W_2Y(Y^{\mathrm{T}}W_2^{\mathrm{T}}W_2Y)^{-1}Q^{\mathrm{T}}W_1^{-T} \\ &\quad + (I - W_2Y(Y^{\mathrm{T}}W_2{}^{\mathrm{T}}W_2Y)^{-1}Y^{\mathrm{T}}W_2{}^{\mathrm{T}})W_2^{-T}P(X^{\mathrm{T}}W_1W_1{}^{\mathrm{T}}X)^{-1}X^{\mathrm{T}}W_1)^{\mathrm{T}}. \end{aligned} \tag{40}$$

Substituting Equations (38) and (39) into Equation (40), we obtain

$$\begin{aligned} \hat{\Delta}\tilde{B}_* &= (W_2^{-T}T(Y^{\mathrm{T}}T)^{-1}Q^{\mathrm{T}}W_1^{-T}, \\ &\quad + (W_2^{-T}P - W_2^{-T}T(Y^{\mathrm{T}}T)^{-1}Y^{\mathrm{T}}P)(X^{\mathrm{T}}S)^{-1}S^{\mathrm{T}}W_1^{-T})^{\mathrm{T}}, \\ &= (W_2^{-T}[T(Y^{\mathrm{T}}T)^{-1}Q^{\mathrm{T}} + (P - T(Y^{\mathrm{T}}T)^{-1}Y^{\mathrm{T}}P)(X^{\mathrm{T}}S)^{-1}S^{\mathrm{T}}]W_1^{-T})^{\mathrm{T}}, \end{aligned}$$

which completes the proof. Due to Lemma 3.1, the formulas (35) and (36) are equivalent. ■

Remark 4 We note that $X^{\mathrm{T}}S$ and $Y^{\mathrm{T}}T$ may not be necessarily symmetric and positive definite for some given pairs S and T. However, the strategy proposed in [36] can be applied in order to find least-change perturbations ΔS and ΔT such that $X^{\mathrm{T}}(S + \Delta S)$ and $Y^{\mathrm{T}}(T + \Delta T)$ become symmetric and positive definite. Theorem 3.4 can then be applied straightforwardly.

We name the update given by formulas (35) or (36) as the *weighted least Frobenius norm two-sided rank-*$(2k)$ (WFTR2) update. We note that different choices of S and T (meaning different weighted Frobenius norms) lead to different updates.

As a special case (with $k=1$ and $\tilde{B}_1 = \tilde{B}_{(j=0)}$) of the block WFTR2 update, the recursive update can be obtained as

$$\tilde{B}_{i+1} = \tilde{B}_i + \frac{s_i(g_i - A\delta y_i - \tilde{B}_i^{\mathrm{T}}\delta x_i)^{\mathrm{T}}}{\delta x_i^{\mathrm{T}} s_i} + \frac{(h_i - \tilde{B}_i\delta y_i)t_i^{\mathrm{T}}}{\delta y_i^{\mathrm{T}} t_i} - \frac{s_i\delta x_i^{\mathrm{T}}(h_i - \tilde{B}_i\delta y_i)t_i{}^{\mathrm{T}}}{\delta x_i^{\mathrm{T}} s_i \delta y_i^{\mathrm{T}} t_i}. \tag{41}$$

The properties of the WFTR2 update can be summarized as follows

- The block WFTR2 update satisfies the hereditary property, whereas the recursive formula does not.
- Let us assume that we have applied the following change of variables

  $$\bar{X} = G^{-1}X \quad \text{and} \quad \bar{Y} = V^{-T}Y$$

  where G and V are non-singular matrices of order ℓ and $(\ell + m)$, respectively. Provided that the relations

  $$\bar{S} = G^{\mathrm{T}}S \quad \text{and} \quad \bar{T} = VT$$

  hold, it can be simply shown by induction that the WFTR2 update is *transformation-invariant*.
- It is shown in Theorem 3.4 that the WFTR2 update satisfies a *least-change property*.

In this section, we are left with the choice of S and of T in the WFTR2 update. We note that when

$$S = [h_1, \ldots, h_k], \tag{42}$$

$$T = [g_1 - A\delta y_1, \ldots, g_k - A\delta y_k], \tag{43}$$

with h_i and g_i $(i = 1, \ldots, k)$ defined in (13) and $\tilde{B}_{j+1}$ is a symmetric and positive definite matrix, the WFTR2 update reduces to the Davidon–Fletcher–Powell update [11,17]. Choosing $S = Q$ and $T = P$ leads to the TR1 update.

In our experiments, since the column vector matrices given by (42) are available as by-products of the Krylov subspace method, we consider to choose these pairs for the WFTR2 update.

So far, we have introduced different update formulas detailed in Section 3.2 and 3.3 whose properties for the recursive formulas are summarized in Table 1. Concerning the block formulas, the hereditary property holds whatever the update, whereas the other two properties remain similar to those given in Table 1.

Table 1. Properties of the different recursive updates of Sections 3.2 and 3.3.

Update type	Rank	Hereditary	Trans. invariance	Least-change
TR1 (Equation (26))	rank-1	yes	yes	no
FTR2 (Equation (33))	rank-2	no	no	yes
WFTR2 (Equation (41)) with pairs defined as (42)	rank-2	no	yes	yes

3.4 *The inverse of the second-level preconditioner*

In this section, we derive a practical formula concerning the inverse of the updated preconditioner to be used when solving the saddle point system (11). The block update formulas of Theorems 3.2–3.4 can be written in the following generic form

$$\Delta\tilde{B}_* = VZU^{\mathrm{T}}, \tag{44}$$

with $V \in \mathbb{R}^{\ell\times p}$, $Z \in \mathbb{R}^{p\times p}$ and $U \in \mathbb{R}^{(\ell+m)\times p}$, where p is equal to either k or $2k$, respectively. For instance, the update (35) of Theorem 3.4 reads

$$\Delta\tilde{B}_* = \begin{bmatrix} S & Q \end{bmatrix} \begin{bmatrix} -(X^{\mathrm{T}}S)^{-1}(X^{\mathrm{T}}Q)(Y^{\mathrm{T}}T)^{-1} & (X^{\mathrm{T}}S)^{-1} \\ (Y^{\mathrm{T}}T)^{-1} & 0 \end{bmatrix} \begin{bmatrix} T^{\mathrm{T}} \\ P^{\mathrm{T}} \end{bmatrix}. \tag{45}$$

