Preprint ANL/MCS-P1919-0711

A LOW-MEMORY APPROACH FOR BEST-STATE ESTIMATION OF HIDDEN MARKOV MODELS WITH MODEL ERROR

MIHAI ANITESCU*, XIAOYAN ZENG†, AND EMIL M. CONSTANTINESCU‡

Abstract. We present a low-memory approach for the best-state estimate (data assimilation) of hidden Markov models where model error is considered. In particular, our findings apply for the 4D-Var framework. The novelty of our approach resides in the fact that the storage needed by our estimation framework, while including model error, is dramatically reduced from $\mathcal{O}$(number of time steps) to $\mathcal{O}(1)$. The main insight is that we can restate the objective function of the state estimation (the likelihood function) from a function of all states to a function of the initial state only. We do so by restricting the other states by recursively enforcing the optimality conditions. This results in a regular nonlinear equation or an optimization problem for which a descent direction can be computed using only a forward sweep. In turn, the best estimate can be obtained by limited-memory quasi-Newton algorithms that need only $\mathcal{O}(1)$ storage with respect to the time steps. Our findings are demonstrated by numerical experiments on Burgers' equations.

Key words. Data Assimilation, Weakly Constrained 4DVar, Hidden Markov Models, Limited Memory Methods, Quasi-Newton Methods

AMS subject classifications. 90C53, 93E10, 62M05

1. Introduction.

Data assimilation is the process of computing the best estimate the trajectory of a dynamical system with observational data [5, 8, 9]. This technique is used extensively in meteorology and hydrology in order to make accurate predictions about the state of atmosphere and oceans [8, 13]. However, recent applications have called for explicit inclusion of model error such as from sub-grid modeling, boundary conditions and forcings. All these modeling uncertainties are aggregated into a component that is generically called *model error* [7, 15, 17], which in turn results in the following best-fit 4DVar-with-model-error functional [11, 12, 23, 24, 26].

$$\begin{aligned}\mathcal{J}(x_{t_0}, x_{t_1}, \ldots, x_{t_N}) =& \frac{1}{2}(x_{t_0} - x_B)^T Q_B^{-1}(x_{t_0} - x_B) + \\ & \frac{1}{2}\sum_{k=0}^{N}(\mathcal{H}_k(x_{t_k}) - y_k)^T R_k^{-1}(\mathcal{H}_k(x_{t_k}) - y_k) + \\ & \frac{1}{2}\sum_{k=0}^{N-1}(x_{t_{k+1}} - M_k(x_{t_k}))^T Q_k^{-1}(x_{t_{k+1}} - M_k(x_{t_k})).\end{aligned} \tag{1.1}$$

All the quantities of interest are indexed by k, $k = 0, 1, \ldots, N$, where t_k is the time instant. Here, the variables x_{t_k}, are the states of the model at times t_i, that need to be identified by minimizing the functional $\mathcal{J}$. The data of the problem are as follows. The quantities x_B and Q_B are the background state and the background covariance matrix, respectively. The vectors y_k represent the observations, whereas the nonlinear mapping $\mathcal{H}(\cdot)$ is the observation operator that maps states into observables. The

*Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, USA.(`anitescu@mcs.anl.gov`)

†Corresponding author. Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, USA(`zeng@mcs.anl.edu`).

‡Mathematics and CSomputer Science Division, Argonne National Laboratory, Argonne, IL, USA (`emconsta@mcs.anl.gov`).

matrix R_k is the covariance error for the observations. The mapping $M_k(\cdot)$ describes the evolution of the physical model, whereas the matrix Q_k quantifies the covariance of the model error. The functional J is the minus log likelihood of the hidden Markov model [20, 21]:

$$x_{t_{k+1}} = M(x_{t_k}) + \eta_k,\ y_k = \mathcal{H}(x_{t_k}) + \varepsilon_k,\ \eta_k \sim \mathcal{N}(\mathbf{0}, Q_k)\ \varepsilon_k \sim \mathcal{N}(\mathbf{0}, R_k).$$

For this reason, we call the minimization of J, which is equivalent to the maximum likelihood calculation for the hidden Markov model, *state estimation of hidden Markov models with model error.*

In the limiting case of 0 model error, that is, $Q_k \to \infty$, we obtain the so-called strongly constrained model [4, 8, 19], which is the one most commonly used in today's applications. Because it now includes the recursive constraints $x_{t_{k+1}} = M_k(x_{t_k})$, it can effectively be thought of as being a function only of the initial condition x_{t_0}, which is the only variable that needs to be stored, all the others being obtained by the recursion.

Unfortunately, this reduction does not apply to the case including model error, also called weakly constrained, which is now a function of $N+1$ times more variables and thus requires substantially more memory to store the result of the minimization of (1.1). As we move to ever higher spatial resolution such as global cloud resolving models that require a horizontal resolution of 1–3 Km^2, the amount of memory and storage space in the case of considering model error would make such computations out of practical reach. We focus on memory requirements because we are entering a phase in computational science where power considerations lead us to reduced available memory per unit of computational power ([6]).

In this study we introduce a numerical method that reduces the memory requirements of running the weakly constrained 4D-Var. The method is based on a shooting philosophy constrained by the optimality conditions for the likelihood function. Burgers' equation is used to illustrate the technique and compare it with a derivative-free or full memory-intensive implementation. While this will be done in a 1+1-DVar (in the sense that the spatial dimension is only 1), our example has the same time-dependence structure as full 4D-Var approaches. Therefore, we expect that conclusions about the dependence of the storage requirements of method on the number of time steps, the main investigation topic here, will carry through to the actual 4D-Var case.

The rest of the paper is structured as follows. In §2 we present our algorithm in an abstract framework, and we analyze its well-posedness. In §3 we discuss the implications of using the algorithms for the 4D-Var setup, and we present stability and regularity conditions and considerations. Numerical experiments to validate our findings are presented in §4. In §5 we summarize our conclusions.

2. A low-memory approach for data assimilation with model error.

We introduce an abstraction of the data assimilation with the model error problem, the 4D-Var problem (1.1). The abstraction will be useful in understanding the fundamentals of our approach, reducing the notation burden, and providing a framework for the extension of these results.

2.1. Abstraction of the problem.

We assume that we are trying to recover the states $\boldsymbol{x}_i$, $i = 0, 1, \ldots, N$, of a system that evolves over N time steps with $\boldsymbol{x}_0$ as an initial state and $\boldsymbol{x}_N$ as a final state.

We assume that this optimal state is recovered by minimizing a cost functional with several components. These components are of two types:

- *Evolution components*, which constrain the relative evolution of two consecutive components, $\phi_i(\boldsymbol{x}_i, \boldsymbol{x}_{i+1})$, $i = 0, 1, \ldots, N-1$.
- *Observational components*, which constrain each state either by means of observations or by means of a background prior, γ_i, $i = 0, 1, \ldots, N$.

We define the scaled cost functional Γ as

$$\Gamma(\boldsymbol{x}_{0:N}) := \frac{1}{N}\Big(\sum_{i=0}^{N-1}[\gamma_i(\boldsymbol{x}_i) + \phi_i(\boldsymbol{x}_i, \boldsymbol{x}_{i+1})] + \gamma_N(\boldsymbol{x}_N)\Big). \tag{2.1}$$

Minimizing this functional Γ will result in the best estimate according to the Γ criterion. The rescaling will not affect the solution of the problem, but it is useful in comparing residuals for increasing N. We will ignore the rescaling in the theoretical derivations, but we will use it when comparing the numerical results.

2.2. Illustration of the abstraction in the case of 4D-Var. In the case of the 4D-Var approach (1.1), we have that, for $i = 0, \cdots, N-1$,

$$\phi_i(\boldsymbol{x}_i, \boldsymbol{x}_{i+1}) = \frac{1}{2}\big(\boldsymbol{x}_{i+1} - \mathcal{M}_i(\boldsymbol{x}_i)\big)^T Q_i^{-1}\big(\boldsymbol{x}_{i+1} - \mathcal{M}_i(\boldsymbol{x}_i)\big) \tag{2.2}$$

corresponds to the model error. Also, for $i = 1, \cdots, N$,

$$\gamma_i(\boldsymbol{x}_i) = \frac{1}{2}\left(\boldsymbol{y}_i - \mathcal{H}_i(\boldsymbol{x}_i)\right)^T R_i^{-1}\left(\boldsymbol{y}_i - \mathcal{H}_i(\boldsymbol{x}_i)\right) \tag{2.3}$$

corresponds to the difference between observations and its model counterparts. For $i = 0$, γ_0 includes the background error measurement for the current value of $\boldsymbol{x}_0$ and is formulated as

$$\gamma_0(\boldsymbol{x}_0) = \frac{1}{2}(\boldsymbol{x}_0 - \boldsymbol{x}_B)^T Q_B^{-1}(\boldsymbol{x}_0 - \boldsymbol{x}_B) + \frac{1}{2}\left(\boldsymbol{y}_0 - \mathcal{H}_0(\boldsymbol{x}_0)\right)^T R_0^{-1}\left(\boldsymbol{y}_0 - \mathcal{H}_0(\boldsymbol{x}_0)\right). \tag{2.4}$$

Here $\boldsymbol{x}_i = x_{t_i}$ denotes a state in the ith step. We also define by $\boldsymbol{x}_{0:N} := [\boldsymbol{x}_0, \cdots, \boldsymbol{x}_N]^T$ for shorthand.

2.3. Reduced-memory algorithm. In this section, our goal is to define an algorithm to minimize functional Γ as in (2.1), while storing at any time only a small number of $\{\boldsymbol{x}_i\}$.

Assumption 1. *Assume that $\phi_i(\boldsymbol{x}_i, \boldsymbol{x}_{i+1})$ and $\gamma_i(\boldsymbol{x}_i)$ are continuously differentiable and that the mixed differentiation function $\nabla^2_{\boldsymbol{x}_{i+1}\boldsymbol{x}_i}\phi_i(\boldsymbol{x}_i, \boldsymbol{x}_{i+1})$ is invertible in the neighborhood of the minimum $x^*_{0:N}$.*

Later we will verify that this assumption is indeed valid for the case of the 4Dvar approach with model error (1.1).

We now define a sequence of functions as follows:

$$\theta_0(\boldsymbol{x}_0, \boldsymbol{x}_1) := \nabla_{\boldsymbol{x}_0}\phi_0 + \nabla_{\boldsymbol{x}_0}\gamma_0; \tag{2.5a}$$

$$\theta_i(\boldsymbol{x}_{i-1}, \boldsymbol{x}_i, \boldsymbol{x}_{i+1}) := \nabla_{\boldsymbol{x}_i}\phi_i + \nabla_{\boldsymbol{x}_i}\phi_{i-1} + \nabla_{\boldsymbol{x}_i}\gamma_i, \quad i = 1, \ldots, N-1; \tag{2.5b}$$

$$\theta_N(\boldsymbol{x}_{N-1}, \boldsymbol{x}_N) := \nabla_{\boldsymbol{x}_N}\phi_{N-1} + \nabla_{\boldsymbol{x}_N}\gamma_N. \tag{2.5c}$$

It immediately follows from (2.1) that the following relationships hold for the partial derivatives of Γ.

$$\nabla_{\boldsymbol{x}_0}\Gamma(x_{0:N}) = \theta_0(\boldsymbol{x}_0, \boldsymbol{x}_1) \tag{2.6a}$$

$$\nabla_{\boldsymbol{x}_i}\Gamma(x_{0:N}) = \theta_i(\boldsymbol{x}_{i-1}, \boldsymbol{x}_i, \boldsymbol{x}_{i+1}), \quad i = 1, \ldots, N-1. \tag{2.6b}$$

$$\nabla_{\boldsymbol{x}_N}\Gamma(x_{0:N}) = \theta_N(\boldsymbol{x}_{N-1}, \boldsymbol{x}_N) \tag{2.6c}$$

The core of our method is based on the following observation.

THEOREM 1. *Under Assumption 1, there exist continuously differentiable mappings $\lambda_i(\boldsymbol{x}_0)$, $i = 1, 2, \ldots, N$, such that*

$$\theta_0(\boldsymbol{x}_0, \lambda_1(\boldsymbol{x}_0)) = 0 \tag{2.7}$$

$$\theta_i(\lambda_{i-1}(\boldsymbol{x}_0), \lambda_i(\boldsymbol{x}_0), \lambda_{i+1}(\boldsymbol{x}_0)) = 0, \quad i = 1, 2, \ldots, N-1. \tag{2.8}$$

Moreover, for any $\boldsymbol{x}_0$, $\{\lambda_i(\boldsymbol{x}_0)\}_{i=1,2\ldots,N}$ are the unique vectors with this property.

