An Analysis of 4D Variational Data Assimilation and Its Application

L. Jiang, Minneapolis, MN and **C.C. Douglas**, Laramie, WY

Abstract

In this paper, we give an analysis and a general procedure for 4D variational data assimilation (4D-Var). In functional partial differential equation setting, the adjoint equation method, sensitivity analysis, and multicomponent operator splitting are discussed. Nonlinear optimization methods and convergence analysis are also investigated for 4D-Var.

AMS Subject Classification: 65J15, 65M32, 49M05
Key words: 4D-Var, operator splitting methods, optimal control problems, dynamic data-driven application systems (DDDAS)

1 Introduction

There is a steadily growing interest in variational data assimilation [1, 18, 22]. Data assimilation systems use the sources of data: observations, a recent forecast (or "background") true at a known time, and error statistics. In many real life problems, observation sets are distributed in three dimensional space plus time, corresponding to four dimensional variational data assimilation (4D-Var). 4D-Var is a method of estimating a set of parameters by optimizing the fit between the solution of a model and a set of observations. The unknown model parameters may be the model's initial conditions or boundary conditions. The problem of determining the model parameters is very important and complex and has become a science in itself. The goal of 4D-Var is to incorporate actual observations (e.g., satellite, radar, ship, land surface, and so on) into mathematical and numerical models in order to build the best approximation (in some sense) of the true state and to create a unified model description of some substance (e.g., atmosphere or chemical pollutant) arising from nature. This can be used by scientific communities to study important phenomena associated with the substance. 4D-Var is a process where a state forecast and observations are combined to produce a

best (optimal) estimate or an analysis of the state or parameters of the model (possibility included in an extended state) [18].

Typically, optimization of parameters in 4D-Var can be regarded as a class of inverse problems and the parameters are often constrained by evolution differential operators. 4D-Var can be reduced to a PDE-constrained optimization problem. We will use a parameter which represents the initial condition of a nonlinear evolution differential operator. Therefore the 4D-Var problem (inverse problem) can be posted as, "what initial condition will seed the model to best predict the known observations?" A practical implementation of the optimization process requires a fast and accurate evaluation of the gradient of a optimization functional which may be provided by an adjoint model. Generally the adjoint model relies on the existence of the forward model itself which is run many times in an iterative procedure for numerical implementation.

In this paper, we discuss some key issues in 4D-Var from a functional perspective and a numerical perspective. We give an analysis for 4D-Var and provide a general procedure for 4D-Var.

The paper is organized as follows. In Section 2, we introduce a functional setting for 4D-Var, where the observation operator and constrained differential operator are both nonlinear in vector spaces. An optimal control problem is derived and sensitivity analysis and multicomponent operator splitting are carried out for the functional 4D-Var. In Section 3, we investigate iterative descent algorithms and convergence analysis of Quasi-Newton iterative methods for 4D-Var. In Section 4, we apply the analysis and the general procedure to a 4D-Var chemical transport model.

2 4D Variational Data Assimilation in Functional Setting

2.1 A Framework for 4D-Var

In this section, we discuss a framework for 4D-Var. 4D-Var is a generalization of 3D-Var for observations that are distributed in time. We set the framework using a nonlinear functional defined in vector spaces.

Let V be vector space with inner product $(\cdot,\cdot)_V$ and $U = \{\phi : [t_0, T] \longrightarrow V\}$ be a time dependent vector space. Let $u^B \in V$ represent the background of the initial values (the initial guess in the assimilation) and $B : V \rightarrow V$ be an associated linear symmetric covariance operator of the estimated background error. Let H be an observation operator which depends on time

t (without specification, we always assume that $H = H(t)$ in the paper) and represents the process of interpolating the model solution u to available observations. Basically, H maps space U into a vector space U_o, whose dimension is often lower than the dimension of U. Let $u_{obs} \in U_o$ be the real observations depending on time t. The linear symmetric covariance operator $R^{-1} : U_o \to U_o$ accounts for observations and representativeness errors and it depends on time. In its general form a cost function can be defined as

$$J(u^0, u) = \frac{1}{2}(u^0 - u^B, B^{-1}(u^0 - u^B)) + \frac{1}{2}(Hu - u_{obs}, R^{-1}(Hu - u_{obs}))_{U_o}, \quad (2.1)$$

where $u \in U$, $u^0 \in V$ and

$$(\cdot, \cdot)_{U_0} = \int_{t_0}^{T} (\cdot, \cdot)(s) ds.$$

Here $(\cdot, \cdot)$ inside the integral represents the inner product of the space $U_o(s)$ (for fixed time s). In this paper, we will use $(\cdot, \cdot)$ to represent generic inner products to avoid using many notations. Different spaces may have different definitions for the $(\cdot, \cdot)$ (e.g., observations in U_o and the corresponding inner product $(\cdot, \cdot)$ means $(\cdot, \cdot)_{U_o}$), but they can be easily distinguished in the context. The cost functional measures the difference between the model output u and the observation u_{obs} and the deviation of the solution from the background state u^B.

Remark 2.1. *In many real life situations, the observation u_{obs} is not defined continuously in a time interval $[t_0, T]$, but is evaluated at a set of discrete time moments $\{t_k\}_{k=0}^N$. For these cases, the cost functional is redefined as*

$$\begin{aligned} J(u^0, u) &= \frac{1}{2}(u^0 - u^B, B^{-1}(u^0 - u^B)) \\ &+ \frac{1}{2}\sum_{k=0}^{N}(H_k u^k - u_{obs}^k, R_k^{-1}(H_k u^k - u_{obs}^k)). \end{aligned} \quad (2.2)$$

Remark 2.2. *If the observation u_{obs} is not defined continuously in domain Ω, but is evaluated at a set of discrete locations $\{x_l\}_{l=0}^L$, then the cost functional is redefined as*

$$J(u^0, u) = \frac{1}{2}(u^0 - u^B, B^{-1}(u^0 - u^B)) + \frac{1}{2}(Hu - u_{obs}, R^{-1}(Hu - u_{obs})\delta(x - x_l)), \quad (2.3)$$

where $\delta(\cdot)$ is the Dirac function.