Using Equation (44), the inexact constraint preconditioner $\mathcal{P}_2$ at the second nonlinear iteration can be thus updated as

$$\begin{aligned} \mathcal{P}_2 &= \mathcal{P}_1 + \begin{pmatrix} 0 & UZ^{\mathrm{T}}V^{\mathrm{T}} \\ VZU^{\mathrm{T}} & 0 \end{pmatrix}, \\ &= \mathcal{P}_1 + \underbrace{\begin{pmatrix} 0 & U \\ V & 0 \end{pmatrix}}_{\mathcal{T}} \underbrace{\begin{pmatrix} ZU^{\mathrm{T}} & 0 \\ 0 & Z^{\mathrm{T}}V^{\mathrm{T}} \end{pmatrix}}_{\mathcal{G}}, \end{aligned} \tag{46}$$

where $\mathcal{T} \in \mathbb{R}^{(2\ell+m)\times(2p)}$ and $\mathcal{G} \in \mathbb{R}^{(2p)\times(2\ell+m)}$, respectively. Using the Sherman–Morrison–Woodbury formula, we obtain the inverse of the second-level preconditioner as

$$\mathcal{P}_2^{-1} = \mathcal{P}_1^{-1} - \mathcal{P}_1^{-1}\mathcal{T}(I + \mathcal{G}\mathcal{P}_1^{-1}\mathcal{T})^{-1}\mathcal{G}\mathcal{P}_1^{-1}. \tag{47}$$

We note that this formula involves the application of the inverse of the first-level preconditioner $\mathcal{P}_1^{-1}$, which is assumed to be easily tractable. This is indeed the case when considering the inexact constraint preconditioner as detailed next.

3.5 *Construction and application of the second-level preconditioner*

Let us assume that the following set of vectors $(\delta x_i, \delta y_i, g_i, h_i)$, $(i = 1, \ldots, k)$ satisfying Equation (12) have been saved during the previous outer loop of the approximate sequence of quadratic programming problems, and that a first-level preconditioner $\mathcal{P}_1$ in the form of (9) is used. We present in Algorithm 1 both the setup phase and the application of the inverse of the second-level preconditioner on a given vector of appropriate dimension.

We want to discuss next a few questions related to the computational cost of the application of the first- and second-level preconditioners, respectively. The inverse of the first-level preconditioner (10) is given by

$$\mathcal{P}_1^{-1} = \begin{pmatrix} 0 & 0 & \tilde{F}^{\mathrm{T}} \\ 0 & R^{-1} & 0 \\ \tilde{F} & 0 & -\tilde{F}D\tilde{F}^{\mathrm{T}} \end{pmatrix} \quad \text{with} \quad \tilde{F} = \begin{pmatrix} I_n & & & & \\ I_n & I_n & & & \\ I_n & I_n & I_n & & \\ I_n & \ddots & \ddots & \ddots & \\ I_n & \ddots & I_n & I_n & I_n \end{pmatrix}. \tag{48}$$

Thus the action of $\mathcal{P}_1^{-1}$ requires one application of D, R^{-1}, $\tilde{F}$ and $\tilde{F}^{\mathrm{T}}$ in total. We note that applying $\tilde{F}$ or $\tilde{F}^{\mathrm{T}}$ is really cheap, since it only involves summations of vectors. Concerning the second-level preconditioner, we have decided to compute and store $\mathcal{G}$, $\mathcal{P}_1^{-1}\mathcal{T}$

Algorithm 1 Setup and application of the inverse of the second-level preconditioner

Setup phase

(1) Obtain P, Q, S and T from relations (16), (17), (42) and (43), respectively.
(2) Deduce $\mathcal{T} \in \mathbb{R}^{(2\ell+m)\times(2p)}$ and $\mathcal{G} \in \mathbb{R}^{(2p)\times(2\ell+m)}$ defined in (46) depending on the update formula.
(3) Compute $\mathcal{P}_1^{-1}\mathcal{T}$ and $(I_{2p} + \mathcal{G}\mathcal{P}_1^{-1}\mathcal{T})^{-1}$.

Application of the inverse of the second-level preconditioner on $r \in \mathbb{R}^{(2\ell+m)}$

(1) Apply the first-level preconditioner $s = \mathcal{P}_1^{-1}r$.
(2) Compute

$$\mathcal{P}_2^{-1}r = s - (\mathcal{P}_1^{-1}\mathcal{T})(I_{2p} + \mathcal{G}\mathcal{P}_1^{-1}\mathcal{T})^{-1}(\mathcal{G}\, s).$$

and $(I + \mathcal{G}\mathcal{P}_1^{-1}\mathcal{T})^{-1}$ in memory, respectively. Consequently, the action of the inverse of the second-level preconditioner in Equation (47) involves only a *single* application of the first-level preconditioner $\mathcal{P}_1^{-1}$ and a matrix–vector product with $\mathcal{G}$, $(I + \mathcal{G}\mathcal{P}_1^{-1}\mathcal{T})^{-1}$ and $\mathcal{P}_1^{-1}\mathcal{T}$, respectively (see Algorithm 1). By exploiting the structure of $\mathcal{G}$ and $\mathcal{T}$, the total additional storage corresponds to $3p$ vectors of length $(2\ell + m)$ and to $2p$ vectors of length $2p$. This amount can be considered as affordable, since p is assumed to be much smaller than ℓ in this study. Computing $\mathcal{P}_1^{-1}\mathcal{T}$ requires a single application of $\mathcal{P}_1^{-1}$ on a set of $2p$ vectors, while the computation of $\mathcal{G}\mathcal{P}_1^{-1}\mathcal{T}$ involves $8p^2(2\ell + m)$ additional floating point operations. Both computations have to be performed at each nonlinear iteration. Applying the second-level preconditioner is thus slightly more expensive than the first-level preconditioner. Nevertheless, in our setting, we note that the application of $\mathcal{A}$ is the most expensive task in the Krylov subspace method since it involves multiple calculations of Jacobians. This will be illustrated in Section 4.