Proof. We have from the definition of θ_i (while temporarily dropping the obvious dependence on $\boldsymbol{x}_0$) that

$$\nabla_{\boldsymbol{x}_{i+1}}\theta_i(\boldsymbol{x}_{i-1}, \boldsymbol{x}_i, \boldsymbol{x}_{i+1}) = \nabla_{\boldsymbol{x}_{i+1}}\nabla_{\boldsymbol{x}_i}\phi_i(\lambda_i, \lambda_{i+1}). \tag{2.9}$$

From Assumption 1 we have that $\nabla^2_{\boldsymbol{x}_{i+1}\boldsymbol{x}_i}\phi_i(\lambda_i, \lambda_{i+1})$ is invertible, which in turn makes the Jacobian of the associated nonlinear equation in (2.9) invertible in $\boldsymbol{x}_{i+1}$, $i = 1, 2, \ldots, N-1$ (with a similar conclusion for $i = 0$). The conclusion follows from application of the implicit function theorem recursively in (2.9). $\square$

Based on Theorem 1, we can rewrite Γ as a function of $\boldsymbol{x}_0$ as follows:

$$\widehat{\Gamma}(\boldsymbol{x}_0) = \frac{1}{N}\Big[\sum_{i=0}^{N-1}\gamma_i\big(\lambda_i(\boldsymbol{x}_0)\big) + \phi_i\big(\lambda_i(\boldsymbol{x}_0), \lambda_{i+1}(\boldsymbol{x}_0)\big) + \gamma_N\big(\lambda_N(\boldsymbol{x}_0)\big)\Big]. \tag{2.10}$$

By transferring the cost function (2.1) into (2.10), a function of initial state, considerable storage space is saved during computation since we reduce the multistate function to a single-state function. The main vehicle for this reduction is the explicit enforcement of the optimality conditions at each of the time steps other than the initial one. In some sense, the optimality conditions become the strong constraint in the approach, replacing the perfect model assumption from current 4DVar data assimilation procedures, that is, of course, if we can manipulate the function $\widehat{\Gamma}$ as required by the optimization algorithms in a way that does maintain an $\mathcal{O}(1)$ storage.

To that end, we need more theoretical support to verify that the optimum solution of (2.10) is the same as the initial state of the original problem's optimum solution. It is well known that for a twice continuously differentiable function f, if x is a local minimizer of f, then there are two necessary conditions must be satisfied: *$f'(x)$ equals 0 (first-order necessary condition; x here is called a stationary point) and $f''(x)$ is positive semi-definite (second-order necessary condition).* The sufficient conditions that x is a local minimizer of f are that *x is a stationary point and $f''(x)$ is positive definite (second-order sufficient condition).* Hence we need to figure out the derivatives first.

The gradient of $\widehat{\Gamma}$ is calculated as

$$\nabla_{\boldsymbol{x}_0}\widehat{\Gamma} = \theta_0(\lambda_0, \lambda_1) + (\nabla_{\boldsymbol{x}_0}\lambda_N)^T\theta_N(\lambda_{N-1}, \lambda_N) + \sum_{i=1}^{N-1}(\nabla_{\boldsymbol{x}_0}\lambda_i)^T\theta_i(\lambda_{i-1}, \lambda_i, \lambda_{i+1}).$$

Because of the way $\lambda_i, i = 1, \cdots, N$ are computed from the recursion (2.9), which implies that $\theta_0(\lambda_0, \lambda_1) \equiv 0$ and $\theta_i(\lambda_{i-1}, \lambda_i, \lambda_{i+1}) \equiv 0, \ \ i = 1, \cdots, N-1$, we have that

$$\nabla_{\boldsymbol{x}_0}\widehat{\Gamma} = (\nabla_{\boldsymbol{x}_0}\lambda_N)^T\theta_N(\lambda_{N-1}, \lambda_N) \quad \boldsymbol{x}_0 \in \mathcal{N}(\boldsymbol{x}_0^*). \tag{2.11}$$

Define $L_i := \nabla_{\boldsymbol{x}_0} \lambda_i$. The second-order derivative of $\widehat{\Gamma}$ at $\boldsymbol{x}_0^*$ is calculated by product rule as

$$\begin{aligned}\nabla^2_{\boldsymbol{x}_0\boldsymbol{x}_0}\widehat{\Gamma} = &\nabla_{\lambda_0}\theta_0 + L_1^T \nabla_{\lambda_1}\theta_0 + \left(L_{N-1}^T \nabla_{\lambda_{N-1}}\theta_N + L_N^T \nabla_{\lambda_N}\theta_N\right) L_N + \\ &\sum_{i=1}^{N-1} \left(L_{i-1}^T \nabla_{\lambda_{i-1}}\theta_i + L_i^T \nabla_{\lambda_i}\theta_i + L_{i+1}^T \nabla_{\lambda_{i+1}}\theta_i\right) L_i + \\ &\sum_{i=1}^{N-1} \left((\theta_i)^T \otimes I_s\right) \nabla_{\boldsymbol{x}_0} \mathrm{vec}(L_i) + \left((\theta_N)^T \otimes I_s\right) \nabla_{\boldsymbol{x}_0} \mathrm{vec}(L_N),\end{aligned} \tag{2.12}$$

where I_s is an $s \times s$ identity matrix with s being the dimension of $\boldsymbol{x}_i$ and $\otimes$ denotes Kronecker product. To prove (2.12), we need only to prove that the first derivative matrix of $s \times s$ matrix M and $s \times 1$ vector $\boldsymbol{u}$, with respect to $s \times 1$ vector $\boldsymbol{x}$, i.e

$$\nabla_{\boldsymbol{x}}(M\boldsymbol{u}) = \left(\boldsymbol{u}^T \otimes I_s\right) \nabla_{\boldsymbol{x}} \mathrm{vec}(M) + M \nabla_{\boldsymbol{x}} \boldsymbol{u}.$$

Here the Kronecker product, $\boldsymbol{u}^T \otimes I_s = \begin{pmatrix} u_1 I_s & \cdots & u_s I_s \end{pmatrix}$, and $\mathrm{vec}(M)$ is a $s^2 \times 1$ vector stacking the columns of the matrix M on top of one another; that is, $\mathrm{vec}(M) = \begin{pmatrix} m_{11} & \cdots & m_{s,1} & \cdots & m_{1s} & \cdots & m_{s,s} \end{pmatrix}^T$. The first derivative matrix of $\mathrm{vec}(M)$ is

$$\nabla_{\boldsymbol{x}} \mathrm{vec}(M) = \begin{pmatrix} \frac{\partial m_{11}}{\partial x_1} & \cdots & \frac{\partial m_{11}}{\partial x_s} \\ \vdots & \vdots & \vdots \\ \frac{\partial m_{ss}}{\partial x_1} & \cdots & \frac{\partial m_{ss}}{\partial x_s} \end{pmatrix}.$$

Hence the ith-row-and-jth-column element of $\left(\boldsymbol{u}^T \otimes I_s\right) \nabla_{\boldsymbol{x}} \mathrm{vec}(M)$ is $\sum_{k=1}^s \frac{\partial m_{ik}}{\partial x_j} u_k$. The ith-row-and-jth-column element of $M \nabla_{\boldsymbol{x}} \boldsymbol{u}$ is $\sum_{k=1}^s m_{i,k} \frac{\partial u_k}{\partial x_j}$. The ith-row element of $M\boldsymbol{u} = \sum_{k=1}^s m_{ik} u_k$ and hence the ith-row-and-jth-column element of $\nabla_{\boldsymbol{x}}(M\boldsymbol{u})$ is $\sum_{k=1}^s m_{ik} \frac{\partial u_k}{\partial x_j} + \sum_{k=1}^s \frac{\partial m_{ik}}{\partial x_j} u_k$. Hence (2.12) is verified.

From Theorem 1, the last line of Equation (2.12) is zero. Then (2.12) can be simplified at $\boldsymbol{x}_0^*$ as

$$\begin{aligned}\nabla^2_{\boldsymbol{x}_0\boldsymbol{x}_0}\widehat{\Gamma} = &\nabla_{\lambda_0}\theta_0 + L_1^T \nabla_{\lambda_1}\theta_0 + \left(L_{N-1}^T \nabla_{\lambda_{N-1}}\theta_N + L_N^T \nabla_{\lambda_N}\theta_N\right) L_N + \\ &\sum_{i=1}^{N-1} \left(L_{i-1}^T \nabla_{\lambda_{i-1}}\theta_i + L_i^T \nabla_{\lambda_i}\theta_i + L_{i+1}^T \nabla_{\lambda_{i+1}}\theta_i\right) L_i^T.\end{aligned}$$

Because $\nabla_{\lambda_j}\theta_i = \nabla_{\boldsymbol{x}_j} \nabla_{\boldsymbol{x}_i} \Gamma|_{\boldsymbol{x}_j = \lambda_j}, j = i-1, i, i+1$, one can easily verify that

$$\nabla^2_{\boldsymbol{x}_0\boldsymbol{x}_0}\widehat{\Gamma} = \Lambda^T \left(\nabla^2_{\boldsymbol{x}_{0:N}} \Gamma(\boldsymbol{x}_0, \lambda_{1:N})\right) \Lambda, \tag{2.13}$$

where

$$\Lambda^T = \left[I, (\nabla_{\boldsymbol{x}_0}\lambda_1)^T, \cdots, (\nabla_{\boldsymbol{x}_0}\lambda_N)^T\right]. \tag{2.14}$$

From (2.6) and (2.11) as well as the definition of the mappings θ_i, it immediately follows that the component $\boldsymbol{x}_0^*$ of a stationary point $\boldsymbol{x}_0^*, \boldsymbol{x}_1^*, \ldots, \boldsymbol{x}_N^*$ of (2.1) also satisfies $\nabla_{\boldsymbol{x}_0}\widehat{\Gamma}(\boldsymbol{x}_0^*) = 0$. Therefore, it is a stationary point of $\widehat{\Gamma}$. In the following result, we show that the reciprocal is also true under some mild assumptions.

THEOREM 2. *If $\boldsymbol{x}_0^*$ is a local minimizer of $\widehat{\Gamma}(\boldsymbol{x}_0)$ and $\lambda_N(\boldsymbol{x}_0)$ is invertible, then $\left(\boldsymbol{x}_0^*, \lambda_1(\boldsymbol{x}_0^*), \cdots, \lambda_N(\boldsymbol{x}_0^*)\right)$ is a stationary point of $\Gamma(\boldsymbol{x}_{0:N})$.*

Proof. From the definition of the mapping $\lambda_i(\cdot)$ in Theorem 1, we have that for $i = 1, \cdots, N-1$,

$$\theta_0(\boldsymbol{x}_0^*, \lambda_1(\boldsymbol{x}_0^*)) = 0, \theta_i(\lambda_{i-1}(\boldsymbol{x}_0^*), \lambda_i(\boldsymbol{x}_0^*), \lambda_{i+1}(\boldsymbol{x}_0^*)) = 0.$$

Furthermore, according to the condition that $\boldsymbol{x}_0^*$ is a local minimizer of $\widehat{\Gamma}(\boldsymbol{x}_0)$, it follows that the derivative of $\widehat{\Gamma}$ with respect to $\boldsymbol{x}_0^*$ is zero. That is, according to (2.11),

$$(\nabla_{\boldsymbol{x}_0}\lambda_N)^T \theta_N(\lambda_{N-1}, \lambda_N) = 0.$$

Because $\nabla_{\boldsymbol{x}_0}\lambda_N$ is invertible, it follows that $\theta_N(\lambda_{N-1}, \lambda_N) = 0$. Let $\boldsymbol{x}_i^* = \lambda_i(\boldsymbol{x}_0^*)$. It is then immediate that $\boldsymbol{x}_{0:N}^*$ satisfies (2.6) and is thus a stationary point of $\Gamma(\boldsymbol{x}_{0:N})$. The proof is complete. □

According to (2.13), the Hessian of $\widehat{\Gamma}$ at its local minimizer is only a lower-dimension projection of the Hessian of Γ at a corresponding point. Hence the local minimum of $\widehat{\Gamma}$ is not necessary to be local minimum of Γ. Let us take a simple one-dimensional problem for a counterexample. Let $\Gamma(x_0, x_1) = x_0 x_1 - \frac{1}{2}x_0^3 + \frac{7}{2}x_0^2 - 6x_0 - 3x_1$. Here $N = 1$, $\phi_0(x_0, x_1) = x_0 x_1$, $\gamma_0(x_0) = -\frac{1}{2}x_0^3 + \frac{7}{2}x_0^2 - 6x_0$, and $\gamma_1(x_1) = -3x_1$. It easy to show that $x_1 = \frac{3}{2}x_0^2 - 7x_0 + 6$ solves $\frac{\partial \Gamma(x_0, x_1)}{\partial x_0} = 0$. By replacing x_1 in Γ by $\frac{3}{2}x^2 + 0 - 7x_0 + 6$, we can get $\widehat{\Gamma}(x_0) = x_0^3 - 8x_0^2 + 21x_0 - 18$. Obviously, $\frac{\partial \widehat{\Gamma}}{\partial x_0}|_{x_0=3} = 0$ and $\frac{\partial^2 \widehat{\Gamma}}{\partial x_0^2}|_{x_0=3} = 2 > 0$; therefore, $x_0 = 3$ is local minimizer of $\widehat{\Gamma}$. However, when $x_0 = 3$, the Hessian of Γ satisfies

$$\nabla^2_{x_0, x_1}\Gamma(x_0, x_1) = \begin{bmatrix} -3x_0 + 7 & 1 \\ 1 & 0 \end{bmatrix} \tag{2.15}$$

and is indefinite with eigenvalues -2.4142 and 0.4142.