Remark 2.3. *If the observation u_{obs} is not defined continuously in a time interval and domain, but is evaluated at a set of discrete time moments $\{t_k\}_{k=0}^N$ and of space locations $\{x_l\}_{l=0}^L$, then the cost functional is redefined as*

$$\begin{aligned} J(u^0, u) = &\frac{1}{2}(u^0 - u^B, B^{-1}(u^0 - u^B)) \\ &+ \frac{1}{2}\sum_{k=0}^{N}(H_k u^k - u_{obs}^k, R_k^{-1}(H_k u^k - u_{obs}^k)\delta(x - x_l)). \end{aligned} \tag{2.4}$$

Remark 2.4. *$J(u^0, u)$ is called a sensitivity functional in the adjoint optimization.*

Let $S(t)$ be a predefined forecast model (a nonlinear operator) from the initial time t_0 to time t. If the model solution u is the solution of such an evolution equation as in (2.14), then $S(t)$ is basically a semigroup. Generally, 4D variational assimilation is defined as the minimization problem,

$$\hat{u^0} = \arg\min\{J(u^0, u) : u(t) = S(t)u^0\},$$

where the model states $u(t)$ are subject to the forward model equation,

$$u(t) = S(t)u^0. \tag{2.5}$$

Let $\bar{S}(t)$ be the linearization of $S(t)$. Corresponding to (2.5), a tangent linear model is defined by

$$\delta u(t) = \bar{S}(t)\delta u^0. \tag{2.6}$$

Let $\bar{S}^*(t)$ be the adjoint of $\bar{S}(t)$, $\bar{H}^*(t)$ the adjoint of $\bar{H}(t)$, and $\bar{H}(t)$ a linearization of $H(t)$. Let ∇_{u^0} be the gradient operator defined on space V with respect to u^0 and $\nabla_{u^0}^2$ the second derivative operator with respect to u^0. Since $u(t) = S(t)u^0$, a direct calculation gives rise to

$$\nabla_{u^0} J = B^{-1}(u^0 - u^B) + \int_{t_0}^{T} \bar{S}^* \bar{H}^* R^{-1}(Hu - u_{obs})(s)ds \tag{2.7}$$

and

$$\nabla_{u^0}^2 J(u^0, u) = B^{-1} + \int_{t_0}^{T} \bar{S}^* \bar{H}^* R^{-1} \bar{H} \bar{S} ds. \tag{2.8}$$

For the numerical approximation of (2.1) we need to discretize time. Without loss of generality, we can discuss the cost functional defined in (2.2) instead of the time discretization of (2.1) or (2.3). This is because the cost functional is the same as in (2.2) in many real life problems and the time

discretization of (2.1) is similar to (2.2). For example, a time discretization approximation of (2.1) can be written for a time step length of $\tau = \frac{T-t_0}{N}$ as

$$J(u^0, u) \approx \frac{1}{2}(u^0 - u^B, B^{-1}(u^0 - u^B)) + \frac{1}{2}\sum_{k=1}^{N} \tau(H_k u^k - u_{obs}^k, R_k^{-1}(H_k u^k - u_{obs}^k)).$$

In this paper, we will use the cost functional in (2.1) to discuss the continuous 4D-Var and use the functional in (2.2) to discuss the numerical 4D-Var.

Hence, 4D-Var is just a nonlinear constrained optimization problem that is very difficult to solve in the general case. For numerical approximation, 4D-Var can be simplified with two hypothesis:

Hypothesis 1: *The forecast model can be expressed as the product of intermediate forecast steps. Let $S_{[t,t+\tau]}$ be the forecast step from t to $t+\tau$, i.e., $u(t+\tau) = S_{[t,t+\tau]}u(t)$. Consequently, for $t_k = t_0 + k\tau$,*

$$u(t_k) = S_{[t_{k-1},t_k]} \cdots S_{[t_1,t_2]} S_{[t_0,t_1]} u^0.$$

Remark 2.5. *Hypothesis 1 usually means that the forecast model is the integration of a numerical prediction model starting with u^0 as the initial condition. Hypothesis 1 is suitable for an operator splitting method for the PDE problem.*

Hypothesis 2: *HSu^0 has a first order Taylor expansion around u^B at any observation time, i.e.,*

$$HSu^0 = HSu^B + \bar{H}\bar{S}(u^0 - u^B),$$

where we recall that $\bar{H} = \bar{H}(t)$ is the linearization of the observation operator $H(t)$ and $\bar{S} = \bar{S}(t)$ is the tangent linear model of $S(t)$, i.e., the differential (perturbation) of $S(t)$.

Remark 2.6. *Hypothesis 2 is a tangent linear hypothesis and implies that*

$$\nabla_{u^0} HS(u^0) = \bar{H}\bar{S}.$$

Let $\bar{S}_{[t_{k-1},t_k]} = \nabla_u S_{[t_{k-1},t_k]} u|_{t=t_{k-1}}$ and $\bar{S}_{[t_0,t_k]} = \Pi_{i=1}^{k} \bar{S}_{[t_{i-1},t_i]}$. With *Hypothesis 1*, it follows that

$$\delta u(t_k) = \bar{S}_{[t_{k-1},t_k]} \delta u(t_{k-1}), \quad \delta u(t_k) = \bar{S}_{[t_0,t_k]} \delta u(t_0).$$

Let $\bar{H}_k = \nabla_u H u|_{t=t_k}$. It is easy to verify that

$$\nabla_{u^0}(H_k u^k - u_{obs}^k) = \bar{H}_k \bar{S}_{[t_0,t_k]}.$$

Since adjoint problems are solved from end time to initial time, we use $\bar{S}^*_{[t_k,t_0]}$ to represent the linearization adjoint model from t_k to t_0. Consequently, it follows that

$$\nabla_{u^0} J = B^{-1}(u^0 - u^B) + \sum_{k=0}^{N} \bar{S}^*_{[t_k,t_0]} \bar{H}^*_k R_k^{-1}(H_k u^k - u^k_{obs}) \tag{2.9}$$

and

$$\nabla^2_{u^0} J = B^{-1} + \sum_{k=0}^{N} \bar{S}^*_{[t_k,t_0]} \bar{H}^*_k R_k^{-1} \bar{H}_k \bar{S}_{[t_0,t_k]}. \tag{2.10}$$

By *Hypothesis 2*, it follows that

$$H_k S_{[t_0,t_k]}(u^0_g + \delta u^0) = H_k S_{[t_0,t_k]} u^0_g + \bar{H}_k \bar{S}_{[t_0,t_k]} \delta u^0 = H_k u^k_g + \bar{H}_k \bar{S}_{[t_0,t_k]} \delta u^0,$$

where u^0_g is a guess of u^0. Consequently, for $d^k = u^k_{obs} - H_k u^k_g$,

$$\begin{aligned} J(\delta u^0) = & \frac{1}{2}(\delta u^0 - (u^B - u^0_g), B^{-1}(\delta u^0 - (u^B - u^0_g))) \\ & + \frac{1}{2}\sum_{k=0}^{N}(\bar{H}_k \bar{S}_{[t_0,t_k]} \delta u^0 - d^k, R_k^{-1}(\bar{H}_k \bar{S}_{[t_0,t_k]} \delta u^0 - d^k)). \end{aligned} \tag{2.11}$$

Thus we obtain that $\hat{u^0} = u^0_g + \delta\hat{u}^0$, which is called the incremental 4D-Var formulation [9]. Rather than a complete minimization of the full nonlinear cost function (2.2), the incremental method is an approximation of the full cost function by a series of minimizations of the quadratic cost functions (2.11) subject to a linear model, where the tangent linear model is used.