3.6 *Spectrum of the preconditioned saddle point system*

We briefly comment on the spectrum analysis of the saddle point system preconditioned by the inexact constraint preconditioner (10). We have

$$\mathcal{P}_1^{-1}\mathcal{A} = \begin{pmatrix} \tilde{F}^{\mathrm{T}}F^{-\mathrm{T}} & \tilde{F}^{\mathrm{T}}H^{\mathrm{T}} & 0 \\ 0 & I & R^{-1}H \\ \tilde{F}D(I - \tilde{F}^{\mathrm{T}}F^{-\mathrm{T}}) & -\tilde{F}D\tilde{F}^{\mathrm{T}}H^{\mathrm{T}} & \tilde{F}F^{-1} \end{pmatrix}.$$

A direct application of Corollary 2.2 of [3] reveals that the eigenvalues of $\mathcal{P}_1^{-1}\mathcal{A}$ (denoted by $\lambda(\mathcal{P}_1^{-1}\mathcal{A})$) are either equal to one or bounded by

$$|\lambda(\mathcal{P}_1^{-1}\mathcal{A}) - 1| \leq \frac{\left\| \begin{bmatrix} (F^{-\mathrm{T}} - \tilde{F}^{-\mathrm{T}})D^{-1/2} & H^{\mathrm{T}}R^{-1/2} \end{bmatrix} \right\|_2}{\sigma_1 \left(\begin{bmatrix} \tilde{F}^{-\mathrm{T}}D^{-1/2} & 0 \end{bmatrix} \right)},$$

where $\sigma_1(A)$ denotes the smallest singular value of A. The last inequality is then equivalent to

$$|\lambda(\mathcal{P}_1^{-1}\mathcal{A}) - 1|^2 \leq \frac{\lambda_{\max}((F^{-\mathrm{T}} - \tilde{F}^{-\mathrm{T}})D^{-1}(F^{-1} - \tilde{F}^{-1}) + H^{\mathrm{T}}R^{-1}H)}{\lambda_{\min}(\tilde{F}^{-\mathrm{T}}D^{-1}\tilde{F}^{-1})}. \tag{49}$$

When $\tilde{F} = F$, the eigenvalues of $\mathcal{P}_1^{-1}\mathcal{A}$ are constrained to lie on the line $x = 1$ in the complex plane [16]. When F is approximated, numerical computations reveal that the eigenvalues do lie

on this line or belong to a disc of centre $(1, 0)$ in the complex plane. The bound (49) gives us an insight on how the approximation $\tilde{F}$ influences the radius of this disc.

We note that the inequality (49) is mainly of theoretical interest since the exact computation of the upper bound is out of reach for large-scale applications. In practice, we usually rely on the computation of Ritz or harmonic Ritz values to cheaply investigate the convergence of GMRES [43]. Numerical experiments in our setting have revealed that most of these values lie in the right-half plane, which is often found to be beneficial to the convergence of GMRES. Finally, we leave to a future study the spectral analysis of $\mathcal{P}_2^{-1}\mathcal{A}$, which we believe is outside the scope of the paper.

4. Numerical experiments

Our numerical experiments are performed by using a simple two-layer quasi-geostrophic model (QG-model) in the Object-Oriented Prediction System platform [34]. In this section, we first describe the two layer QG-model, then explain the numerical set-up and finally present the numerical results.

4.1 *A two-layer quasi-geostrophic model*

This section describes the simple two-layer quasi-geostrophic [2] (nearly geostrophic) channel model which is widely used in theoretical atmospheric studies, since it is simple enough for numerical calculations and adequately captures the most relevant large-scale dynamics in the atmosphere.

The two-layer quasi-geostrophic model equations are based on the non-dimensional quasi-geostrophic potential vorticity, whose evolution represents large-scale circulations of the atmosphere. The quasi-geostrophic potential vorticity on the first (upper) and second (lower) layers can be written, respectively, as

$$q_1 = \nabla^2\psi_1 - \frac{f_0^2L^2}{g'H_1}(\psi_1 - \psi_2) + \beta y, \tag{50}$$

$$q_2 = \nabla^2\psi_2 - \frac{f_0^2L^2}{g'H_2}(\psi_2 - \psi_1) + \beta y + R_s, \tag{51}$$

where ψ is the stream function, ∇^2 is the two-dimensional Laplacian operator, R_s represents orography or heating, β is the (non-dimensionalized) northward variation of the Coriolis parameter at the fixed latitude y, f_0 is the Coriolis parameter at the southern boundary of the domain. L is the typical length scale of the motion we wish to describe, H_1 and H_2 are the depths of the two layers, $g' = g\Delta\theta/\bar{\theta}$ is the reduced gravity where $\bar{\theta}$ is the mean potential temperature, and $\Delta\theta$ is the difference in potential temperature across the layer interface. Details of the derivation of these non-dimensional equations can be found in [13, 32]. The conservation of potential vorticity in each layer is thus described by

$$\frac{D_iq_i}{Dt} = 0, \quad i = 1, 2. \tag{52}$$

This evolution equation is given in terms of the advective (total) derivative D_i/Dt defined by

$$\frac{D_i}{Dt} = \frac{\partial}{\partial t} + u_i\frac{\partial}{\partial x} + v_i\frac{\partial}{\partial y},$$

where

$$u_i = -\frac{\partial \psi_i}{\partial y} \quad \text{and} \quad v_i = \frac{\partial \psi_i}{\partial x}, \tag{53}$$

are the horizontal velocity components at each layer. Therefore, the potential vorticity at each time step is determined by using the conservation of potential vorticity given by Equation (52). In this process, time stepping consists of a simple first order semi-Lagrangian advection of potential vorticity. Given the potential vorticity at fixed time, Equations (50) and (51) can be solved for the stream function at each grid point and then the velocity fields obtained through Equation (53). Periodic boundary conditions in the west-east direction and Dirichlet boundary conditions in the north-south direction are employed. For the experiments in this paper, we choose $L = 10^6$ m, $H_1 = 6000$ m, $H_2 = 4000$ m, $f_0 = 10^{-4}\text{s}^{-1}$ and $\beta = 1.5$. For more details on the model and its solution, we refer the reader to [34].

The domain for the experiments is 12,000 km by 6300 km for both layers. The horizontal discretization consists of 40×20 points, so that the east-west and the north-south resolution is approximately 300 km. The dimension of the state vector of the model is 1600.

4.2 *Experimental set-up*

Our numerical experiments are performed by using a two-layer model as described above. A reference stream function is generated from a model with layer depths of $H_1 = 6000$ m and $H_2 = 4000$ m, and the time step is set to 300 s, whereas the assimilating model has layer depths of $H_1 = 5500$ m and $H_2 = 4500$ m, and the time step is set to 3600 s. These differences in the layer depths and the time step provide a source of model error.

For all the experiments presented here, observations of the non-dimensional stream function, vector wind and wind speed were taken from the reference of the model at 100 points randomly distributed over both levels. Observations were taken every 12 hours. Observation errors were assumed to be independent from each others and uncorrelated in time, the standard deviations were chosen to be 0.4 for the stream function observation error, 0.6 for the vector wind and 1.2 for the wind speed. The observation operator is the bi-linear interpolation of the model fields to horizontal observation locations.