We can prove that the initial state of local minimizer of (2.1) is also the local minimizer of (2.10). Moreover, and perhaps more important we can now prove that the minimization of (2.10) is equivalent to a nonlinear equation with nonsingular Jacobian, whose residual can be computed by doing forward sweeps only.

THEOREM 3. *Let $\boldsymbol{x}_0^*$ be the first component of a local minimizer of $\Gamma(\boldsymbol{x}_{0:N})$ that satisfies the second-order sufficient condition. Then the following hold:*

- *[i] $\boldsymbol{x}_0^*$ is a local minimizer of $\widehat{\Gamma}(\boldsymbol{x}_0)$ that satisfies the second-order sufficient conditions in x_0.*
- *[ii] The matrix $\nabla_{\boldsymbol{x}_0}\lambda_N(x_0)$ is invertible at $\boldsymbol{x}_0^*$, where $\lambda_N(\boldsymbol{x}_0)$ is one of the mappings from Theorem 1.*
- *[iii] In a neighborhood of $\boldsymbol{x}_0^*$, we have that*
  - *[iii-a] $\nabla_{\boldsymbol{x}_0}\theta_N(\lambda_{N-1}(\boldsymbol{x}_0), \lambda_N(\boldsymbol{x}_0))$ is invertible in a neighborhood of $\boldsymbol{x}_0^*$.*
  - *[iii-b] $\theta_N(\lambda_{N-1}(\boldsymbol{x}_0), \lambda_N(\boldsymbol{x}_0)) = 0 \Rightarrow \boldsymbol{x}_0 = \boldsymbol{x}_0^*$*
  - *[iii-c] There exists C_θ such that*

$$||\theta_N(\lambda_{N-1}(\boldsymbol{x}_0)\lambda_N(\boldsymbol{x}_0))|| \geq C_\theta \left|\left|\nabla_{x_0}\widehat{\Gamma}(x_0)\right|\right|$$

Proof. If $\boldsymbol{x}_{0:N}^*$ is a local minimizer of $\Gamma(\boldsymbol{x}_{0:N})$, then $\boldsymbol{x}_{0:N}^*$ satisfies (2.9), and $\theta_N(\boldsymbol{x}_{N-1}, \boldsymbol{x}_N) = 0$. Then, $\lambda_i(\boldsymbol{x}_0^*) = \boldsymbol{x}_i^*$ and $\nabla_{\boldsymbol{x}_0}\widehat{\Gamma}(\boldsymbol{x}_0^*) = 0$.

Furthermore the second-order sufficient condition is satisfied by $\boldsymbol{x}_{0:N}^*$ for Γ; in other words, $\nabla^2_{\boldsymbol{x}_{0:N}\boldsymbol{x}_{0:N}}\Gamma(\boldsymbol{x}_{0:N}^*)$ is positive definite. Then, $\nabla^2_{\boldsymbol{x}_0\boldsymbol{x}_0}\widehat{\Gamma}(\boldsymbol{x}_0^*)$ is also positive definite from (2.13) and the fact that the matrix Λ in that equation is full rank because of the inclusion of an identity block.

To sum up, $\boldsymbol{x}_0^*$ is a stationary point of $\widehat{\Gamma}(\boldsymbol{x}_0^*)$ that satisfies the second-order sufficient condition. Then, it is also a local minimizer of $\widehat{\Gamma}$, and part [i] of the Theorem is proved.

For part [ii], we use (2.11) to obtain that (where we drop the dependence of λ_i on x_0 to simplify notation)

$$\begin{aligned}\nabla^2_{\boldsymbol{x}_0\boldsymbol{x}_0}\widehat{\Gamma}(\boldsymbol{x}_0^*) = (\nabla_{\boldsymbol{x}_0}\lambda_N)^T\nabla_{\lambda_{N-1}}\theta_N(\lambda_{N-1},\lambda_N)\nabla_{\boldsymbol{x}_0}\lambda_{N-1}+ \\ (\nabla_{\boldsymbol{x}_0}\lambda_N)^T\nabla_{\lambda_N}\theta_N(\lambda_{N-1},\lambda_N)\nabla_{\boldsymbol{x}_0}\lambda_N. \end{aligned} \tag{2.16}$$

Note that the component of the Hessian involving the second-derivative λ_N disappears since $\theta_N(\lambda_{N-1},\lambda_N)=0$ at $\boldsymbol{x}_0^*$. Also note that the above formula for $\nabla^2_{\boldsymbol{x}_0\boldsymbol{x}_0}\widehat{\Gamma}(\boldsymbol{x}_0^*)$ does not imply that it is nonsymmetric (which would be a contradiction). The symmetry of the matrix would eventually unfold after using the recursion for λ_i and, implicitly, their Jacobians. Nevertheless, the form presented is sufficient for us to reach our conclusions.

Assume now that $\nabla_{\boldsymbol{x}_0}\lambda_N$ were not invertible. Then, there must be a vector $u \neq 0$ such that $\nabla_{\boldsymbol{x}_0}\lambda_N u = 0$. Using (2.16), we obtain that $u^T\nabla^2_{\boldsymbol{x}_0\boldsymbol{x}_0}\widehat{\Gamma}(\boldsymbol{x}_0^*)u = 0$, which contradicts the conclusion reached at part [i]. This proves the part [ii] of the theorem.

For part [iii], we use (2.11) to obtain

$$(\nabla_{\boldsymbol{x}_0}\lambda_N)^{-T}\nabla_{\boldsymbol{x}_0}\widehat{\Gamma}(\boldsymbol{x}_0) = \theta_N(\lambda_{N-1},\lambda_N), \tag{2.17}$$

which in turn, with $\nabla_{\boldsymbol{x}_0}\widehat{\Gamma}(\boldsymbol{x}_0^*)=0$, results in

$$\nabla_{\boldsymbol{x}_0}\theta_N(\lambda_{N-1}(\boldsymbol{x}_0^*),\lambda_N(\boldsymbol{x}_0^*)) = (\nabla_{\boldsymbol{x}_0}\lambda_N)^{-T}(\boldsymbol{x}_0^*)\nabla^2_{\boldsymbol{x}_0\boldsymbol{x}_0}\widehat{\Gamma}(\boldsymbol{x}_0^*).$$

Since the latter relationship—following parts [i] and [ii]— is a multiplication between two nonsingular matrices, it follows that $\nabla_{\boldsymbol{x}_0}\theta_N(\lambda_{N-1}(\boldsymbol{x}_0^*),\lambda_N(\boldsymbol{x}_0^*))$ is nonsingular which proves [iii-a] and [iii-b]. From (2.17), and part [ii] the conclusion [iii-c] follows as well, as $\nabla_{\boldsymbol{x}_0}\lambda_N$ is continuous and thus invertible in a neighborhood of $\boldsymbol{x}_0^*$.

This completes the proof of part [iii] and of the theorem. □

2.4. Our low-memory approach. The essence of our approach follows from Theorem 1 and Theorem 3. From these theorems, the minimizer x_0^* of (2.10) and, implicitly, the first component of the minimizer of the target function (2.1) can be obtained by solving the nonlinear systems of equations in $\boldsymbol{x}_0$.

$$\theta_N(\lambda_{N-1}(\boldsymbol{x}_0),\lambda_N(\boldsymbol{x}_0)) = 0 \tag{2.18}$$

For given $\boldsymbol{x}_0$, the function on the left of the preceding equation is evaluated by computing $\lambda_i(\boldsymbol{x}_0)$ recursively using Theorem 1. In turn, the nonlinear equation (2.18) is well-posed from Theorem 3[iii]. The resulting nonlinear equation can now be solved by limited-memory quasi-Newton nonlinear equation methods such as limited-memory Broyden methods [25, 3]. Alternatively, under some conditions, the same recursion can be used to compute the objective function (2.10) and a descent direction for it, as we will illustrate in §3. In turn, this can be used in a limited-memory quasi-Newton optimization arproach such L-BFGS [3, 14].

Therefore, in principle, (2.18) can be solved by using only $\mathcal{O}(1)$ stored vectors. The only vectors that need to be stored are the current $\boldsymbol{x}_0$, the vectors at the current recursion step ($\boldsymbol{x}_i$ and $\boldsymbol{x}_{i+1}$ at the ith step of the recursion in Theorem 1), and the vectors needed by limited-memory Broyden's method. Once the convergence criterion is satisfied, the sought-after quantity (typically, the best estimate of the last state x_N^*) can be output after one more recursion.

In any case, our approach compares favorably with a brute-force minimization of (2.1) where, in principle, *all vectors* $\boldsymbol{x}_i$ need to be stored, $i = 0, 1, 2, \dots N$. For high-fidelity simulations in memory-starved environments, as the emerging high-end computing facilities appear to be, this can be a major handicap.

2.5. Comparison with the strong constraint case. Some of the difficulties in the direct approach to (2.1) appear in the case with strong constraints:

$$\begin{array}{l} \min \Gamma(\boldsymbol{x}_{0:N}) := \frac{1}{N}\left(\sum_{i=0}^{N-1}[\gamma_i(\boldsymbol{x}_i) + \phi_i(\boldsymbol{x}_i, \boldsymbol{x}_{i+1})] + \gamma_N(\boldsymbol{x}_N)\right), \\ \boldsymbol{x}_{i+1} = M_i(\boldsymbol{x}_i), \quad i = 0, 1, 2, \dots, N-1. \end{array} \tag{2.19}$$

Note that, because of the constraints, this new problem has only 1 vector degree of freedom, whereas the problem of minimizing (2.1) had $N+1$ degrees of freedom. In the 4DVar case with strong constraints, as applied operationally, the terms ϕ_i do not appear, but we preserve them for generality; they will not change our approach.

The optimality conditions for (2.19) can be obtained by introducing Lagrange multipliers $\boldsymbol{\mu}_i$, $i = 0, 1, \dots, N-1$ and the Lagrangian function

$$\mathcal{L}(\boldsymbol{x}_{0:N}, \boldsymbol{\mu}_{0:N-1}) = \Gamma(\boldsymbol{x}_{0:N}) + \sum_{i=0}^{N-1} \left(\boldsymbol{x}_{i+1} - M_i(x_i)\right)^T \boldsymbol{\mu}_i. \tag{2.20}$$

The optimality-feasibility conditions become $\nabla_{\boldsymbol{x}_i}\mathcal{L} = 0$, $i = 0, 1, \dots, N$, $\nabla_{\boldsymbol{\mu}_i}\mathcal{L} = 0$, $i = 0, 1, \dots, N-1$,. That is,

$$0 = \nabla_{\boldsymbol{x}_0}\gamma(\boldsymbol{x}_0) + \nabla_{\boldsymbol{x}_0}\phi_0(\boldsymbol{x}_0, \boldsymbol{x}_1) - \nabla_{\boldsymbol{x}_0} M_0^T(\boldsymbol{x}_0)\boldsymbol{\mu}_0 \tag{2.21a}$$

$$\begin{array}{l} 0 = \nabla_{\boldsymbol{x}_i}\gamma(\boldsymbol{x}_i) + \nabla_{\boldsymbol{x}_i}\phi_{i-1}(\boldsymbol{x}_{i-1}, \boldsymbol{x}_i) + \nabla_{\boldsymbol{x}_i}\phi_i(\boldsymbol{x}_i, \boldsymbol{x}_{i+1}) + \\ \qquad \boldsymbol{\mu}_{i-1} - \nabla_{\boldsymbol{x}_i} M_i^T \boldsymbol{\mu}_i, \quad i = 1, 2 \dots, N-1, \end{array} \tag{2.21b}$$

$$0 = \nabla_{\boldsymbol{x}_N}\gamma(\boldsymbol{x}_N) + \nabla_{\boldsymbol{x}_N}\phi_{N-1}(\boldsymbol{x}_{N-1}, \boldsymbol{x}_N) + \boldsymbol{\mu}_{N-1} \tag{2.21c}$$

$$0 = \boldsymbol{x}_{i+1} - M_i(\boldsymbol{x}_i), \quad i = 0, 1, \dots, N-1. \tag{2.21d}$$

We are now faced with two options. The first is the classical adjoint approach, which can be thought to follow from Pontryagin's principle of optimal control. That is, one can think of $\boldsymbol{x}_0$ being the only (vector) degree of freedom.