Let $\partial_{u^{k-1}} J$ be the partial differential of $J(u^0, u)$ with respect to u^{k-1} for a fixed first argument u^0 and $u = \{u^0, \cdots, u^N\}$. For the cost functional defined in (2.2), a time-discrete adjoint problem is defined such that the solution w to the adjoint model can be explicitly described by

$$w^N = 0 \quad \text{and} \quad w^{k-1} = \bar{S}^*_{[t_k,t_{k-1}]} w^k + \partial_{u^{k-1}} J. \tag{2.12}$$

Then we obtain the gradient of J with respect to u^0 by the adjoint problem (2.12). Thus, we have the following lemma:

Lemma 2.1. *Let w be the solution of (2.12). Then*

$$\nabla_{u^0} J(u^0, u) = w^0 + B^{-1}(u^0 - u^B).$$

Proof. Since $\partial_{u^k} J = \bar{H}_k^* R_k^{-1}(H_k u^k - u_{obs}^k)$, it follows from (2.9) that

$$\begin{aligned}
\nabla_{u^0} J &= B^{-1}(u^0 - u^B) + \sum_{k=0}^{N} \bar{S}^*_{[t_k,t_0]} \partial_{u^k} J \\
&= B^{-1}(u^0 - u^B) + \sum_{k=0}^{N} \bar{S}^*_{[t_1,t_0]} \cdots \bar{S}^*_{[t_k,t_{k-1}]} \partial_{u^k} J \qquad (2.13) \\
&= B^{-1}(u^0 - u^B) + I\partial_{u^0} J \\
&+ \bar{S}^*_{[t_1,t_0]}(\partial_{u^1} J + \bar{S}^*_{[t_2,t_1]}(\partial_{u^2} J + \cdots + \bar{S}^*_{[t_{N-1},t_{N-2}]}(\partial_{u^{N-1}} J + \bar{S}^*_{[t_N,t_{N-1}]} \partial_{u^N} J))).
\end{aligned}$$

By the recurrence definition of w in (2.12), it follows that

$$\nabla_{u^0} J = w^0 + B^{-1}(u^0 - u^B).$$

□

Remark 2.7. *By Lemma 2.1, we use the solutions of the adjoint equation (2.12) to evaluate the gradient of the cost functional.*

2.2 Constraint Specified by An Evolution Operator

We now discuss the case when the constraint $u(t) = S(t)u_0$ is specified by a nonlinear evolution operator A defined on a vector space U. The variational data assimilation problem associated with A and J is essentially an optimal control problem that can be formulated by the following minimization problem: Find the solution $\phi \in U$ of

$$\left\{ \begin{array}{rcl} D_t\phi &=& A(\phi) \\ \phi(t = t_0) &=& u^0 \\ \hat{u^0} &=& \arg\inf_{u^0} J(u^0, \phi), \end{array} \right. \qquad (2.14)$$

where $J(u^0, \phi)$ is defined in (2.1) if we substitute $\phi(t)$ for $u(t)$ in (2.1). In the paper, we assume that the nonlinear evolution equation in (2.14) has a unique solution.

Remark 2.8. *Since u^0 is the initial condition, $\phi = \phi(u^0)$. Hence, it is meaningful that $\hat{u^0} = \arg\inf_{u^0} J(u^0, \phi)$ in (2.14).*

Theorem 2.2. *The optimal control problem (2.14) is equivalent to the optimal system*

$$\left\{ \begin{array}{rcl} D_t\phi &=& A(\phi) \\ \phi(t = t_0) &=& u^0 \\ -D_t\phi^* &=& (\nabla A(\phi))^*\phi^* - \bar{H}^* R^{-1}(H\phi - \phi_{obs}) \\ \phi^*(t = T) &=& 0 \\ \phi^*(t = t_0) &=& B^{-1}(u^0 - u^B), \end{array} \right. \qquad (2.15)$$

where $\phi \in U$ and $\phi^* \in U$.

Proof. Let $u_1 = u_0 + \epsilon u'$ for $\epsilon \in R$ and $u' \in V$. We consider the problem with initial condition u_1,

$$\begin{cases} D_t\phi_1 &= A(\phi_1) \\ \phi_1(t = t_0) &= u_1. \end{cases} \tag{2.16}$$

Let $\phi_0 = \frac{\phi - \phi_1}{\epsilon}$. Then $\phi_1 = \phi + \epsilon\phi_0$. Suppose that A is Gâteaux differentiable, then the Taylor expansion yields

$$A(\phi_1) = A(\phi) + \nabla A(\phi)(\epsilon\phi_0) + O(\epsilon). \tag{2.17}$$

Substituting (2.17) into (2.16) with just the first order term (neglecting the higher order term $O(\epsilon)$) and subtracting the equation in (2.14) from (2.16), we obtain an equation with corresponding solution ϕ_0,

$$\begin{cases} D_t\phi_0 &= (\nabla A(\phi))\phi_0 \\ \phi_0(t = t_0) &= u'. \end{cases} \tag{2.18}$$

Since ϕ and u represent the optimal control problem (2.14), $J(u^0 + \epsilon u')$ is minimal when $\epsilon = 0$. This implies that $\frac{dJ}{d\epsilon}|_{\epsilon=0} = 0$. By the definition of J in (2.1), we have

$$\begin{aligned} &\frac{dJ(u^0 + \epsilon u', \phi + \epsilon\phi_0)}{d\epsilon} = (u', B^{-1}(u^0 - u^B)) + \epsilon(u', u') \\ &+ (\bar{H}\phi_0, R^{-1}(H\phi - \phi_{obs})) + \epsilon(\bar{H}\phi_0, R^{-1}\bar{H}\phi_0). \end{aligned} \tag{2.19}$$

Since $\frac{dJ}{d\epsilon}|_{\epsilon=0} = 0$, we have

$$(u', B^{-1}(u^0 - u^B)) + (\bar{H}\phi_0, R^{-1}(H\phi - \phi_{obs})) = 0$$

or

$$(u', B^{-1}(u^0 - u^B)) + (\phi_0, \bar{H}^* R^{-1}(H\phi - \phi_{obs})) = 0. \tag{2.20}$$