The background error covariance matrix (B_b matrix) and the model error covariances (matrices Q_i) used in these experiments correspond to vertical and horizontal correlations. The vertical and horizontal structures are assumed to be separable. In the horizontal plane covariance matrices correspond to isotropic, homogeneous correlations of stream function with Gaussian spatial structure obtained from a Fast Fourier Transform approach [12,30]. For the background error covariance matrix B_b, the standard deviation and the horizontal correlation length scale in this experiments are set to 0.8 and 10^6 m, respectively. For the model error covariance matrices Q_i, the standard deviation and the horizontal correlation length scale are set to 0.2 and 2×10^6 m, respectively. The vertical correlation is assumed to be constant over the horizontal grid and the correlation coefficient value between the two layers was taken as 0.5.

The length of the assimilation window is set to 24 hours. The assimilation window is divided into three sub-windows (sub-intervals). Thus, the length of the each subwindow is equal to 8 hours and the model is considered as sufficiently accurate within each subwindow. The control variable is defined as the initial state of each subwindow. We have performed 3 GN iterations, and 10 inner loops at each GN iteration. Inner loops are solved by using full GMRES [35].

4.3 *Numerical results*

We aim at solving the equality constrained least squares problem (4) by using our two-level preconditioning technique in which a sequence of saddle point systems (Equations (7)) is solved. The least squares problem solves for the stream function of the two layer quasi-geostraphic model as explained above by using a priori information x_b and observations y_i over the assimilation window (24 hours). Figure 1(a) shows the reference stream function (the truth) used to generate a priori information and observations, while Figure 1(b) shows a priori fields (forecast) for the stream function and potential vorticity that we have used in our experiments.

The computational times (in milliseconds) needed to apply the operators appearing in the saddle point system (7) are given in Table 2. These values have been obtained by running a numerical simulation on a single core of a desktop computer.

As outlined in Table 2, applying the Jacobian matrix of the QG-model $\mathcal{M}_i$ (at time t_i) is more expensive by at least a factor of 70 than applying the covariance matrices D and R. Hence, this shows that the computational cost related to the application of either the first- or second-level preconditioner is significantly much lower than the cost of a matrix–vector product with the saddle point matrix of (7).

We have performed numerical experiments by using both the two levels of preconditioning, that is, the inexact preconditioner given by (10), and the second-level preconditioners obtained from the TR1 and WFTR2 updates, respectively. When constructing the updates, the last 8 pairs coming from the previous outer loop are used. Figure 2 shows the objective function values along the iterations when using these preconditioners. As seen especially from Figure 2(b), the effect of the second-level preconditioners can be remarked only at the third outer loop.

The improvement on large problems when using updates of Broyden type in nonlinear optimization is strongly affected by an improved numerical stability. It is indeed shown in [37] that if the initial update matrix is unscaled, or scaled without knowledge of the actual magnitude of the elements of the true inverse Hessian, a severe loss of accuracy due to round-off errors can

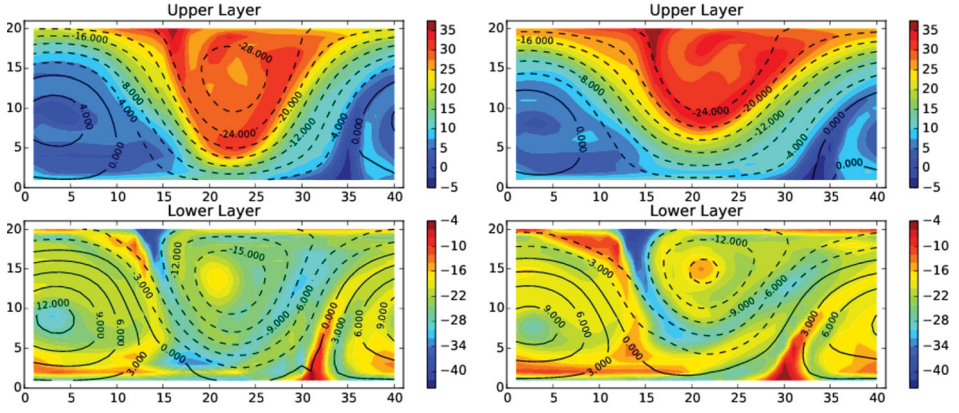
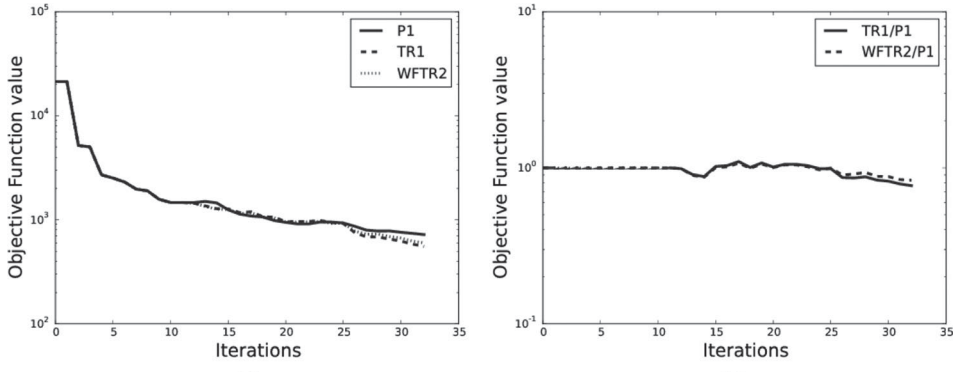


Figure 1. Stream function (top) and potential vorticity (bottom) values of (a) truth and (b) forecast (a priori) fields.

Table 2. Computational time (ms) associated with the application of the different operators appearing in Equation (7).