Indeed, this setup is identical to the optimal discrete nonlinear control setup [2, Proposition 3.2]. It can be seen from that reference that the situation described here corresponds to the case in which the control over the first time stage is the initial state variable, $\boldsymbol{x}_0$, and the dynamics and the objective function for the other variables do not depend on the control. Following the maximum principle in this setup, at a given $\boldsymbol{x}_0$, one computes the states by carrying out the forward recursion (2.21d) *and stores them*. Subsequently, the Lagrange multiplier $\boldsymbol{\mu}_{N-1}$ is computed from (2.21c). Then, all other Lagrange multipliers (the "adjoint variables") are computed recursively from (2.21b) *backwards* all the way to $\boldsymbol{\mu}_0$. Next, the quantity

$$\nabla_{\boldsymbol{x}_0}\mathcal{L} = \nabla_{\boldsymbol{x}_0}\gamma(\boldsymbol{x}_0) + \nabla_{\boldsymbol{x}_0}\phi_0(\boldsymbol{x}_0, \boldsymbol{x}_1) - \nabla_{\boldsymbol{x}_0} M_0^T(\boldsymbol{x}_0)\boldsymbol{\mu}_0$$

is evaluated. This is simply the derivative of the objective function restricted on the feasible manifold defined by (2.21d) but unrestricted in $\boldsymbol{x}_0$.

Subsequently, since the gradient is available, one has the option of carrying out a quasi-Newton optimization approach or, similar to the weakly constrained case described before, of solving the nonlinear equation resulting from setting the gradient to zero, that is, (2.21a). Nevertheless, note that to carry out the backward recursion, as is the case with all adjoint approaches, one needs to store at some point all vectors $x_{0:N}$, which may be a significant cost.

Alternatively and related to the approach in this work, one can look at the optimality-feasibility conditions (2.21) as the nonlinear equation

$$\nabla_{\boldsymbol{x}_N}\gamma(\boldsymbol{x}_N(\boldsymbol{x}_0)) + \nabla_{\boldsymbol{x}_N}\phi_{N-1}(\boldsymbol{x}_{N-1}(\boldsymbol{x}_0), \boldsymbol{x}_N(\boldsymbol{x}_0)) + \boldsymbol{\mu}_{N-1}(\boldsymbol{x}_0) = 0. \tag{2.22}$$

Here, the component functions of $\boldsymbol{x}_0$ are defined recursively as follows. From a prescribed $\boldsymbol{x}_0$, equation (2.21d) is solved for $i = 0$, and $\boldsymbol{x}_1(\boldsymbol{x}_0)$ is obtained. Subsequently, equation (2.21a) is solved for $\boldsymbol{\mu}_0(\boldsymbol{x}_0)$, which exists uniquely if $\nabla_x M_0(\boldsymbol{x}_0)$ is invertible (which is the case for all time resolvents). Then a recursion is carried out through (2.21b) and (2.21d), obtaining at each step $\boldsymbol{x}_{i+1}(\boldsymbol{x}_0)$ and $\boldsymbol{\mu}_i(\boldsymbol{x}_0)$ up to $i = N - 1$. At that point, all the elements needed to evaluate the left-hand side of (2.22) are computed, and that quantity can be evaluated. At this point, one can apply the limited-memory Broyden method and carry out the solution of the optimality system with $\mathcal{O}(1)$ vector storage as in the weakly constrained session.

On the other hand, the case for using the nonlinear equation—limited-memory method for the strongly constrained case—is less compelling, since for the adjoint case $\mathcal{O}(\log N)$ vector storage schemes exist by using checkpointing on the adjoint calculation while regenerating the $\boldsymbol{x}$ vectors as needed from (2.21d). While this results in substantial additional computational expense, the approach is well understood and has the advantage of leading to an optimization problem and guarantees of global convergence to stationary points. Moreover, one does not need an extra solve with $\nabla_{\boldsymbol{x}} M(\boldsymbol{x})$ at every step. Otherwise, in terms of conceptual complexity, the limited-memory quasi-Newton approach for the adjoint-optimization approach seems to be comparable to the limited-memory quasi-Newton approaches proposed in this work.

In the weakly constrained case considered here (2.1), however, the backward recursion option does not seem to exist. The reason is that the problem is now truly a problem over an $(N + 1)d$ dimensional space defined by $\boldsymbol{x}_{0:N}$, as opposed to over a d dimensional case defined by $\boldsymbol{x}_0$ in the strongly constrained case. Therefore there is no projected gradient to speak of, which is an important concept in adjoint calculations. One could consider the optimality conditions of Theorem 1 as constraints on (2.1) and then apply the approach described through Equations (2.21). Doing so, however, would require second derivatives of $M_i(x)$, which seems a steep price to pay for a first-order algorithm insofar as optimization properties are concerned. Therefore our approach in §2.3, while related with Pontryagin's maximum principle, cannot be really inferred from it. We will thus concentrate on the algorithm described in §2.3.

The comparison with the strongly constrained case reveals another interesting insight. In the control literature, the forward-nonlinear equation approach is thought of as a shooting approach for a boundary value problem. We can thus think of the approach from this work as a shooting approach for the nonlinear equation of the optimality conditions of (2.1) combined with a quasi-Newton method.

3. Low-memory algorithm for weakly constrained 4D-Var. In this section, we show how the abstract approach from §2 works in the weakly constrained

4D-Var frame. That is, we study the case when ϕ_i and γ_i are defined from (2.2) and (2.3), respectively.

First, we need to ensure that Assumption 1 holds. For the weakly constrained 4D-Var method defined in (2.1), (2.2), (2.3) and (2.4) we know that

$$\nabla_{\boldsymbol{x}_{i+1}} \nabla_{\boldsymbol{x}_i} \phi_i(\boldsymbol{x}_i, \boldsymbol{x}_{i+1}) = -(\nabla_{\boldsymbol{x}_i} \mathcal{M}_i(\boldsymbol{x}_i))^T Q^{-1}.$$

Therefore it is invertible if and only if $\nabla_{\boldsymbol{x}_i} \mathcal{M}_i(\boldsymbol{x}_i)$ is invertible. In addition, for satisfying Assumption 1 completely, $\mathcal{M}_i$, $\mathcal{H}_i$ must be continuously differentiable.

Since in most applications $\mathcal{M}_i$ represents the solution flow of a regular ordinary differential equation, the assumption that $\mathcal{M}_i$ is smooth and invertible holds. Since the observation operator $\mathcal{H}_i$ can indeed be assumed to be continuously differentiable, we conclude that Assumption 1 holds. Therefore Theorem 1 holds, and the reduced objective function stationary point of the reduced objective function (2.10) is a solution of the nonlinear equation (2.18). In addition, if the weakly constrained 4D-Var problem satisfies the strong-second order condition at its solution, then Theorem 3 also holds in this case, and the nonlinear equation is locally well-posed and solvable by limited-memory Broyden methods.

3.1. Form of the recurrence in the weakly constrained 4D-Var case. Because of quasi-quadratic form of ϕ_i and γ_i, for a fixed initial state $\boldsymbol{x}_0$, we get $\boldsymbol{x}_1$ by solving the $\theta_0(\boldsymbol{x}_0, \boldsymbol{x}_1) = 0$ as

$$\begin{aligned} \boldsymbol{x}_1 = {} & \mathcal{M}_0(\boldsymbol{x}_0) + Q_0(\nabla_{\boldsymbol{x}_0} \mathcal{M}_0(\boldsymbol{x}_0))^{-T} Q_B^{-1}(\boldsymbol{x}_0 - \boldsymbol{x}_B) + \\ & Q_0(\nabla_{\boldsymbol{x}_0} \mathcal{M}_0(\boldsymbol{x}_0))^{-T} (\nabla_{\boldsymbol{x}_0} \mathcal{H}_0(\boldsymbol{x}_0))^T R_0^{-1} \left(\mathcal{H}_0(\boldsymbol{x}_0) - \boldsymbol{y}_0\right); \end{aligned} \tag{3.1}$$

and we get $\boldsymbol{x}_{i+1}$ by solving the $\theta_i(\boldsymbol{x}_{i-1}, \boldsymbol{x}_i, \boldsymbol{x}_{i+1}) = 0$ for $i = 1, \cdots, N-1$ as

$$\begin{aligned} \boldsymbol{x}_{i+1} = {} & \mathcal{M}_0(\boldsymbol{x}_i) + Q_i(\nabla_{\boldsymbol{x}_i} \mathcal{M}_i(\boldsymbol{x}_i))^{-T} (\nabla_{\boldsymbol{x}_i} \mathcal{H}_i(\boldsymbol{x}_i))^T R_i^{-1} \left(\mathcal{H}_i(\boldsymbol{x}_i) - \boldsymbol{y}_i\right) \\ & + Q_i(\nabla_{\boldsymbol{x}_i} \mathcal{M}_i(\boldsymbol{x}_i))^{-T} Q_{i-1}^{-1} \left(\boldsymbol{x}_i - \mathcal{M}_{i-1}(\boldsymbol{x}_{i-1})\right). \end{aligned} \tag{3.2}$$

With $\boldsymbol{x}_{N-1}, \boldsymbol{x}_N$ computed by recurrence (3.2), we have

$$\begin{aligned} \theta_N(\boldsymbol{x}_{N-1}, \boldsymbol{x}_N) = {} & 2Q_{N-1}^{-1}(\boldsymbol{x}_N - \mathcal{M}_{N-1}(\boldsymbol{x}_{N-1})) - \\ & 2(\nabla_{\boldsymbol{x}_N} \mathcal{H}_N(\boldsymbol{x}_N))^T R_N^{-1}(\boldsymbol{y}_N - \mathcal{H}_N(\boldsymbol{x}_N)). \end{aligned} \tag{3.3}$$

3.2. Stability issues. The advantage of our method is most evident at large N values. On the other hand, in that regime the recursive nature of the solution opens the door to having an unstable scheme that, even if formally well defined, results in quantities too large to be practical. These difficulties are not by themselves unique to our method; the recurrence in the maximum principle approach (adjoint approach) in the case of strong constraints is also susceptible to instability if the number of steps considered is too large in relationship to the size of the eigenvalues of $\nabla_x \mathcal{M}(\boldsymbol{x})$ [2, Equation (3.38)].

Therefore, the stability of the recurrence (3.2) needs to be studied. We are particularly interested in the limit case $N \to \infty$. For a dynamical system such as Burgers' equation, given a fixed time interval, it is desirable that when the time step goes to zero (i.e., the iteration number N increases to infinite) and the time interval T is fixed, the solution of (3.2) will remain bounded.