Equation (2.20) contains ϕ_0 which is a solution to (2.18). Let us use the dual representation for $(\phi_0, \bar{H}^* R^{-1}(H\phi - \phi_{obs}))$, which is a linear functional on ϕ_0, through a solution of the adjoint problem of the perturbation problem (2.18) of the form

$$\begin{cases} -D_t\phi^* &= (\nabla A(\phi))^*\phi^* - \bar{H}^* R^{-1}(H\phi - \phi_{obs}) \\ \phi^*(t = T) &= 0, \end{cases} \tag{2.21}$$

where operator $(\nabla A(\phi))^*$ is the adjoint to the operator $\nabla A(\phi)$ and satisfies the Lagrange identity for each time t: $(\nabla A(\phi)\psi, \psi^*) = (\psi, (\nabla A(\phi))^*\psi^*)$ for any $\psi \in D(\nabla A(\phi))$ and $\psi \in D((\nabla A(\phi))^*)$.

Multiplying (2.18) by ϕ^* and (2.21) by ϕ_0 and subtracting them from each other, we get

$$(u', \phi^*(t = t_0)) = -(\phi_0, \bar{H}^* R^{-1}(H\phi - \phi_{obs})). \tag{2.22}$$

Then

$$(u', \phi^*(t = t_0)) = (u', B^{-1}(u^0 - u^B)).$$

From here we have $B^{-1}(u^0 - u^B) - \phi^*(t = t_0) = 0$ since u' is arbitrary. This completes the proof. □

Remark 2.9. *The fifth equation in (2.15) is the optimality condition.*

Remark 2.10. *Let $S^A(t)$ denote an operator semigroup with generator A. By the notation of a semigroup,*

$$\phi(t) = S^A(t)u^0.$$

Hereafter we will use similar notations to denote semigroups. If A is a maximal dissipative operator on V, the exponential formula of the nonlinear semigroup [2] is

$$S^A(t)u^0 = \lim_{n\to\infty}(1 - \frac{t}{n}A)^{-n}u^0,$$

where the limit is taken in the strong topology sense. Further,

$$\phi^*(t) = -\int_t^T S^{-(\nabla A(\phi))^*}(t - s)(\bar{H}^* R^{-1}(H\phi - \phi_{obs}))(s)ds, \tag{2.23}$$

where $S^{-(\nabla A(\phi))^*}t = e^{-(\nabla A(\phi))^* t}$ since the generator $-(\nabla A(\phi))^*$ is a bounded linear operator.

We can rewrite (2.15) as an operator system of equations. We introduce operators G, S and $\bar{G}^*$, where $G = D_t - A$ denotes the forward transport operator associated with the first equation in (2.15), S extends a spatial field at initial time onto space-time, $\bar{S}^*$ restricts a space-time field to a spatial field at initial time, and $\bar{G}^* = -D_t - (\nabla A(\phi))^*$ is the adjoint transport operator associated with the third equation in (2.15). With these definitions, we can rewrite (2.15) as

$$\begin{bmatrix} \bar{H}^* R^{-1} H & 0 & \bar{G}^* \\ 0 & B^{-1} & -\bar{S}^* \\ G & -S & 0 \end{bmatrix} \begin{bmatrix} \phi \\ u^0 \\ \phi^* \end{bmatrix} = \begin{bmatrix} \bar{H}^* R^{-1}\phi_{obs} \\ B^{-1}u^B \\ 0 \end{bmatrix} \tag{2.24}$$

Hence we obtain the following proposition:

Proposition 2.3. *Let* $\mathbf{H} = B^{-1} + \bar{S}^*(\bar{G}^*)^{-1}\bar{H}^*R^{-1}HG^{-1}S$ *and* $g = \bar{S}^*(\bar{G}^*)^{-1}\bar{H}^*R^{-1}\phi_{obs}$, *where* G, $\bar{G}^*$, S, *and* $\bar{S}^*$ *are defined in (2.24). Then the optimal solution of (2.14) is the solution of the equation*

$$\mathbf{H}u^0 = g. \tag{2.25}$$

Proof. The optimal control problem (2.14) is equivalent to the system (2.24). The third equation in (2.24) implies that

$$\phi = G^{-1}Su^0.$$

Substituting $\phi = G^{-1}Su^0$ into the first equation in (2.24) to eliminate ϕ gives us

$$\phi^* = (\bar{G}^*)^{-1}\bar{H}^*R^{-1}(\phi_{obs} - HG^{-1}Su^0).$$

After eliminating ϕ^* in the second equation of (2.24), it follows immediately that $\mathbf{H}u^0 = g$. □

Remark 2.11. *The system (2.25) is the Schur complement system and* $\mathbf{H}$ *is the reduced Hessian operator and maps* V *to* U. *The discrete form of (2.24) can be computed by conjugate gradient methods. Given an initial guess* v, $w = \mathbf{H}v$ *can be computed through a matrix-free fashion [1].*

Remark 2.12. *Proposition 2.3 provides an idea of how to solve (2.14). Suppose* $\{E_\nu\}$ *is a sequence of invertible operators,* $\{a_\nu\}$ *is a sequence of real numbers, and* ν *is the index of* ν*-th iteration. We define an iterative method that is applied to* $\mathbf{H}u^0 = g$ *by*

$$E_\nu \frac{u^{0,(\nu+1)} - u^{0,(\nu)}}{a_\nu} = -(\mathbf{H}u^{0,(\nu)} - g). \tag{2.26}$$

For $L_\nu = I - a_\nu E_\nu^{-1}\mathbf{H}$ *and* $g_\nu = a_\nu E_\nu^{-1}g$, *(2.26) can be rewritten as*

$$u^{0,(\nu+1)} = L_\nu u^{0,(\nu)} + g_\nu.$$

When $\|L_\nu\| < 1$, *the scheme in (2.26) converges. By Proposition 2.3, the iterative process in (2.26) is equivalent to the following:*

$$\begin{cases} D_t\phi^{(\nu)} &= A(\phi^{(\nu)}) \\ \phi^{(\nu)}(t = t_0) &= u^{0,\nu} \\ -D_t\phi^{*,(\nu)} &= (\nabla A(\phi^{(\nu)}))^*\phi^{*,(\nu)} - \bar{H}^*R^{-1}(H\phi^{(\nu)} - \phi_{obs}) \\ \phi^{*,(\nu)}(t = T) &= 0 \\ E_\nu \frac{u^{0,(\nu+1)} - u^{0,(\nu)}}{a_\nu} &= \phi^{*,(\nu)}(t = t_0) - B^{-1}(u^{0,(\nu)} - u^B). \end{cases} \tag{2.27}$$

It is not easy to find both E_ν *and* a_v *in practice. So it is used mainly in theoretical analysis. In the next section, we introduce iterative methods in the setting of optimization, which are often used in numerical experiments.*

Next we discuss the sensitivity analysis in a functional analysis framework. We still consider the nonlinear evolution equation

$$\begin{cases} D_t\phi &= A(\phi) \\ \phi(t=t_0) &= u^0 \end{cases} \tag{2.28}$$

and a nonlinear functional $J(u^0, \phi)$. We want to know how sensitive the functional J is to the perturbation $\delta\phi$. Let $\delta_\phi J$ be the perturbation of $J(u^0, \phi)$ when $\delta\phi$ is nonzero.