	B_b and Q_i	R	M_i and M_i^{T}	H_i and H_i^{T}
Time (ms)	0.11	0.017	7.9	0.19



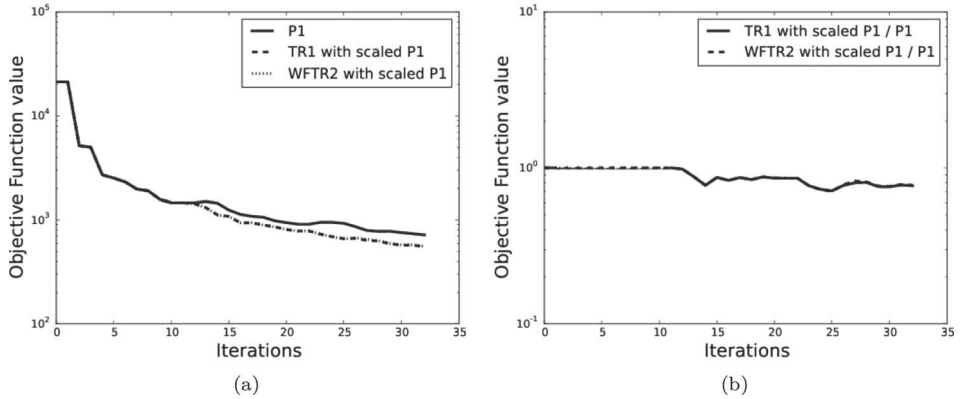
(a) (b)

Figure 2. (a) Objective cost function values along iterations for the first- and second-level preconditioners updated by using the TR1 or WFTR2 approaches. The first-level preconditioner is taken as the inexact constraint preconditioner $\mathcal{P}_1$. (b) Ratios of the cost function values associated with the second-level preconditioners to the cost function value associated with $\mathcal{P}_1$.

occur. Therefore, it is very common to use an initial scaling based on the inverse approximation of the Hessian to possibly improve the convergence [29,31,37,39]. One of the most successful choice for the initial scaling is shown to be an approximation of one of the eigenvalues of the inverse Hessian [29,31]. This choice actually attempts to make the range of the entries of the initial matrix similar to those of the inverse Hessian. With the same motivation in mind now for the initial preconditioner $\mathcal{P}_1$, we choose a scaling factor as

$$\alpha = \frac{\begin{pmatrix} \delta y_k \\ \delta x_k \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} g_k \\ h_k \end{pmatrix}}{\begin{pmatrix} g_k \\ h_k \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} g_k \\ h_k \end{pmatrix}},$$

with pairs satisfying Equation (12) (k being the index of the last pair obtained from the previous minimization). By using Equation (12) and defining $\delta w_k = \begin{pmatrix} g_k \\ h_k \end{pmatrix}$, α can be simply rewritten as a



(a) (b)

Figure 3. (a) Objective cost function values along iterations for the first- and second-level preconditioners updated by using the TR1 or WFTR2 approaches. The first-level preconditioner is taken as the scaled $\mathcal{P}_1$. (b) Ratios of the cost function values associated with the second-level preconditioners shown in (a) to the cost function value associated with $\mathcal{P}_1$.

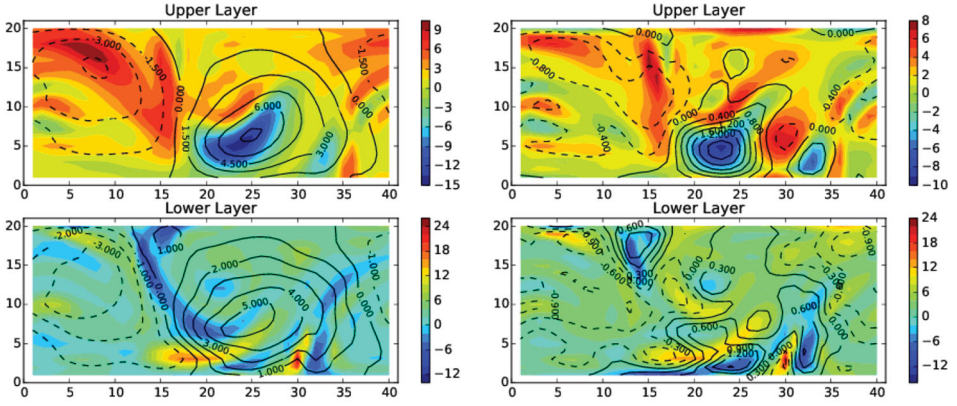
Rayleigh quotient

$$\alpha = \frac{\delta w_k^{\mathrm{T}} \mathcal{A}_{j+1}^{-1} \delta w_k}{\delta w_k{}^{\mathrm{T}} \delta w_k}.$$

Therefore, this scaling factor attempts to make the magnitude of entries of $\mathcal{P}_1$ closer to those of $\mathcal{A}_{j+1}^{-1}$. Figure 3 shows the numerical results when the scaled preconditioner is used in our experiments. In this setting, the second-level preconditioners constructed from the scaled initial preconditioner accelerate the convergence also at the second outer loop. We stress that, for a large-scale data assimilation system, this gain is very crucial (here approximately 30%), since the cost of each iteration is very large (see Table 2). Thus, the role of the scaling factor for

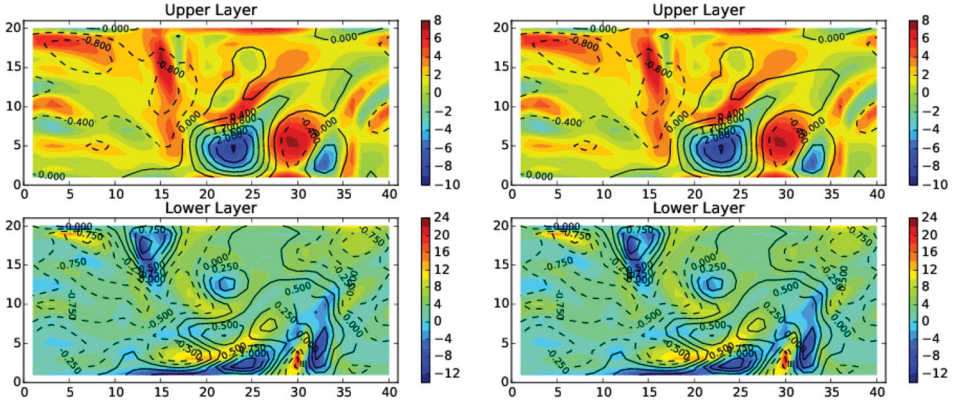
Table 3. RSE values for the forecast field and solutions obtained by using different preconditioners.

x	RSE value
A priori	≈ 2.53
Solution with $\mathcal{P}_1$	≈ 0.58
Solution with TR1	≈ 0.51
Solution with WFTR2	≈ 0.51



(a) Truth - *A priori*

(b) Truth - solution with $\mathcal{P}_1$



(c) Truth - solution with TR1

(d) Truth - solution with WFTR2

Figure 4. Difference between the true solution and a priori (a), solution obtained with $\mathcal{P}_1$ (b), solution obtained with TR1 update (c), solution obtained with WFTR2 update (d), respectively.

the initial preconditioner may be quite significant in practical applications, since the operational 4D-Var is typically performed only with two or three outer loops.