Since a complete analysis is difficult for nonlinear systems, we will carry out this analysis for linear time-invariant systems. That is, we will investigate only the case of linear $\mathcal{M}_i$ in (2.2) and $\mathcal{H}_i$ from (2.3):

$$\mathcal{M}_i(\boldsymbol{x}_i) = A\boldsymbol{x}_i; \tag{3.4}$$
$$\mathcal{H}_i(\boldsymbol{x}_i) = B\boldsymbol{x}_i. \tag{3.5}$$

By replacing (3.4) and (3.5) in (3.2), the recurrence formula for computing $\boldsymbol{x}_{i+1}, i = 1, \cdots, N-1$, becomes

$$\begin{aligned}\boldsymbol{x}_{i+1} = &- QA^{-T}B^TR^{-1}\boldsymbol{y}_i - QA^{-T}Q^{-1}A\boldsymbol{x}_{i-1} + \\ &(QA^{-T}Q^{-1} + A + QA^{-T}B^TR^{-1}B)\boldsymbol{x}_i,\end{aligned} \tag{3.6}$$

and

$$\boldsymbol{x}_1 = -QA^{-T}(B^TR^{-1}\boldsymbol{y}_0 + Q_B^{-1}\boldsymbol{x}_B) + (QA^{-T}Q_B^{-1} + A + QA^{-T}B^TR^{-1}B)\boldsymbol{x}_0. \tag{3.7}$$

We want to allow for an asymptotic analysis with $h = \frac{T}{N} \to 0$, and with T fixed. We thus discuss how the various quantities of interest should behave with h. In the following we use the Landau notations: $a = \mathcal{O}(h)$, and, respectively $a = o(h)$ indicates that $||a/h||$ is bounded, and respectively, converges to 0, as $h \to 0$.

To mimic the discretization of a continuous dynamical system, the propagator of the dynamical system should satisfy $A = I + \mathcal{O}(h) = I + \mathcal{O}\left(\frac{T}{N}\right)$. Since the covariance matrix R models instrument error, it is reasonable to assume that it is independent of the time step, and we will thus take it constant. About the numerical error model consistency requires that the error be no larger than $\mathcal{O}(h) = \mathcal{O}(\frac{T}{N})$ the size of the time step and thus, its variance, no larger then the square of it. We make a marginally stronger assumption below.

Assumption [A] We assume that $A = A(h) = I + hP + \mathcal{O}(h^2)$, $Q = Q(h) = \psi(h)(Q_0 + \mathcal{O}(h))$, $R = \mathcal{O}(1)$, $Q_B = \mathcal{O}(1)$ for $h \to 0$. Here Q_0 is a constant invertible covariance matrix, and $\psi(h) = o(h^2)$. Here $h = T/N$, and N is the number of time intervals considered in the system.

To carry out the stability analysis under these circumstances, we first prove the following Lemma. Note that A and Q depend on h.

LEMMA 1. *Under Assumption [A]* $\left|\left|QA^{-T}Q^{-1}\right|\right|^N$, $\left|\left|QA^TQ^{-1}\right|\right|^N$ *and* $||A||^N$ *are bounded for all* h *sufficiently small.*

Proof. Let $A = I + hP_1$, with $P_1 := P_1(h) = P + \mathcal{O}(h)$. For h sufficiently small the series expansion of $(I + hP_1)$ in h holds to give

$$\begin{aligned}\left|\left|QA^{-T}Q^{-1}\right|\right|^N &= \left|\left|\sum_{i=0}^{\infty}\left(-hQP_1^TQ^{-1}\right)^i\right|\right|^N \\ &\le ||Q|| \left(\sum_{i=0}^{\infty}\left(h\left|\left|P_1^T\right|\right|\right)^i\right)^N \left|\left|Q^{-1}\right|\right| = ||Q||\left|\left|Q^{-1}\right|\right|\left(1 - h\left|\left|P_1^T\right|\right|\right)^{-N}.\end{aligned}$$

When $N \to \infty$, $\left(1 - h\,||Q||\,\left|\left|Q^{-1}\right|\right|\,\left|\left|P_1^T\right|\right|\right)^{-N} \to \exp(T\,||Q_0||\,\left|\left|Q_0^{-1}\right|\right|\,\left|\left|P^T\right|\right|)$. Similarly,

$$\left|\left|QA^TQ^{-1}\right|\right|^N = \left|\left|I + hQP_1^TQ^{-1}\right|\right|^N \le \left(1 + h\,||Q||\,\left|\left|Q^{-1}\right|\right|\,\left|\left|P_1^T\right|\right|\right)^N. \tag{3.8}$$

and $||A||^N = ||I + hP_1||^N \le (1 + h\,||P_1||)^N$. When $N \to \infty$, we have that

$$\left(1 + h\,||Q||\,\left|\left|Q^{-1}\right|\right|\,\left|\left|P_1^T\right|\right|\right)^N \to \exp(T\,||Q_0||\,\left|\left|Q_0^{-1}\right|\right|\,\left|\left|P^T\right|\right|),$$

and $(1+h\,||P_1||)^N \to \exp(T\,||P||)$. Hence the boundedness of the quantities in the statement follow since sequences admitting limits are bounded and the proof is complete. $\square$

Note that the second-order recurrence in (3.6), can be written in a matrix-vector form as

$$\begin{pmatrix} \boldsymbol{x}_{i+1} \\ \boldsymbol{x}_i \end{pmatrix} = L \begin{pmatrix} \boldsymbol{x}_i \\ \boldsymbol{x}_{i-1} \end{pmatrix} + S \begin{pmatrix} \boldsymbol{y}_i \\ 0 \end{pmatrix}, \tag{3.9}$$

where

$$L := \begin{pmatrix} D & -E \\ I_s & \mathbf{0} \end{pmatrix}, \tag{3.10}$$

$E := QA^{-T}Q^{-1}A$, I_s is the $s\times s$ identity matrix, and $D := QA^{-T}Q^{-1} + A + QA^{-T}B^TR^{-1}B$. Clearly L is a matrix with special form and we can derive L^N with some extra efforts. The first attempt is made in following lemma.

LEMMA 2. *Let U and V be $s\times s$ square matrices and* $L := \begin{pmatrix} U+V & -UV \\ I_s & 0 \end{pmatrix}$. *Then*

$$L^n = \begin{pmatrix} g_n & -g_{n-1}UV \\ g_{n-1} & g_n - g_{n-1}(U+V) \end{pmatrix}, \tag{3.11}$$

where $g_n = \sum_{i=0}^{n} V^iU^{n-i}$.

Proof. It is easy to verify that (3.11) holds for L^1 because $g_0 = I_s$, $U+V = g_1$, $-UV = -g_0UV$, $g_1 - g_0(U+V) = 0$. Assume that (3.11) holds for L^n.

$$\begin{aligned} L^{n+1} &= \begin{pmatrix} g_n & -g_{n-1}UV \\ g_{n-1} & g_n - g_{n-1}(U+V) \end{pmatrix} \begin{pmatrix} U+V & -UV \\ I_s & 0 \end{pmatrix} \\ &= \begin{pmatrix} g_n(U+V) - g_{n-1}UV & -g_nUV \\ g_n & -g_{n-1}UV \end{pmatrix}. \end{aligned} \tag{3.12}$$

We also have that

$$\begin{aligned} g_n(U+V) - g_{n-1}UV &= \sum_{i=0}^{n} V^iU^{n-i}U + \sum_{i=0}^{n} V^iU^{n-i}V - \sum_{i=0}^{n-1} V^iU^{n-1-i}UV \\ &= \sum_{i=0}^{n} V^iU^{n+1-i} + \sum_{i=0}^{n} V^iU^{n-i}V - \sum_{i=0}^{n-1} V^iU^{n-i}V \\ &= \sum_{i=0}^{n+1} V^iU^{n+1-i} = g_{n+1}. \end{aligned}$$

This proves the induction hypothesis for the upper left corner element of L^{n+1}. By rearranging the above equality we obtain $g_{n+1} - g_n(U+V) = -g_{n-1}UV$, which demonstrates the induction hypothesis for the lower right element. Since the other elements of L^{n+1} are in the algebraic form required by the induction hypothesis, the proof completes. $\square$

However, the matrix L in Lemma 2 is still a bit different in our case, a case we begin to investigate with the following Lemma.

LEMMA 3. *Let U, V and C be $s \times s$ square matrices, and*

$$L := \begin{pmatrix} U+V+C & -UV \\ I_s & 0 \end{pmatrix}.$$

Then

$$L^n = \begin{pmatrix} f_n & -f_{n-1}UV \\ f_{n-1} & -f_{n-2}UV \end{pmatrix}, \tag{3.13}$$

where $f_n(U+V+C) - f_{n-1}UV = f_{n+1}$, with $f_{-1} = \mathbf{0}_s$, $f_0 = I_s$.

Proof. Let f_n be defined by the above recursion and initial conditions. The case $n = 1$ immediately holds, and the recursion relation can immediately be verified by inspection. □

The difficulty with Lemma 3.13 is that the term C makes a general solution for f_n very complicated algebraically. To reduce the calculation of f_n to the calculation of g_n, we prove the following.

LEMMA 4. *Let $J_1(h), J_2(h) \in \mathbb{R}^{s\times s}$ be matrices satisfying $J_1(h) = J_1^0 + \mathcal{O}(h)$; $J_2(h) = J_2^0 + \mathcal{O}(h)$; such that J_1^0 and $-J_2^0$ have no common eigenvalues and $C(h) \in \mathbb{R}^{s\times s}$, $C(h) = o(h^2)$. Define the matrices $U(h) = I_s + hJ_1(h)$; $V(h) = I_s - hJ_2(h)$. There exists h_0 such that $\forall 0 \le h \le h_0$ there exist the matrices $\widehat{U}(h)$ and $\widehat{V}(h)$ satisfying*

$$\widehat{U}(h) + \widehat{V}(h) = C(h) + U(h) + V(h), \quad \widehat{U}(h)\widehat{V}(h) = U(h)V(h) \tag{3.14}$$

and

$$\left|\left|\widehat{U}(h) - U(h)\right|\right| = o(h), \quad \left|\left|\widehat{V}(h) - V(h)\right|\right| = o(h). \tag{3.15}$$

Proof. We write (3.14) in an equivalent form, by introducing the matrix-valued mappings $\Psi_1(h)$, $\Psi_2(h)$ satisfying $\widehat{U}(h) = I_s + hJ_1(h) + h\Psi_1(h)$, and $\widehat{V}(h) = I_s - hJ_2(h) - h\Psi_2(h)$. It immediately follows that the first equation in (3.14) is equivalent to

$$h\left(\Psi_1(h) - \Psi_2(h)\right) = C(h). \tag{3.16}$$

Replacing the same ansatz in the second equation of (3.14) we obtain that

$$\begin{aligned} &\left(I_s + hJ_1(h) + h\Psi_1(h)\right)\left(I_s - hJ_2(h) - h\Psi_2(h)\right) = \left(I_s + hJ_1(h)\right)\left(I_s - hJ_2(h)\right) \Leftrightarrow \\ &\quad h\Psi_1(h)\left(I_s - hJ_2(h)\right) - \left(I_s + hJ_1(h)\right)h\Psi_2(h) - h^2\Psi_1(h)\Psi_2(h) = 0 \end{aligned}$$

Replacing now $\Psi_2(h)$ from (3.16) in the last relationship, and dividing by h^2 we obtain that (3.14) holds if and only if there exists $\Psi = \Psi_1(h)$ such that

$$\Theta(h;\Psi) := -\Psi\left(J_2(h) + \frac{C(h)}{h}\right) - J_1(h)\Psi + \frac{C(h)}{h^2} + J_1(h)\frac{C(h)}{h} + \Psi^2 = 0 \tag{3.17}$$

By our assumptions, the mapping $\Theta(h;\Psi)$ is continuous in h and infinitely differentiable in Ψ (in effect, polynomial), and so are all its derivatives with respect to Ψ. It also satisfies $\Theta(0, \mathbf{0}_s) = 0$. The action of its Ψ derivative at the point $(0, \mathbf{0}_s)$ along a direction Ψ_d (which can be seen as a matrix in $\mathbb{R}^{s\times s}$, making the derivative a 4 dimensional tensor) satisfies:

$$\nabla_\Psi \Theta(0, \mathbf{0}_s)\Psi_d = -\Psi_d J_1^0 - J_2^0 \Psi_d. \tag{3.18}$$

The right hand side of (3.18) is closely connected to Sylvester's equation: $AX+XB = C$, where $A, B, C \in \mathbb{R}^{s\times s}$ and X is an unknown matrix in $\mathbb{R}^{s\times s}$. If A and $-B$ have no common eigenvalues, then Sylvester's equation has a unique solution X for every C [22, Theorem 1.16]. As J_1^0 and $-J_2^0$ have no common eigenvalues, it follows from the properties of Sylvester's equation that, the mapping $\nabla_\Psi \Theta(0, \mathbf{0}_s)\Psi_d$ is one-to-one and onto on $\mathbb{R}^{s\times s}$, and, thus, invertible with an inverse we denote by $\nabla_\Psi \Theta^{-1}$. This makes the equation in Ψ (3.17), $\Theta(h, \Psi) = 0$, regular at $(0, \mathbf{0}_s)$, and thus defines locally Ψ uniquely as a function of h.