Proposition 2.4. *Let J and u^0 be defined in (2.14). Then*

$$\delta_\phi J = -(\delta u^0, \phi^*(t=t_0)). \tag{2.29}$$

Proof. By (2.18), we have

$$\begin{cases} D_t\delta\phi - (\nabla A(\phi))\delta\phi &= 0 \\ \phi_0(t=t_0) &= \delta u^0. \end{cases} \tag{2.30}$$

By (2.21), we have

$$\begin{cases} -D_t\phi^* - (\nabla A(\phi))^*\phi^* &= -\bar{H}^* R^{-1}(H\phi - \phi_{obs}) \\ \phi^*(t=T) &= 0. \end{cases} \tag{2.31}$$

The proof of equation (2.22) implies that

$$(\delta u^0, \phi^*(t=0)) = -(\delta\phi, \bar{H}^* R^{-1}(H\phi - \phi_{obs})).$$

Since

$$\delta_\phi J = \int_{t_0}^{T} (\nabla_\phi J)\delta\phi(s)ds = (\delta\phi, \bar{H}^* R^{-1}(H\phi - \phi_{obs})),$$

we have

$$\delta_\phi J = -(\delta u^0, \phi^*(t=t_0)).$$

□

Equation (2.29) shows how the sensitivity of the functional $J(u^0, \phi)$ is related to initial condition u^0. From the above analysis, we find that the solution ϕ^* of the adjoint problem (2.21) is responsible for the sensitivity of the functional J to the initial condition. It is often called an influence function or importance function [16].

2.3 Operator Splitting Methods

In this subsection we show that *Hypothesis 1* can be realized using a constructive process and error estimates are given.

If $A(\phi)$ in (2.28) has a complicated structure and consists of different parts, e.g., $A(\phi)$ consists of an advection operator, a diffusion operator and a reaction operator, then splitting methods are advocated for solving (2.28) [12]. The basic idea behind operator splitting is to reduce a complicated problem into smaller or simpler subproblems such that different parts can be solved efficiently with appropriate integration formulas. Here we will use the Marchuk-Strang symmetrical multi-component splitting [15, 21] to solve (2.28).

Let $A(\phi) = \sum_{j=1}^{J} A_j(\phi)$, where $\mathcal{A}_j$ is the Lie operator associated with each operator A_j ($\mathcal{A}$ is associated with A). These Lie operators $\mathcal{A}_j$ (or $\mathcal{A}$) are linear operators on the space of operators acting on the solution space U of (2.28). For any $v \in U$ and any operator g on U, by definition of a Lie operator [12], it follows that

$$\mathcal{A}g(v) = g'(v)A(v).$$

So for the solution $\phi(t)$ of (2.28),

$$\mathcal{A}g(\phi(t)) = g'(\phi(t))A(\phi(t)) = \frac{\partial}{\partial t} g(\phi(t)).$$

Let I be identity operator. Using Lie-Taylor series [8] gives us

$$\phi(t+\tau) = (e^{\tau\mathcal{A}} I)\phi(t).$$

We split the problem (2.28) into J subproblems, $D_t\phi_j = A_j(\phi_j)$, $j = 1, \cdots, J$. We apply the Marchuk-Strang symmetrical multi-component splitting over time intervals $[t_k, t_{k+1}]$, where $t_{k+1} = t_k + \tau$ with constant time step length τ, to obtain

$$\left\{\begin{array}{rcl}
D_t\phi_1 &=& A_1(\phi_1), \quad \phi_1(t_k) = \phi_1'(t_k), \quad t \in [t_k, t_k + \frac{\tau}{2}] \\
&\cdots& \\
D_t\phi_{J-1} &=& A_{J-1}(\phi_{J-1}), \quad \phi_{J-1}(t_k) = \phi_{J-2}(t_k + \frac{\tau}{2}), \quad t \in [t_k, t_k + \frac{\tau}{2}] \\
D_t\phi_J &=& A_J(\phi_J), \quad \phi_J(t_k) = \phi_{J-1}(t_k + \frac{\tau}{2}), \quad t \in [t_k, t_k + \tau] \\
D_t\phi'_{J-1} &=& A_{J-1}(\phi'_{J-1}), \quad \phi'_{J-1}(t_k + \frac{\tau}{2}) = \phi_J(t_{k+1}), \quad t \in [t_k + \frac{\tau}{2}, t_{k+1}] \\
&\cdots& \\
D_t\phi'_1 &=& A_1(\phi'_1), \quad \phi_1(t_k + \frac{\tau}{2}) = \phi'_2(t_{k+1}), \quad t \in [t_k + \frac{\tau}{2}, t_{k+1}].
\end{array}\right. \tag{2.32}$$

Let $\mathcal{S}_{j,\frac{1}{2}\tau} = e^{\frac{1}{2}\tau\mathcal{A}_j}$, $\mathcal{S}_{J,\tau} = e^{\tau\mathcal{A}_J}$, and ϕ_k be approximations of $\phi(t_k)$, $k = 1, \cdots, N$. Applying the Baker-Campbell-Hausdorf formula of a Lie operator

[12] gives us

$$\phi(t_{k+1}) = \mathcal{S}_{1,\frac{1}{2}\tau} \cdots \mathcal{S}_{J-1,\frac{1}{2}\tau} \mathcal{S}_{J,\tau} \mathcal{S}_{J-1,\frac{1}{2}\tau} \cdots \mathcal{S}_{1,\frac{1}{2}\tau} I\phi(t_k) + O(\tau^3) \tag{2.33}$$

and

$$\phi_{k+1} = \mathcal{S}_{1,\frac{1}{2}\tau} \cdots \mathcal{S}_{J-1,\frac{1}{2}\tau} \mathcal{S}_{J,\tau} \mathcal{S}_{J-1,\frac{1}{2}\tau} \cdots \mathcal{S}_{1,\frac{1}{2}\tau} I\phi_k. \tag{2.34}$$