We want to mention that the performance of the second-level preconditioners is subject to small changes arising in the sequence of saddle point systems. Taking larger steps in an approximate sequence of quadrating problems changes the information crucially in the next saddle point system which prevents from using inherited information for the new system. Since in practical applications, only a limited number of inner iterations can be performed, we usually face a slowly varying sequence of saddle point systems which explains why these preconditioners are efficient in our experiments.

Examining the root square error (RSE) from the truth at the initial time given by

$$\mathrm{RSE} = \sqrt{\frac{1}{n}(\mathrm{truth} - x_0^*)^{\mathrm{T}}(\mathrm{truth} - x_0^*)},$$

we notice that all preconditioners are able to reduce the error for a given number of iterations, the second-level preconditioners leading to a further reduction in the error (see Table 3). Finally, the difference fields are shown in Figure 4. We observe that both second-level preconditioners lead to very similar results for these experiments.

5. Conclusions

Geophysical inverse problems where a state vector is estimated using physical observations are very common in the simulation of complex systems such as those arising from meteorology or oceanography. In this study, we focus on the estimation of the state vectors at different times over an assimilation window in which the observations are assimilated. This process is known as a data assimilation problem. When perfect models are used for the dynamics, the estimation process leads to a large-scale unconstrained problem where the constraints are directly inserted into the objective function. The solution method is then based on a truncated GN technique. When accounting for model errors, this approach is known to be difficult to parallelize in the time dimension which makes it unsuitable for modern parallel computer architectures.

It has been recently shown that a saddle point formulation of the estimation process can introduce highly parallel algorithms in the time dimension. In this paper, we present this formulation that involves a sequence of QP problems, in which the constraints represent both the system dynamics and the observation operator. Each QP results in a linear system where the matrix has the usual saddle point structure.

Efficient iterative solution algorithms for saddle point systems strongly rely on the availability of efficient preconditioners. The structure of the saddle point system arising in our application is markedly different from those usually considered in the literature, in the sense that the computational costs of matrix–vector products with the constraint blocks is much more important than those related to the diagonal blocks. For our preconditioning strategy we proposed in this paper to start with a low-cost first-level preconditioner for the first QP, and to further improve this preconditioner along the minimization process by using inherited information form earlier QP's. Due to the particular nature of our problem, we focus on the limited memory updates for such blocks. One interesting property of these blocks is that they are rectangular matrices, which is again a case that is less covered in the literature than the standard square case.

It is shown that these updates can be derived as a solution of the two-sided secant equations, namely *adjoint* and *direct secant equations*. We have also shown on numerical simulations that these second-level preconditioners do improve the convergence and that they are promising when

solving either the sequence of saddle point systems with multiple right-hand sides or the sequence of slowly varying systems.

We are aware that open issues still need to be addressed. An important complementary aspect is related to the analysis of the properties of the resulting preconditioned operator to characterize the convergence of the Krylov subspace method. Another aspect is to exploit globalization strategies such as trust region methods or line searches strategies. Both issues are left as future lines of research.

Acknowledgments

The authors wish to thank the two anonymous referees for their constructive comments which helped to improve the manuscript.

Disclosure statement

No potential conflict of interest was reported by the authors.

Notes

1. Since the derivations related to the update formula (23) are quite similar, we omit the corresponding formula.
2. Quasi-geostrophic motion means that, in the horizontal direction of the atmospheric flow, the Coriolis force caused by the rotation of the Earth, and the pressure gradient force are in approximate balance.

References

[1] M. Benzi, G.H. Golub, and J. Liesen, *Numerical solution of saddle point problems*, Acta Numer. 14 (2005), pp. 1–137.
[2] L. Bergamaschi, J. Gondzio, M. Venturin, and G. Zilli, *Inexact constraint preconditioners for linear systems arising in interior point methods*, Comput. Optim. Appl. 36 (2007), pp. 136–147.
[3] L. Bergamaschi, J. Gondzio, M. Venturin, and G. Zilli, *Erratum to: Inexact constraint preconditioners for linear systems arising in interior point methods*, Comput. Optim. Appl. 49 (2011), pp. 401–406.
[4] L. Biegler and A. Wächter, *DAE-constrained optimization*, SIAG/OPT Views-and-News. 14 (2003), pp. 10–15.
[5] Å. Björck, *Numerical Methods for Least Squares Problems*, SIAM, Philadelphia, 1996.
[6] J.H. Bramble and J.E. Pasciak, *A preconditioning technique for indefinite systems resulting from mixed approximations of elliptic problems*, Math. Comp. 50 (1988), pp. 1–17.
[7] C.G. Broyden, *Quasi-Newton methods and their application to function minimization*, Math. Comp. 21 (1967), pp. 368–381.
[8] R.H. Byrd, J. Nocedal, and R.B. Schnabel, *Representations of quasi-Newton matrices and their use in limited memory methods*, Math. Program. 63 (1994), pp. 129–156.
[9] P. Courtier, E. Andersson, W. Heckley, D. Vasiljevic, M. Hamrud, A. Hollingsworth, F. Rabier, M. Fisher, and J. Pailleux, *The ECMWF implementation of three-dimensional variational assimilation (3D-Var). i: Formulation*, Q. J. R. Meteorol. Soc. 124 (1998), pp. 1783–1807.
[10] P. Courtier, J.N. Thépaut, and A. Hollingsworth, *A strategy for operational implementation of 4D-Var using an incremental approach*, Q. J. R. Meteorol. Soc. 120 (1994), pp. 1367–1388.
[11] W.C. Davidon, *Variable metric method for minimization*, Tech. Rep. ANL-5990, Argonne National Laboratory, 1959.
[12] C.R. Dietrich and G.N. Newsam, *Fast and exact simulation of stationary Gaussian processes through circulant embedding of the covariance matrix*, SIAM J. Sci. Comput. 18 (1997), pp. 1088–1107.
[13] C.B. Fandry and L.M. Leslie, *A two-layer quasi-geostrophic model of summer through formation in the Australian subtropical easterlies*, J. Atmospheric Sci. 41 (1984), pp. 807–817.
[14] M. Fisher and S. Gürol, *Parallelisation in the time dimension of four-dimensional variational data assimilation*, (2016), submitted to Quarterly Journal of the Royal Meteorological Society.
[15] M. Fisher, J. Nocedal, Y. Trémolet, and S.J. Wright, *Data assimilation in weather forecasting: A case study in PDE-constrained optimization*, Optim. Eng. 10 (2009), pp. 409–426.
[16] M. Fisher, Y. Trémolet, H. Auvinen, D. Tan, and P. Poli, *Weak-constraint and long-window 4D-Var*, Tech. Rep. Technical Memorandum 655, ECMWF, 2011.
[17] R. Fletcher and M.J.D. Powell, *A rapidly convergent descent method for minimization*, Comput. J. 6 (1963), pp. 163–168.