Since all the derivatives with Ψ of Θ are continuous in h it follows that there exists a neighborhood of $(0, \mathbf{0}_s)$ in which Θ, $\nabla_\Psi \Theta$, and $\nabla_\Psi(\Theta)^{-1}$ exist, and are continuous and their norms are bounded by C_θ. Moreover, $\nabla_\Psi \Theta$ is uniformly Lipschitz in Ψ with respect to h (as it is differentiable, and its derivative is continuous in h and Ψ). We assume without loss of generality that the Lipschitz constant is upper bounded by C_θ.

We also have from (3.17) that $\Theta(h; \mathbf{0}_s) = \frac{C}{h^2} + \frac{C}{h}\Psi_1(h) = \beta(h)$, with $||\beta(h)|| \to 0$ as $h \to 0$. It then follow that there exists an h_0 such as $\alpha(h) = \eta(h)C_\theta^2 \le \frac{1}{2}$, $\forall 0 \le h \le h_0$, where

$$\eta(h) = \left|\left|\nabla_\Psi \Theta(h, \mathbf{0}_s)^{-1}\Theta(h, \mathbf{0}_s)\right|\right| \le C_f \left|\left|\beta(h)\right|\right|. \tag{3.19}$$

As a result, the conditions for Kantorovich's theorem [16, §12.6.2] are met. There exists a solution of the equation $\Theta(h, \Psi_1(h)) = 0$ satisfying $||\Psi_1(h)|| \le C_\Psi \beta(h)$ for some $C_\Psi > 0$ and all $h \le h_0$.

From the equivalence of (3.17) with (3.14) it follows that $\widehat{U}(h)$ and $\widehat{V}(h)$ exist and satisfy $\widehat{U}(h) - U(h) = h\Psi_1(h) = o(h)$, and $\widehat{V}(h) - V(h) = h\Psi_2(h) = C - h\Psi_1(h) = o(h)$. This proves (3.15) and the claim. □

The key bounding calculation is now provided by the following Lemma.

Lemma 5. *Let f_n be the sequence from Lemma 3 as applied to* (3.9)–(3.10). *To this end, we use the identification $U = QA^{-T}Q^{-1}$, $V = A$, and $C = QA^{-T}B^TR^{-1}B$. Assume that Assumption [A] holds and that P^T and $-P$ have no common eigenvalues. Then the following hold:*

i $\frac{1}{N}||f_n||$ *is bounded for all N and $1 \le n \le N$.*

ii For any ϵ there exists N_0 such that for all $N \ge N_0$, we have that

$$\left|\left|f_N - f_{N-1}QA^{-T}Q^{-1}\right|\right| < \left|\left|e^{PT} - I_s\right|\right| + \epsilon.$$

Note that Q, A depend on $h = \frac{T}{N}$ as defined in Assumption [A].

Proof. We first verify that the conditions needed to use Lemma 4 apply. With the definition of U we have that $U(h) = Q(h)A^{-T}(h)Q(h)^{-1} = I + hQ_0P^TQ_0^{-1} + \mathcal{O}(h^2)$, $V(h) = A(h) = I + hP + \mathcal{O}(h^2)$, and $C(h) = o(h^2)$. Moreover $Q_0P^TQ_0^{-1}$ has the same eigenvalues as P^T and P. Therefore $Q_0P^TQ_0^{-1}$ has common eigenvalues with $-P$ if and only if P^T and $-P$ do, which is excluded by our hypothesis.

Therefore the conclusions of Lemma 4 apply to give matrices $\widehat{U}$ and $\widehat{V}$ satisfying (3.14) and (3.15). It then follows that the matrix L in (3.10) has the same form as in 2, and application of that result in conjunction with the definition of f_n in Lemma 3 results in

$$f_n = \sum_{i=0}^{n} \widehat{V}^i \widehat{U}^{n-i}. \tag{3.20}$$

In turn, this implies that $\frac{||f_n||}{N+1} \le \max\left\{\left|\left|\widehat{V}\right|\right|^N, \left|\left|\widehat{U}\right|\right|^N\right\}$. From the fact that $h = \frac{T}{N}$ and (3.15) it follows that $\left|\left|\widehat{V}\right|\right|^N \le (1 + h\,||P|| + o(h))^N \to \exp\{(||P||\,T)\}$ and the sequence is thus bounded. From similar arguments, so is $\left|\left|\widehat{V}\right|\right|$ which proves part [i] of the claim.

For part [ii] we notice from (3.20) that

$$f_n - f_{n-1}\widehat{U} = \sum_{i=0}^{n} \widehat{V}^i \widehat{U}^{n-i} - \sum_{i=0}^{n-1} \widehat{V}^i \widehat{U}^{n-i} = \widehat{V}^n. \tag{3.21}$$

Using from (3.15) that $QA^{-T}Q^{-1} = \widehat{U} + o(h)$, we obtain that

$$\begin{aligned} \left|\left|f_N - f_{N-1}QA^{-T}Q^{-1} - I_s\right|\right| &\le \left|\left|f_N - f_{N-1}\widehat{U} - I_s\right|\right| + ||f_{N-1}||\, o(h) \\ &= \left|\left|\widehat{V}^N - I_s\right|\right| + \frac{||f_{n-1}||}{N}\,(No(h)) \to ||\exp\{PT\} - I_s||\,. \end{aligned}$$

The last relationship follows from the fact that $\frac{||f_{n-1}||}{N}$ is bounded from part [i] whereas $No(h) = \frac{o(h)}{h} \to 0$, as well as the fact that $\widehat{V} = I + hP + o(h)$. From the properties of the limit the proof is complete. ▯

Discussion The only assumption we made beyond Assumption [A] is that P and $-P^T$ have no common eigenvalues. This is the case for example if A is the propagator of the dynamical system $\frac{dx}{dt} = Px$, where P is a stable matrix. Therefore, the condition is satisfied if the target system is stable.

THEOREM 4. *Suppose that the sequence $\boldsymbol{x}_i, i = 1, \cdots, N$ is derived by recurrence formula* (3.2)*, and θ_N is computed by* (3.3)*. Then $||\nabla_{\boldsymbol{x}_0}\boldsymbol{x}_N||$ is bounded as $N \to \infty$ and thus the recurrence* (3.6) *is stable.*

Proof. We first prove that

$$\nabla_{\boldsymbol{x}_0}\boldsymbol{x}_N = f_N + f_{N-1}\left(-QA^{-T}Q^{-1} + QA^{-T}Q_B^{-1}\right)$$

where f_n is defined in Lemma 3.

The second-order recurrence (3.6) can be written as (3.9) with L defined in (3.10). Let L_1^N, L_2^N, L_3^N, and L_4^N denote the upper left block, upper right block, bottom left block, and bottom left block of L^N, respectively. According to (3.9), we will have

$$\nabla_{\boldsymbol{x}_0}\boldsymbol{x}_N = L_3^N \frac{\partial \boldsymbol{x}_1}{\boldsymbol{x}_0} + L_4^N = L_3^N\left(QA^{-T}Q_B^{-1} + A + QA^{-T}B^TR^{-1}B\right) + L_4^N. \tag{3.22}$$

Let E, D as in (3.10) According to Lemma 3, $L_3^N = f_{N-1}$, $L_4^N = -f_{N-2}E$, and $f_nD - f_{n-1}E = f_{n+1}$. Moreover, we have that

$$\begin{aligned} \nabla_{\boldsymbol{x}_0}\boldsymbol{x}_N &= f_{N-1}D - f_{N-2}E + f_{N-1}\left(QA^{-T}Q_B^{-1} - QA^{-T}Q\right) \\ &= f_N + f_{N-1}\left(QA^{-T}Q_B^{-1} - QA^{-T}Q^{-1}\right) \end{aligned}$$

In turn, this leads to the inequality

$$||\nabla_{\boldsymbol{x}_0}\boldsymbol{x}_N|| \le \left|\left|f_N - f_{N-1}QA^{-T}Q^{-1}\right|\right| + ||f_{N-1}||\,\left|\left|QA^{-T}Q_B^{-1}\right|\right| \tag{3.23}$$

From Lemma 5[ii] the first term is bounded, whereas the second term can be written as $\frac{f_{N-1}}{N}\left|\left|NQA^{-T}Q_B^{-1}\right|\right|$, of which the first factor is bounded from Lemma 5[i] and the second factor is $o(h)$ from Assumption [A]. Consequently $||\nabla_{\boldsymbol{x}_0}\boldsymbol{x}_N||$ is bounded which proves the claim. ∎

This result proves that even as $N \to \infty$, the essential components of our algorithm will stay bounded. In that regime, as our algorithm stores $\mathcal{O}(1)$ vectors our storage will be $\mathcal{O}(1/N)$ relative to a classical approach which stores all vectors x_i; a large and increasing storing efficiency.

3.3. Optimization-based low-memory approach. Here we investigate the possibility of obtaining a descent direction for $\widehat{\Gamma}$ *by doing forward sweeps only.* The advantage of such an approach compared to an adjoint one is that no information needs to be stored for a reverse sweep which ensures a low-memory behavior. The aim is to find a vector which is guaranteed to have a positive inner product with $\nabla_{\boldsymbol{x}_0}\widehat{\Gamma}$. In turn, this would provide a theoretical basis for using limited-memory, optimization-based quasi-Newton methods such as L-BFGS methods [14].

We prove the main results for optimization-based approaches below.

LEMMA 6. *Suppose that assumption [A] holds and that the sequence $\boldsymbol{x}_i, i = 1, \cdots, N$ is derived by the recurrence formula* (3.2)*, and θ_N is computed by* (3.3)*. Then there exists a T_δ and an N_0, such that $\nabla_{\boldsymbol{x}_0}\boldsymbol{x}_N$ is positive definite for $T < T_\delta$ and $N \geq N_0$.*

Proof. We use the same notations as in the proof of Theorem 4. Following on (3.23) and invoking Lemma 5 we obtain that $G_N = \nabla_{\boldsymbol{x}_0}\boldsymbol{x}_N$ satisfies $||G_N - I_s|| \to ||\exp(PT) - I_s||$. Choose now T_δ such that $||\exp(PT) - I_s|| \leq \frac{1}{4}$, $\forall T \leq T_\delta$. Then, from the preceding limit, there exists N_0 such that $||G_N - I_s|| \leq \frac{1}{3}$, $\forall N \geq N_0$. Since this implies that $\left|\left|G_N^T - I_s\right|\right| \leq \frac{1}{3}$, it follows that $\left|\left|\frac{G_N^T + G_N}{2} - I_s\right|\right| \leq \frac{1}{3}$, and thus $G_N^T + G_N$ is symmetric and positive definite, and so is G_N. This proves the claim. ∎

THEOREM 5. *Suppose that Assumption [A] holds and the sequence $\boldsymbol{x}_i, i = 1, 2, \cdots, N$ is derived by the recurrence formula* (3.2)*, and θ_N is computed by* (3.3)*. Then there exists a T_δ, such that $(\nabla_{\boldsymbol{x}_0}\widehat{\Gamma})^T\theta_N$ is positive for $T < T_\delta$.*

Proof. From (2.3), we have

$$(\nabla_{\boldsymbol{x}_0}\widehat{\Gamma})^T\theta_N = (\theta_N)^T(\nabla_{\boldsymbol{x}_0}\boldsymbol{x}_N)\theta_N \tag{3.24}$$

According to Lemma (6), there exists a T_δ, such that $\nabla_{\boldsymbol{x}_0}\boldsymbol{x}_N$ is positive definite for $T < T_\delta$.

Hence the proof is complete. ∎

The significance of the result of Theorem 5 is that scaling the vector θ_N obtained from the forward recursion (2.9) will now provide a descent direction for $\widehat{\Gamma}(\boldsymbol{x}_0)$ when the time interval is small enough under the conditions described in the Theorem.

Of course, the condition $T \leq T_\delta$ may be quite limiting. On the other hand, as proved in Theorem 3 [iii-c] we have that θ_N will be proportional with the distance from the current point to the solution. Therefore its size is proportional to the one of the gradient, and while we cannot ensure it will provide a descent direction, it is quite likely that either it or its reciprocal will provide a substantial descent. So we will use it even for T larger than Theorem 5 with an expectation that it could work well, even as it cannot be generally proved to be the case.