Let $\mathcal{S}_{[t_k,t_{k+1}]} = \mathcal{S}_{1,\frac{1}{2}\tau} \cdots \mathcal{S}_{J-1,\frac{1}{2}\tau} \mathcal{S}_{J,\tau} \mathcal{S}_{J-1,\frac{1}{2}\tau} \cdots \mathcal{S}_{1,\frac{1}{2}\tau}$ be the operator splitting procedure over $[t_k, t_{k+1}]$. Then

$$\phi_N = \Pi_{k=0}^{N-1} \mathcal{S}_{[t_k,t_{k+1}]} I\phi_0. \tag{2.35}$$

Remark 2.13. *By means of the Lie operator formalism, we transform a nonlinear splitting into the compositions of linear operators and so Hypothesis 1 is realized.*

Remark 2.14. *The term $O(\tau^3)$ in (2.33) represents the leading term of the local splitting error. It is a second order splitting scheme in the time because $\tau^{-1}\|\phi(t_{k+1}) - \phi(t_k)\| = O(\tau^2)$. The symmetrical operator splitting scheme has second order consistency. When $A_j(\phi)$ and $A_l(\phi)$ commute each other, i.e., for any $j \neq l$, $A_j' A_l = A_j A_l'$, where A_j' is the derivative with regard to ϕ, then no splitting error occurs [12].*

Remark 2.15. *In fact, (2.35) gives rise to the formula*

$$\lim_{n\to\infty} [e^{\frac{t}{2n}\mathcal{A}_1} \cdots e^{\frac{t}{2n}\mathcal{A}_{J-1}} e^{\frac{t}{n}\mathcal{A}_J} e^{\frac{t}{2n}\mathcal{A}_{J-1}} \cdots e^{\frac{t}{2n}\mathcal{A}_1}]^n Iu^0 = e^{t\mathcal{A}} Iu^0.$$

This generalizes Strang's product formula [14].

We can also use operator splitting to solve the perturbation equation (2.30).

We split the problem (2.31) into J subproblems, $-D_t\phi_j^* = (\nabla A_j(\phi))^* \phi_j^*$, $j = 1, \cdots, J$. We apply the Marchuk-Strang symmetrical multi-component splitting over a time interval $[t_{k+1}, t_k]$ to get

$$\begin{cases} -D_t\phi_1^* &= (\nabla A_1(\phi))^* \phi_1^*, \quad \phi_1^*(t_{k+1}) = \phi_1^{*'}(t_{k+1}), \quad t \in [t_{k+1}, t_k + \frac{\tau}{2}] \\ & \cdots \\ -D_t\phi_{J-1}^* &= (\nabla A_{J-1}(\phi))^* \phi_{J-1}^*, \quad \phi_{J-1}^*(t_{k+1}) = \phi_{J-2}^*(t_{k+1} - \frac{\tau}{2}), \\ & t \in [t_{k+1}, t_k + \frac{\tau}{2}] \\ -D_t\phi_J^* &= (\nabla A_J(\phi))^* \phi_J^*, \quad \phi_J(t_{k+1})^* = \phi_{J-1}^*(t_{k+1} - \frac{\tau}{2}), \quad t \in [t_{k+1}, t_k] \\ -D_t\phi_{J-1}^{*'} &= (\nabla A_{J-1}(\phi))\phi_{J-1}^{*'}, \quad \phi_{J-1}^{*'}(t_k + \frac{\tau}{2}) = \phi_J^*(t_k), \quad t \in [t_k + \frac{\tau}{2}, t_k] \\ & \cdots \\ -D_t\phi_1^{*'} &= (\nabla A_1(\phi))^* \phi^{*'}, \quad \phi_1^{*'}(t_k + \frac{\tau}{2}) = \phi_2^{*'}(t_k), \quad t \in [t_k + \frac{\tau}{2}, t_k]. \end{cases} \tag{2.36}$$

Thus, we obtain

$$\bar{\mathcal{S}}^*_{j,\frac{1}{2}\tau} = e^{-\frac{1}{2}\tau(\nabla A_j(\phi))^*}, \quad (j = 1, \cdots, J-1), \quad \bar{\mathcal{S}}^*_{J,\tau} = e^{-\tau(\nabla A_J(\phi))^*} \tag{2.37}$$

and

$$\begin{aligned} \phi^*_k &= \bar{\mathcal{S}}^*_{[t_{k+1},t_k]}\phi^*_{k+1} + \partial_{\phi_k} J(u^0, \phi) \\ &= \bar{\mathcal{S}}^*_{1,-\frac{1}{2}\tau} \cdots \bar{\mathcal{S}}^*_{J-1,-\frac{1}{2}\tau} \bar{\mathcal{S}}^*_{J,-\tau} \bar{\mathcal{S}}^*_{J-1,-\frac{1}{2}\tau} \cdots \bar{\mathcal{S}}^*_{1,-\frac{1}{2}\tau} \phi^*_{k+1} + \partial_{\phi_k} J(u^0, \phi). \end{aligned} \tag{2.38}$$

Let $\nabla^\tau_{u^0} J(u^0, \phi(u^0))$ be the approximation of $\nabla_{u^0} J(u^0, \phi(u^0))$ by the operator splitting method. Then we have the following theorem:

Theorem 2.5. *Let ϕ^* be defined in (2.38), the Marchuk-Strang operator splitting gives rise to*

$$\nabla^\tau_{u^0} J(u^0, \phi(u^0)) = \phi^*_0 + B^{-1}(u^0 - u^B),$$

where ϕ^*_0 is the operator splitting solution at t_0 defined in (2.38).

The proof is similar to the one for Lemma 2.1.

From Theorem 2.5 and the operator splitting procedure, we compute $J(u^0)$ by solving an adjoint problem that depends on the forward trajectory.