[18] S. Gratton, A. Sartenaer, and J. Tshimanga, *On a class of limited memory preconditioners for large scale linear systems with multiple right-hand sides*, SIAM J. Optim. 21 (2011), pp. 912–935.
[19] A. Griewank and A. Walther, *On constrained optimization by adjoint based quasi-Newton methods*, Optim. Methods Softw. 17 (2002), pp. 869–889.
[20] E. Haber, *Quasi-Newton methods for large-scale electromagnetic inverse problems*, Inverse Problems. 21 (2005), pp. 305–324.
[21] E. Hólm, E. Andersson, A. Beljaars, P. Lopez, J.F. Mahfouf, A. Simmons, and J.N. Thépaut, *Assimilation and modelling of the hydrological cycle: ECMWF's status and plans*, Tech. Rep. 383, ECMWF, 2002.
[22] N.B. Ingleby, A.C. Lorenc, K. Ngan, F. Rawlins, and D.R. Jackson, *Improved variational analyses using a nonlinear humidity control variable*, Q. J. R. Meteorol. Soc. 139 (2013), pp. 1875–1887.
[23] D. Katz, A.S. Lawless, N.K. Nichols, M.J.P. Cullen, and R.N. Bannister, *Correlations of control variables in variational data assimilation*, Q. J. R. Meteorol. Soc. 137 (2011), pp. 620–630.
[24] C. Keller, N.I.M. Gould, and A.J. Wathen, *Constraint preconditioning for indefinite linear systems*, SIAM J. Matrix Anal. Appl. 21 (2000), pp. 1300–1317.
[25] Y. Kuznetsov, *Efficient iterative solvers for elliptic finite element problems on nonmatching grids*, Russian J. Numer. Anal. Math. Model. 10 (1995), pp. 187–212.
[26] J.R. Magnus and H. Neudecker, *Matrix Differential Calculus with Applications in Statistics and Econometrics*, John Wiley and Sons, Chichester, 2007.
[27] J.L. Morales and J. Nocedal, *Automatic preconditioning by limited memory quasi-Newton updating*, SIAM J. Optim. 10 (1999), pp. 1079–1096.
[28] M.F. Murphy, G.H. Golub, and A.J. Wathen, *A note on preconditioning for indefinite linear systems*, SIAM J. Sci. Comput. 21 (1999), pp. 1969–1972.
[29] J. Nocedal and S.J. Wright, *Numerical Optimization*, Series in Operations Research and Financial Engineering, Springer-Verlag, New-York, 2006.
[30] W. Nowak, S. Tenkleve, and O. Cirpka, *Efficient computation of linearized cross-covariance and auto-covariance matrices of interdependent quantities*, Math. Geol. 35 (2003), pp. 53–66.
[31] S.S. Oren and E. Spedicato, *Optimal conditioning of self-scaling variable metric algorithms*, Math. Program. 10 (1976), pp. 70–90.
[32] J. Pedlosky, *Geophysical Fluid Dynamics*, Springer, New-York, 1979.
[33] M.J.D. Powell, *A new algorithm for unconstrained optimization*, in *Nonlinear Programming*, J.B. Rosen, O.L. Mangasarian, and K. Ritter, eds., Academic Press, New-York, 1970, pp. 31–65.
[34] F. Rabier and M. Fisher, *Data assimilation in meteorology*, in *Ecole de Physique des Houches Special Issue 28 May–15 June: Advanced Assimilation for Geosciences*, E. Blayo, M. Bocquet, E. Cosme, and L.F. Cugliandolo, eds., Chap. 19, Oxford University Press, Oxford, 2015, pp. 433–457.
[35] Y. Saad and M.H. Schultz, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput. 7 (1986), pp. 856–869.
[36] R.B. Schnabel, *Quasi-Newton methods using multiple secant equations*, Tech. Rep. ADA131444, Colorada Univ., Dept. of Computer Science, Boulder, 1983.
[37] D.F. Shanno and K.H. Phua, *Matrix conditioning and nonlinear optimization*, Math. Program. 14 (1978), pp. 149–160.
[38] J. Sun, *Structured backward errors for KKT systems*, Linear Algebra Appl. 288 (1999), pp. 75–88.
[39] W. Sun and Y.X. Yuan, *Optimization Theory and Methods*, Nonlinear Programming Vol. 1, Springer, New York, 2006.
[40] C. Tavolato and L. Isaksen, *On the use of a Huber norm for observation quality control in the ECMWF 4D-Var*, Q. J. R. Meteorol. Soc. 141 (2015), pp. 1514–1527.
[41] Y. Trémolet, *Accounting for an imperfect model in 4D-Var*, Q. J. R. Meteorol. Soc. 132 (2006), pp. 2483–2504.
[42] Y. Trémolet, *Model-error estimation in 4D-Var*, Q. J. R. Meteorol. Soc. 133 (2007), pp. 1267–1280.
[43] H.A. van der Vorst, *Iterative Krylov Methods for Large Linear Systems*, Cambridge University Press, Cambridge, 2003.
[44] A. Weaver and P. Courtier, *Correlation modelling on the sphere using a generalized diffusion equation*, Q. J. R. Meteorol. Soc. 127 (2001), pp. 1815–1846.
[45] S. Yoshikazu, *Numerical variational analysis with weak constraint and application to surface analysis of severe storm gust*, Mon. Weather Rev. 98 (1970), pp. 899–910.