3.4. Regularity of the weakly constrained 4D-Var problem. We now prove the counterpart of Theorem 3 in 4D-Var framework. We first find the Hessian matrix of Γ. Define

$$W_i := \left(\nabla_{\boldsymbol{x}_i}\mathcal{H}_i(\boldsymbol{x}_i)\right)^T R_i^{-1}\nabla_{\boldsymbol{x}_i}\mathcal{H}_i(\boldsymbol{x}_i)+$$

$$\Big(\big((\boldsymbol{x}_{i+1}-\mathcal{M}_i(\boldsymbol{x}_i))^T Q_i^{-T}\big)\bigotimes I_s\Big)\nabla_{\boldsymbol{x}_i}\mathrm{vec}\big((\nabla_{\boldsymbol{x}_i}\mathcal{M}_i(\boldsymbol{x}_i))^T\big)+$$
$$\Big(\big((\boldsymbol{y}_i-\mathcal{H}_i(\boldsymbol{x}_i))^T R_i^{-T}\big)\bigotimes I_m\Big)\nabla_{\boldsymbol{x}_i}\mathrm{vec}\big((\nabla_{\boldsymbol{x}_i}\mathcal{H}_i(\boldsymbol{x}_i))^T\big)$$

for $i=0,\cdots,N$. Here S is a symmetric block tridiagonal matrix with $V_i, i=0,\cdots,N$ as diagonal and $-U_i, i=0,\cdots,N-1$ as subdiagonal, where

$$\begin{aligned}
U_i &:= \big(\nabla_{\boldsymbol{x}_i}\mathcal{M}_i(\boldsymbol{x}_i)\big)^T Q_i^{-1}, \quad i=0,\cdots,N-1\\
V_i &:= Q_{i-1}^{-1}+\big(\nabla_{\boldsymbol{x}_i}\mathcal{M}_i(\boldsymbol{x}_i)\big)^T Q_i^{-1}\nabla_{\boldsymbol{x}_i}\mathcal{M}_i(\boldsymbol{x}_i), \quad i=0,\cdots,N-1\\
V_0 &:= Q_B^{-1}+\big(\nabla_{\boldsymbol{x}_0}\mathcal{M}_0(\boldsymbol{x}_0)\big)^T Q_0^{-1}\nabla_{\boldsymbol{x}_0}\mathcal{M}_0(\boldsymbol{x}_0), \quad V_N := Q_{N-1}^{-1}.
\end{aligned}$$

Because Γ takes the special form as in (2.1)–(2.4), its Hessian matrix is a block tridiagonal matrix. We can verify that $U_i=-N\nabla_{\boldsymbol{x}_i}\nabla_{\boldsymbol{x}_{i+1}}\Gamma(\boldsymbol{x}_{0:N})$, $V_i+W_i=N\nabla^2_{\boldsymbol{x}_i}\Gamma(\boldsymbol{x}_{0:N})$.

$$\nabla^2_{\boldsymbol{x}_{0:N}}\Gamma(\boldsymbol{x}_{0:N})=\frac{1}{N}(S+\mathrm{diag}(W_0,\cdots,W_N)). \tag{3.25}$$

LEMMA 7. *Suppose that the first- and second-order derivatives of $\mathcal{M}_i$ and $\mathcal{H}_i$ are bounded; $\nabla_x\mathcal{M}_i$ is nonsingular; Q_i, R_i, Q_B are positive definite (all of which are standard 4D-Var conditions); and W_i are positive semi-definite at the solution $\boldsymbol{x}^*_{0:N}$, $i=0,1,\ldots,N$. Then $\nabla^2_{\boldsymbol{x}_{0:N}}\Gamma(\boldsymbol{x}_{0:N})$ is positive definite at that solution.*

The significance of this result is that the optimization problem of the weakly constrained 4D-Var satisfies the second-order sufficient condition. Therefore, from Theorem 3, the nonlinear equation (2.18) is well-posed and thus can be solved by the limited-memory Broyden method. Of all the conditions invoked, only the one concerning the positive definiteness of W_i is nonstandard. They hold, for example, for linear systems or for the case where the model and observation error is 0 at the solution. Note, however, that these conditions are sufficient but not necessary for well-posedness of the nonlinear equation (2.18). The only necessary condition is the second-order condition, though it is of course difficult to ensure a priori in all nonlinear problems for any variational approach, including ours.

Proof. If $Z\neq 0$, then

$$Z^T S Z=\boldsymbol{z}_0^T Q_B^{-1}\boldsymbol{z}_0+\sum_{i=0}^{N-1}\left(\nabla_{\boldsymbol{x}_i}\mathcal{M}_i(\boldsymbol{x}_i)\boldsymbol{z}_i-\boldsymbol{z}_{i+1}\right)^T Q_i^{-1}\left(\nabla_{\boldsymbol{x}_i}\mathcal{M}_i(\boldsymbol{x}_i)\boldsymbol{z}_i-\boldsymbol{z}_{i+1}\right)>0.$$

The inequality holds because if the right-hand side is 0, then $\boldsymbol{z}_{i+1}=\big(\nabla_{\boldsymbol{x}_i}\mathcal{M}_i(\boldsymbol{x}_i)\big)\boldsymbol{z}_i$; and $\boldsymbol{z}_0=0$, which in turn implies $Z=0$, a contradiction. If W_i positive semi-definite, then the Hessian matrix of Γ, (3.25) is positive definite, and the proof is complete. ∎

4. Numerical experiments. We now present numerical experiments that illustrate the theoretical findings discussed in §2 and §3. We solve both the nonlinear formulation (2.18), which we expect to be regular based on Theorem 3 and the optimization formulation with the objective (2.10), where a descent direction is obtained based on Theorem 5. To solve the nonlinear equation (2.18) in a low-memory fashion we use the limited-memory Broyden method defined in [25], whereas for the optimization approach we use limited memory BFGS [14].

4.1. Model problem. In this study we focus on Burgers' equation ([1, 10, 18]), which describes the interaction between nonlinear advection and turbulent dissipation. This equation is a fundamental problem in fluid mechanics and has been used

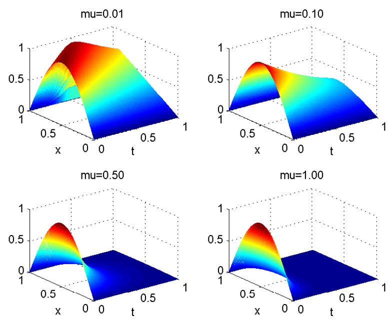

FIG. 4.1. *Burgers' equation with viscosity coefficient* $\mu = 0.01, 0.1, 0.5, 1$ *with initial condition* $u(0, x) = sin(\pi x)$.

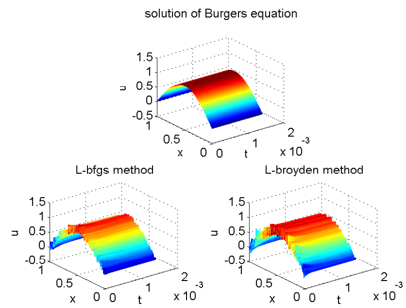

FIG. 4.2. *Plots of numerical solutions of L-BFGS (bottom left)and L-Broyden(bottom right) methods and the solution of the Burgers equation (top) for* $\mu = 0.01$, $\triangle t = \triangle x/1000$, *and* $N = 700$.

extensively as a benchmark in meteorology (see [10] and references therein). The inviscid form ($\mu = 0$) is also important because it captures the essence of the large-scale transient waves of mid-latitudes. Variational data assimilation for Burgers equation is discussed by [1].

Burgers' equation has the following definition:

$$\frac{\partial u}{\partial t} + \frac{1}{2}\frac{\partial (u^2)}{\partial x} = \mu \frac{\partial^2 u}{\partial x^2}, x \in (0,1) \times (0,T), \mu > 0, \tag{4.1a}$$

$$u(0,t) = u(1,t) = 0; \tag{4.1b}$$

$$u(x,0) = u_0(x). \tag{4.1c}$$

Here μ is the viscosity coefficient. The solution of Burgers' equation with viscosity coefficient $\mu = 0.01, 0.1, 0.5, 1$ is shown in Fig. 4.1.

As seen in Fig. 4.1, the function value drops sharply when the viscosity coefficient is larger than 0.5. In such cases, the information content is limited, and therefore, we choose the cases when μ is small.

In terms of the numerical discretization of the problem, we let u_j^m denote the function value $u(j\triangle x, m\triangle t)$. According to [1], a centered finite-difference scheme for Burgers' equation is

$$\frac{u_j^{m+1} - u_j^m}{\triangle t} + \frac{(u_{j+1}^m)^2 - (u_{j-1}^m)^2}{4\triangle x} - \frac{\mu}{(\triangle x)^2}(u_{j+1}^{m+1} - 2u_j^{m+1} + u_{j-1}^{m+1}) = 0. \tag{4.2}$$

Let U^m denote the vector determined by $u_j^m, j = 0, \cdots, N$. The scheme in (4.2) results in a discrete dynamical system that can be written compactly as $PU^{m+1} = \mathcal{S}(U^m)$. Here, P is a symmetric tridiagonal matrix with $(1 + 2\mu(\triangle t)/(\triangle x)^2)$ on the diagonal and $-\mu(\triangle t)/(\triangle x)^2$ on the sub- and superdiagonal. This defines the discrete dynamical mapping $\mathcal{M}(\cdot)$ discussed in §2 and §3. Specifically, we have that $\mathcal{M}_i(U^i) = P^{-1}\mathcal{S}(U^i)$, and $\nabla\mathcal{M}_i(U^i) = P^{-1}\nabla\mathcal{S}(U^i)$. Obviously, $P = I + \frac{T}{N}B^0$, B^0 is a tridiagonal matrix, with $\frac{2\mu}{(\triangle x)^2}$ on the diagonal and $-\frac{\mu}{(\triangle x)^2}$ above and below, and $\nabla\mathcal{S}(U^i) = I - \frac{T}{N}(B^1)$, where B^1 is the tridiagonal matrix with zero on the diagonal, $\frac{U^i_{2:N}}{2\triangle x}$ on the superdiagonal, and $-\frac{U^i_{1:N-1}}{2\triangle x}$ on the subdiagonal. Hence $\nabla\mathcal{M}_i(U^i) =$

$I+\frac{T}{N}B^3+\cdots$ with $B^3 = B^1-B^0$. Therefore $\mathcal{M}(\cdot)$ satisfies all the conditions required of it for the theoretical developments in §2 and §3.

However, not every finite-difference scheme has this property. A counterexample is the implicit Lax-Friedrichs scheme discussed by [1]. This scheme uses the average of u^m_{j+1} and u^m_{j-1} in place of u^i_j. By doing so, leads to $\nabla\mathcal{M}_i(U^i)$ violating Assumption [A].

4.2. Numerical results. We now describe in detail the numerical experiments, the objective being the minimization of (2.10). The function $\mathcal{M}_i$ in (2.2) is derived from the centered finite-difference scheme applied to Burgers' equation. We consider $\triangle x = 1/501$ and initial state $U^0 = \sin(\pi x)$ to generate the data set $\mathcal{G} := \{U^0, \mathcal{M}_i(U^i), i = 1,\cdots,N\}$. The observation data are computed by applying $\mathcal{H}_i(\mathcal{G})$ and perturbed by normal random noise times with standard deviation 0.1 to mimic the action of a noisy nonlinear operator. To be closer to a real situation, the observations are taken every 10 steps in space-time (i.e., at time node $i10\triangle t$ and space node $i10\triangle x$).

We use the L-BFGS algorithm to compute the minimizer of (2.10) (but with search direction Θ_N as indicated by Theorem 5). We also use the L-Broyden algorithm to compute the solution of (2.18). We choose Q to be a diagonal matrix $(\triangle t)^2[2,1,\cdots,1,2]$ on diagonal and Q_B and R to be $100\cdot I$. The initial solution U^0 is perturbed with normal random noise times with standard deviation 0.1 and used as the initial guess for this algorithm. Note that only the y-axis of each plot of results is set to be log scaled. Also note that all numerical results are scaled by the corresponding values of the initial guess.

In Fig. 4.3 and 4.4, we plot function values of $\widehat{\Gamma}$ as in (2.10) at each iteration of the L-BFGS algorithm. We compare the results obtained by using different numbers of stored vectors $p = 2,4,6,8$ for $N = 700$ in Fig. 4.3. Note that the convergence rates are similar. In 4.4, we plot the results of $N = 800, 900, 1000, 1100$. In Fig. 4.5 and 4.6, we plot function values of (2.10) at each iteration of the L-Broyden algorithm. In Fig. 4.7 and 4.8 we show the norms of residuals of (3.3) at each iteration of the L-Broyden algorithm. In Fig. 4.5 and 4.6 we compare the results obtained when using the L-Broyden method for different $p = 2,4,6,8$ when $N = 700$. The results for L-Broyden with $N = 800, 900, 1000, 1100$ for $p = 4$ are shown in Fig. 4.7 and 4.8. We see from our numerical simulations that the objective function is significantly reduced (by 2-5 orders of magnitude).