As for stability of the operator splitting scheme, if we have

$$\|e^{\frac{1}{2}\tau\mathcal{A}_j}\| \le e^{\frac{1}{2}\tau\omega_j}, \quad j = 1, \cdots, j-1, \quad \text{and} \quad \|e^{\tau\mathcal{A}_J}\| \le e^{\tau\omega_J}, \tag{2.39}$$

then $\|\phi_{k+1}\| \le e^{\tau\omega}\|\phi_k\|$, where $\omega = \sum_{j=1}^J \omega_j$. Hence, the stability holds on any finite time interval $[t_0, T]$ if $\omega > 0$. It also holds for an arbitrary large time interval if $\omega \le 0$. In practice the operator splitting scheme is stable if each sub-step is stable. By Lax's equivalence theorem [14], consistency and stability together imply convergence, and higher order consistency yields faster convergence. In particular, we have the following theorem for the global splitting error:

Theorem 2.6. *Let $\mu(\nabla A(\phi)) := \lim_{h\to 0+} \frac{\|I+h\nabla A(\phi)\|-1}{h} \le \lambda$, then*

$$\|\phi(t_n) - \phi_n\| \le C\tau^3(e^{n\lambda\tau} - 1)(e^{\lambda\tau} - 1)^{-1},$$

where C is a positive number independent of τ.

Proof. If the perturbation of the initial condition of (2.28) is δu^0, then by the perturbation equation (2.30) and using a semigroup expression, we have

$$\delta\phi(t) = e^{t\nabla A(\phi)}\delta u^0.$$

Consequently,

$$\|\delta\phi(t)\| \le \|e^{t\nabla A(\phi)}\|\|\delta u^0\| \le e^{\mu(\nabla A(\phi))}\|\delta u^0\| \le e^{\lambda t}\|\delta u^0\|, \tag{2.40}$$

where we have used Proposition 2.1 in [20] in the second step. By (2.33), the local splitting errors do not exceed $C\tau^3$ for some constant C. In computing ϕ_2 there is an error of $C\tau^3$ in the initial condition, and by (2.40), the effect of this error at t_2 is $C\tau^3 e^{\lambda\tau}$. Thus, the global splitting error at t_2 is $C\tau^3 + C\tau^3 e^{\lambda\tau}$. Similarly the global splitting error at t_3 is

$$C\tau^3 + (C\tau^3 + C\tau^3 e^{\lambda\tau})e^{\lambda\tau}.$$

Repeating the procedure in the same way we find that the global splitting error at t_n is

$$\sum_{k=1}^{n} C\tau^3 e^{(n-k)\lambda\tau} = C\tau^3 \sum_{k=0}^{n-1} e^{k\lambda\tau} = C\tau^3 (e^{n\lambda\tau} - 1)(e^{\lambda\tau} - 1)^{-1}.$$

□

Remark 2.16. *The $\mu(\nabla A(\phi))$ defined in Theorem 2.6 is called a logarithmic norm [20] of the bounded linear operator $\nabla A(\phi)$.*

Remark 2.17. *Since $e^z - 1 = O(z)$, we find that $(e^{n\lambda\tau}-1)(e^{\lambda\tau}-1)^{-1} = O(\frac{1}{\tau})$, and so $\|\phi(t_n) - \phi_n\| = O(\tau^2)$. Hence, the global Marchuk-Strang splitting error is second order for a time step of length τ.*

By utilizing Theorem 2.6 and Remark 2.17, the following proposition follows immediately.

Proposition 2.7. *Let $\nabla^{\tau}_{u^0} J(u^0, \phi(u^0))$ be the approximation of $\nabla_{u^0} J(u^0, \phi(u^0))$ by the Marchuk-Strang operator splitting method. If $\mu(\nabla A(\phi))$ and $\mu((\nabla A(\phi))^*)$ are bounded, then*

$$\|\nabla_{u^0} J(u^0, \phi(u^0)) - \nabla^{\tau}_{u^0} J(u^0, \phi(u^0))\| \le C\tau^2,$$

where C is a positive number independent of τ.

3 Iterative Descent Algorithm

When solving the optimal control problem (2.14), gradient based iterative methods are often utilized. Since the solution ϕ of the evolution equation in

(2.14) depends on u^0, we can rewrite $J(u^0) = J(u^0, \phi(u^0))$ and consider the minimizer of

$$\min\{J(u^0)|D_t\phi = A(\phi), \phi(t=0) = u^0\}.$$

We still define $\nabla J(u^0) = \nabla_{u^0} J(u^0)$ and use an iterative procedure of the form

$$u^{0,(\nu+1)} = u^{0,(\nu)} + a_\nu d_\nu$$

to obtain the convergence $u^{0,(\nu)} \longrightarrow \hat{u^0}$, where a_ν is the step length parameter and $d_\nu = d(u^{0,(\nu)})$ is the search direction. Generally d_ν can be written as

$$d_\nu = -E_\nu \nabla J(u^{0,(\nu)})$$

for some suitable positive operator E_ν. At each step ν, we choose

$$a_\nu = \arg\min_{a\in\mathbb{R}} J(u^{0,(\nu)} + a d_\nu), \tag{3.41}$$

which is a one dimensional optimization problem in $\mathbb{R}$. We use one dimensional searches (e.g., Fibonacci, golden section, polynomial interpolation, etc.) to find a_ν. Different choices of d_ν lead to different iterative methods. The three major iterative methods are listed as follows:

- *Gradient and conjugate gradient type iterative methods.* These methods use the first derivative ∇J to determine the search direction. For example, $d_\nu = -\nabla J(u^{0,(\nu)})$ gives rise to the Steepest Descent Method. d_ν generated as $\nabla^2 J$-conjugate directions leads to the Conjugate Gradient Method (or Fletcher Reeves Method) [10].

- *Newton type iterative methods.* These methods use the first derivative ∇J and second derivative $\nabla^2 J$ (Hessian of J) to determine the search direction. For example, $d_\nu = -(\nabla^2 J(u^{0,(\nu)}))^{-1}\nabla J(u^{0,(\nu)})$ leads to Newton's method.

- *Quasi-Newton type methods.* These methods use the first derivative ∇J and an approximating operator (or matrix) $\mathbb{H}^{-1}$ to the Hessian inverse $(\nabla^2 J(u^{0,(\nu)}))^{-1}$ to generate search direction, which is of the form $d_\nu = -\mathbb{H}_\nu^{-1}\nabla J(u^{0,(\nu)})$. For example, the BFGS (Broyden-Fletcher-Goldfarb-Shanno) method and the DFP (Davidon-Fletcher-Powell) method [17].

By either Lemma 2.1 or Theorem 2.5, $\nabla J(u^{0,(\nu)}) = \phi_0^*(u^{0,(\nu)}) + B^{-1}(u^{0,(\nu)} - u^B)$, where

$$\phi_0^*(u^{0,(\nu)}) = -\int_{t_0}^{T} e^{s(\nabla A(\phi))^*}(\bar{H}^* R^{-1}(He^{s\mathcal{A}} I u^{0,(\nu)} - \phi_{obs}))ds, \tag{3.42}$$

which is a direct result of (2.23). As for the step length defined in (3.41), we have the following proposition.