Appendix . Proof of Lemma 3.1

Proof We note that Equations (18) and (19) are consistent since both X and Y are supposed to be of full column rank [26, Chapter 2, Theorem 13]. We write the QR decompositions of the ℓ by k matrix X and of the $(\ell + m)$ by k matrix Y as

$$X = \mathcal{Q}^{(1)}\mathcal{R}^{(1)}, \tag{A.1}$$

$$Y = \mathcal{Q}^{(2)}\mathcal{R}^{(2)}. \tag{A.2}$$

We note that both $\mathcal{R}^{(1)}$ and $\mathcal{R}^{(2)}$ are invertible k by k matrices, since X and Y are supposed to be of full column rank. Equations (18) and (19) can then be recast as

$$\Delta\tilde{B}^{\mathrm{T}}\mathcal{Q}^{(1)}\mathcal{R}^{(1)} = P,$$
$$\Delta\tilde{B}\mathcal{Q}^{(2)}\mathcal{R}^{(2)} = Q,$$

or equivalently as

$$\Delta\tilde{B}^{\mathrm{T}}\mathcal{Q}^{(1)} = P(\mathcal{R}^{(1)})^{-1}, \tag{A.3}$$
$$\Delta\tilde{B}\mathcal{Q}^{(2)} = Q(\mathcal{R}^{(2)})^{-1}. \tag{A.4}$$

We refer to the ℓ by $\ell - k$ matrix $\mathcal{Q}_{\perp}^{(1)}$ as the orthogonal complement of $\mathcal{Q}^{(1)}$ in $\mathbb{R}^{\ell}$, whereas the $(\ell + m)$ by $(\ell + m - k)$ matrix $\mathcal{Q}_{\perp}^{(2)}$ denotes the orthogonal complement of $\mathcal{Q}^{(2)}$ in $\mathbb{R}^{(\ell+m)}$, respectively. $\Delta\tilde{B}$ can then be written as

$$\Delta\tilde{B} = [\mathcal{Q}^{(1)}\ \mathcal{Q}_{\perp}^{(1)}]\Delta\tilde{B}'\begin{bmatrix}(\mathcal{Q}^{(2)})^{\mathrm{T}}\\(\mathcal{Q}_{\perp}^{(2)})^{\mathrm{T}}\end{bmatrix}, \tag{A.5}$$

where $\Delta\tilde{B}'$ is a ℓ by $(\ell + m)$ matrix of the form

$$\Delta\tilde{B}' = \begin{bmatrix}\Lambda_{11} & \Lambda_{12}\\ \Lambda_{21} & \Lambda_{22}\end{bmatrix}.$$

Postmultiplying Equation (A.5) by $\mathcal{Q}^{(2)}$ leads to

$$\Delta\tilde{B}\mathcal{Q}^{(2)} = [\mathcal{Q}^{(1)}\ \mathcal{Q}_{\perp}^{(1)}]\quad \Delta\tilde{B}'\begin{bmatrix}I_k\\0\end{bmatrix} = \mathcal{Q}^{(1)}\Lambda_{11} + \mathcal{Q}_{\perp}^{(1)}\Lambda_{21}.$$

Using Equation (A.4), we obtain

$$\mathcal{Q}^{(1)}\Lambda_{11} + \mathcal{Q}_{\perp}^{(1)}\Lambda_{21} = Q(\mathcal{R}^{(2)})^{-1}, \tag{A.6}$$

which leads to

$$\Lambda_{11} = (\mathcal{Q}^{(1)})^{\mathrm{T}}Q(\mathcal{R}^{(2)})^{-1}. \tag{A.7}$$

Similarly, by premultiplying Equation (A.5) by $(\mathcal{Q}^{(1)})^{\mathrm{T}}$, we obtain

$$(\mathcal{Q}^{(1)})^{\mathrm{T}}\Delta\tilde{B} = [I_k\ 0]\Delta\tilde{B}'\quad \begin{bmatrix}(\mathcal{Q}^{(2)})^{\mathrm{T}}\\(\mathcal{Q}_{\perp}^{(2)})^{\mathrm{T}}\end{bmatrix} = \Lambda_{11}(\mathcal{Q}^{(2)})^{\mathrm{T}} + \Lambda_{12}(\mathcal{Q}_{\perp}^{(2)})^{\mathrm{T}}.$$

Using Equation (A.3) then leads to

$$\Lambda_{11}(\mathcal{Q}^{(2)})^{\mathrm{T}} + \Lambda_{12}(\mathcal{Q}_{\perp}^{(2)})^{\mathrm{T}} = (\mathcal{R}^{(1)})^{-T}P^{\mathrm{T}}. \tag{A.8}$$

Finally, postmultiplying this last relation by $\mathcal{Q}^{(2)}$ gives

$$\Lambda_{11} = (\mathcal{R}^{(1)})^{-T}P^{\mathrm{T}}\mathcal{Q}^{(2)}. \tag{A.9}$$

From Equations (A.7) and (A.9), we conclude that

$$\begin{aligned}(\mathcal{Q}^{(1)})^{\mathrm{T}}Q(\mathcal{R}^{(2)})^{-1} &= (\mathcal{R}^{(1)})^{-T}P^{\mathrm{T}}\mathcal{Q}^{(2)},\\ (\mathcal{Q}^{(1)})^{\mathrm{T}}Q &= (\mathcal{R}^{(1)})^{-T}P^{\mathrm{T}}\mathcal{Q}^{(2)}\mathcal{R}^{(2)},\\ (\mathcal{R}^{(1)})^{\mathrm{T}}(\mathcal{Q}^{(1)})^{\mathrm{T}}Q &= P^{\mathrm{T}}Y,\\ X^{\mathrm{T}}Q &= P^{\mathrm{T}}Y,\end{aligned}$$

which completes the proof. Finally, we note that the proof is based on the equivalence of Equations (A.8) and (A.6) to Equations (A.3) and (A.4), respectively. Equations (A.8) and (A.6) helped us to determine specific blocks of rows or columns of $\Delta\tilde{B}$ in the given decomposition basis. More precisely, we have determined a possible solution of the direct and adjoint secant equations as

$$\Delta\tilde{B} = \mathcal{Q}^{(1)}\Lambda_{11}(\mathcal{Q}^{(2)})^{\mathrm{T}} + \mathcal{Q}^{(1)}\Lambda_{12}(\mathcal{Q}_{\perp}^{(2)})^{\mathrm{T}} + \mathcal{Q}_{\perp}^{(1)}\Lambda_{21}(\mathcal{Q}^{(2)})^{\mathrm{T}} + \mathcal{Q}_{\perp}^{(1)}\Lambda_{22}(\mathcal{Q}_{\perp}^{(2)})^{\mathrm{T}},$$

with Λ_{11} given in Equation (A.9),

$$\Lambda_{12} = (\mathcal{R}^{(1)})^{-T}P^{\mathrm{T}}\mathcal{Q}_{\perp}^{(2)},$$
$$\Lambda_{21} = (\mathcal{Q}_{\perp}^{(1)})^{\mathrm{T}}Q(\mathcal{R}^{(2)})^{-1},$$

and Λ_{22} a $(\ell - k)$ by $(\ell + m - k)$ matrix that can be freely chosen. ■