Though the problems are not solved to high accuracy the solution does approach a perturbed version of the original solution, as can be seen in Fig. 4.2. There, we illustrate the numerical solutions of L-BFGS (bottom left) and L-Broyden (bottom right) methods together with the solution of the Burgers equation (top) for $\mu = 0.01$, $\triangle t = \triangle x/1000$, and $N = 700$.

Certainly, this is a limited set of experiments, for example $\triangle t$ is much smaller than would be used in practical problems, and for large $\triangle t$ we have definitely seen the instability that we analyze in §3.2 and which we can guarantee to not occur only for fixed T and $\triangle t$ sufficiently small. Also, we do not find in the experiments a large dependence with p which is uncommon for quasi-Newton methods, which also indicates that the circumstances here are quite particular.

Nevertheless, in these limited circumstances (which are the only ones in which we can guarantee at the moment the method to work for large and increasing N, where the method would be practically interesting) we observe that p can stay essentially $\mathcal{O}(1)$ and still achieve convergence. We can see from the results described above that

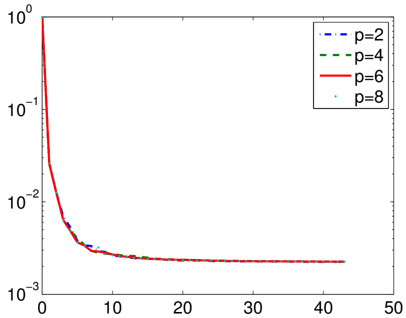

Fig. 4.3. *Scaled function value of* (2.10) *at each iteration of L-BFGS for* $\mu = 0.01$, $\triangle t = \triangle x/1000$, $N = 700$, *and* $p = 2, 4, 6, 8.$.

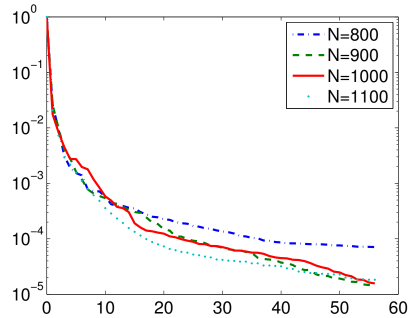

Fig. 4.4. *Scaled function value of* (2.10) *at each iteration of L-BFGS for* $\mu = 0.01$, $\triangle t = \triangle x/1000$, $p = 6$, *and* $N = 800, 900, 1000, 1100$.

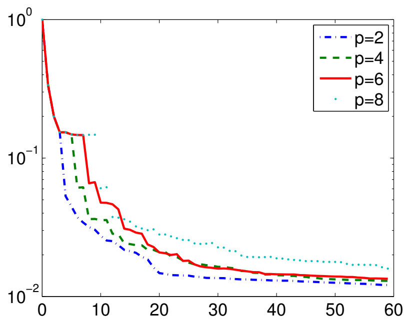

Fig. 4.5. *Scaled function values of* $\widehat{\Gamma}$ *as in* (2.10) *at each iteration of L-Broyden algorithm for* $p = 2, 4, 6, 8$, $\mu = 0.01$, $\triangle t = \triangle x/1000$ *and* $N = 700$.

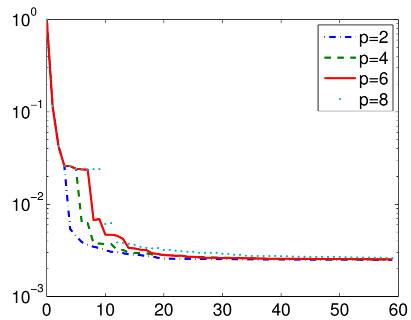

Fig. 4.6. *Scaled norms of residuals of* (3.3) *at each iteration of L-Broyden algorithm for* $p = 2, 4, 6, 8$, $\mu = 0.01$, $\triangle t = \triangle x/1000$ *and* $N = 700$.

the L-BFGS method using only forward sweeps converges faster than L-Broyden, though our theory here through Theorem 5 applies only in the regime of small T. Overall, we find that the numerical experiments validate the findings from §2 and §3, that the nonlinear equation obtained by our reduction procedure (2.10) is well-posed and can be solved both by using the L-Broyden method or L-BFGS method with forward sweeps only even though p is much smaller than the dimension of x. That is, the memory savings compared with that of the full method are significant: we use little relative storage (i.e., $p \ll N$, some of our experiments have even produced good results for $N/p > 100$).

5. Conclusions. Hidden Markov models with physical model error pose new challenges to data assimilation. One of these challenges is the fact that, being weakly constrained, the model can no longer be used to reduce the storage needs by deriving a state from another state. Instead, the entire estimated trajectory must be stored. This challenge is particularly burdensome with the emergence of new architectures where less memory will be available per node.

We addressed this challenge by using a new approach, which constrains the problem with the optimality conditions at the states other than initial. In turn, this results in a nonlinear equation whose residual vector can be computed by forward sweeps only,

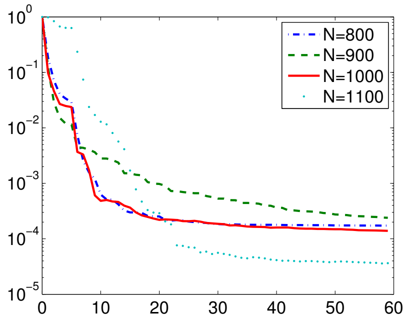

Fig. 4.7. *Scaled function values of $\widehat{\Gamma}$ as in* (2.10) *at each iteration of L-Broyden algorithm for* $N = 800, 900, 1000, 1100,\ p = 4,\ \mu = 0.01,\ \triangle t = \triangle x/1000$.

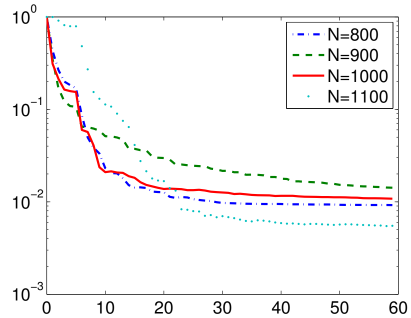

Fig. 4.8. *Scaled norms of residuals of* (3.3) *at each iteration of L-Broyden algorithm for* $N = 800, 900, 1000, 1100,\ p = 4,\ \mu = 0.01$ $\triangle t = \triangle x/1000$.

or an optimization problem where an approximation to the gradient can be computed with foward sweeps only. In turn, no intermediate states need to be stored for advancing the best estimate algorithm. In conjunction with limited-memory algorithms (Broyden or BFGS) we can solve such a problem with low or even $\mathcal{O}(1)$ storage. In a numerical experiment using Burgers' equation, we achieved up to 100 times reduction in memory usage while computing the solution. We demonstrated this finding with a numerical experiment using Burgers' equation.

On the other hand, the approach poses other issues; in particular, it is prone to instability problems. Our proofs in the interesting case, the one of large N, work only in the limit of small time step h and the numerical demonstrations are also done in this regime. To make the method practical beyond this context, we will pursue several other avenues, such as multiple shooting and preconditioning based on a coarser or reduced system. Nevertheless, we believe that algorithms reducing storage (and implicitly, communication) are important issues that this method helps address.

Acknowledgment. This work was supported by the Department of Energy under Contract No. DE-AC02-06CH11357. We are grateful to Jorge Moré for advice and assistance on limited-memory BFGS methods.

REFERENCES

[1] A. Apte, D. Auroux, and M. Ramaswamy, *Variational data assimilation for discrete Burgers equation*, in Electronic Journal of Differential Equations, vol. 19, 2010, pp. 15–30.
[2] D.P. Bertsekas, *Dynamic programming and optimal control*, Athena Scientific Belmont, MA, 1995.
[3] R.H. Byrd, J. Nocedal, and R.B. Schnabel, *Representations of quasi-Newton matrices and their use in limited memory methods*, Mathematical Programming, 63 (1994), pp. 129–156.
[4] P. Courtier, J.N. Thepaut, and A. Hollingsworth, *A strategy for operational implementation of 4D-Var, using an incremental approach*, Quarterly Journal of the Royal Meteorological Society, 120 (1994), pp. 1367–1387.
[5] R. Daley, *Atmospheric data analysis*, Cambridge University Press, 1993.
[6] A. Geist and S. Dosanjh, *IESP exascale challenge: co-design of architectures and algorithms*, International Journal of High Performance Computing Applications, 23 (2009), p. 401.
[7] J. Glimm, S. Hou, Y.H. Lee, D.H. Sharp, and K. Ye, *Sources of uncertainty and error in the simulation of flow in porous media*, Computational & Applied Mathematics, 23 (2004),

pp. 109–120.
[8] E. Kalnay, *Atmospheric Modeling, Data Assimilation and Predictability*, Cambridge University Press, 2003.
[9] F.X. Le Dimet and O. Talagrand, *Variational algorithms for analysis and assimilation of meteorological observations: theoretical aspects*, Tellus A, 38 (1986), pp. 97–110.
[10] J.M. Lewis, S. Lakshmivarahan, and S.K. Dhall, Cambridge University Press, 2006.
[11] M. Lindskog, D. Dee, Y. Trémolet, E. Andersson, G. Radnóti, and M. Fisher, *A weak-constraint four-dimensional variational analysis system in the stratosphere*, Quarterly Journal of the Royal Meteorological Society, 135 (2009), pp. 695–706.
[12] MJ Martin, MJ Bell, and NK Nichols, *Estimation of systematic error in an equatorial ocean model using data assimilation*, International Journal for Numerical Methods in Fluids, 40 (2002), pp. 435–444.
[13] I.M. Navon, *Practical and theoretical aspects of adjoint parameter estimation and identifiability in meteorology and oceanography*, Dynamics of Atmospheres and Oceans. Special issue in honor of Richard Pfeffer, 27 (1998), pp. 55–79.
[14] J. Nocedal and S.J. Wright, *Numerical optimization*, Springer Verlag, 1999.
[15] D. Orrell, L. Smith, J. Barkmeijer, and T.N. Palmer, *Model error in weather forecasting*, Nonlinear Processes in Geophysics, 8 (2001), pp. 357–371.
[16] J.M. Ortega and W.C. Rheinboldt, *Iterative solution of nonlinear equations in several variables*, (2000).
[17] T.N. Palmer, G.J. Shutts, R. Hagedorn, F.J. Doblas-Reyes, T. Jung, and M. Leutbecher, *Representing model uncertainty in weather and climate prediction*, Annu. Rev. Earth Planet. Sci, 33 (2005), pp. 163–93.
[18] G.W. Platzman, *An exact integral of complete spectral equations for unsteady one-dimensional flow*, Tellus, 16 (1964), pp. 422–431.
[19] F. Rabier, H. Jarvinen, E. Klinker, J.F. Mahfouf, and A. Simmons, *The ECMWF operational implementation of four-dimensional variational assimilation, I: Experimental results with simplified physics*, Quarterly Journal of the Royal Meteorological Society, 126 (2000), pp. 1148–1170.
[20] L.R. Rabiner, *A tutorial on hidden Markov models and selected applications in speech recognition*, Proceedings of the IEEE, 77 (1989), pp. 257–286.
[21] L. Rabiner and B. Juang, *An introduction to hidden Markov models*, IEEE ASSp Magazine, 3 (1986), pp. 4–16.
[22] G.W. Stewart, *Matrix Algorithms: Eigensystems*, vol. 2, Society for Industrial Mathematics, 2001.
[23] Y. Trémolet, *Accounting for an imperfect model in 4D-Var*, Quarterly Journal of the Royal Meteorological Society, 132 (2006), pp. 2483–2504.
[24] ———, *Model-error estimation in 4D-Var*, Quarterly Journal of the Royal Meteorological Society, 133 (2007), pp. 1267–1280.
[25] B.A. van de Rotten and S.M.V. Lunel, *A limited memory Broyden method to solve high-dimensional systems of nonlinear equations*, [sn][Enschede]: PrintPartners Ipskamp (m), [Sl], 2003.
[26] M. Zupanski, D. Zupanski, T. Vukicevic, K. Eis, and T.V. Haar, *CIRA/CSU four-dimensional variational data assimilation system*, Monthly Weather Review, 133 (2005), p. 829.