Proposition 3.1. *Let a_ν be defined as in (3.41). Then a_ν satisfies*

$$-(\phi_0^*(u^{0,(\nu)} + a_\nu d_\nu), d_v) - a_\nu(B^{-1}d_\nu, d_\nu) = B^{-1}(u^{0,(\nu)} - u^B, d_\nu), \qquad (3.43)$$

where $\phi_0^*(u^{0,(\nu)} + a_\nu d_\nu)$ is defined in (3.42), but evaluated at $(u^{0,(\nu)} + a_\nu d_\nu)$. In particular, if operator A is bounded and linear and H is linear, then

$$a_\nu = \frac{(B^{-1}(u^{0,(\nu)} - u^B), d_\nu) - (\int_{t_0}^T e^{sA^*}H^*R^{-1}(He^{sA}u^{0,(\nu)} - \phi_{obs})ds, d_\nu)}{(\int_{t_0}^T e^{sA^*}H^*R^{-1}He^{sA}dsd_\nu, d_\nu) - (B^{-1}d_\nu, d_\nu)}. \qquad (3.44)$$

Proof. Let $\ell(a) = J(u^{0,(\nu)} + ad_\nu)$. Since a_ν satisfies equation (3.41), $\ell'(a) = 0$. Theorem 2.5 and straightforward calculation lead to (3.43). If A is bounded and linear and H is linear, then $\nabla A(\phi) = A$ and $\bar{H} = H$. Consequently, (3.43) implies that

$$(\int_{t_0}^T (e^{sA})^* H^* R^{-1}(He^{sA}u^{0,(\nu)} - \phi_{obs})ds, d_\nu) + a_\nu(\int_{t_0}^T (e^{sA})^* H^* R^{-1} He^{sA}dsd_\nu, d_\nu)$$
$$-a_\nu(B^{-1}d_\nu, d_\nu) = (B^{-1}(u^{0,(\nu)} - u^B), d_\nu),$$

which is equivalent to (3.44). □

Convergence of gradient and conjugate gradient type iterative methods is slow in many cases. Newton type methods need to use the Hessian of J, which is very expensive computationally. In the following, we discuss Quasi-Newton methods, which are often used in 4D-Var, particularly when u^0 is in a high dimensional space for a real life problem.

For the Quasi-Newton method, we introduce the following notations:

$$\Delta u^{0,(\nu)} = u^{0,(\nu+1)} - u^{0,(\nu)}, \quad \Delta g_\nu = \nabla J(u^{0,(\nu+1)}) - \nabla J(u^{0,(\nu)}).$$

Let a sequence $\{\mathbb{H}_\nu\}$ have the secant property: at each stage ν, $\mathbb{H}_{\nu+1}$ satisfies the equation

$$\mathbb{H}_{\nu+1}^{-1}\Delta g^{(i)} = \Delta u^{0,(i)}, \quad 0 \le i \le \nu.$$

Quasi-Newton Iterative Algorithm
Step 1. Set $\nu = 0$, $\mathbb{H}_0 = I$ and initialize $u^{0,(0)}$.
Step 2. Set the search direction $d_\nu = -\mathbb{H}_\nu^{-1}g_\nu$, where $g_\nu = \nabla J(u^{0,(\nu)})$.
Step 3. Solve $a_\nu = Arg\min_{a\in R} J(u^{0,(\nu)} + ad_\nu)$ and set $u^{0,(\nu+1)} = u^{0,(\nu)} + a_\nu d_\nu$.
Step 4. Set $\nu = \nu + 1$ and go to *Step 2.*

Remark 3.1. *The proof of Proposition 3.1 implies*

$$(g_{\nu+1}, d_\nu) = 0. \qquad (3.45)$$

Let $Q := \nabla^2 J(u^0) = B^{-1} + \int_{t_0}^{T} \bar{S}^* \bar{H}^* R^{-1} \bar{H} \bar{S} ds$. If H and A are linear, then (2.7) and (2.8) imply that

$$Q\Delta u^{0,(\nu)} = \Delta g_\nu. \tag{3.46}$$

If H and A are linear and in a numerical setting, then they are matrices. The finite dimensional and linear property of A also implies that S is linear and finite dimensional. Hence, Q is linear and finite dimensional (i.e., can be represented as a matrix). For this case, we have the following theorem.

Theorem 3.2. *If H and A are linear and further Q is symmetric, positive and definite, then the Quasi-Newton Iterative method has the following properties:*
(1). $\{d_\nu\}$ are Q-conjugate.
(2). $(d_i, g_\nu) = 0$ for $i = 0, \cdots, \nu - 1$.
(3). $u^{0,(n)} = \hat{u^0}$ for $u^0 \in V$ if $\dim V = n$.
(4). $\mathbb{H}_n = Q$ for $u^0 \in V$ if $\dim V = n$.

Proof. (1). We prove (1) by induction. The result is true for $i = 0$. In fact,

$$\begin{aligned}
(Qd_0, d_1) &= -(Qd_0, \mathbb{H}_1^{-1} g_1) = -(Q(\frac{\Delta u^{0,(0)}}{a_0}), \mathbb{H}_1^{-1} g_1) \\
&= -(\frac{\Delta g_0}{a_0}, \mathbb{H}_1^{-1} g_1) = -(\frac{\mathbb{H}_1^{-1} \Delta g_0}{a_0}, g_1) \\
&= -(\frac{\Delta u^{0,(0)}}{a_0}, g_1) = -(d_0, g_1) \\
&= 0,
\end{aligned} \tag{3.47}$$

where we have used (3.45) and (3.46). Assume that it holds for $i \leq \nu$. We need to show it holds for $i = \nu + 1$. By straightforward computation, we have

$$(Qd_i, d_{\nu+1}) = -(d_i, g_{\nu+1}), \quad i = 0, 1, \cdots, \nu.$$

Since $u^{0,(\nu+1)} = u^{0,(i+1)} + \sum_{j=i+1}^{\nu} a_j d_j$, the linearity of H and A and (2.7)-(2.8) give rise to

$$g_{\nu+1} = g_{i+1} + \sum_{j=i+1}^{\nu} a_j Q d_j, \quad 0 \leq i \leq \nu - 1.$$

Consequently,

$$(g_{\nu+1}, d_i) = (g_{i+1}, d_i) + \sum_{j=i+1}^{\nu} a_j (Qd_j, d_i) = 0, \quad 0 \leq i \leq \nu - 1,$$

where we have used (3.45) and the assumption for the induction. It is trivial that $(g_{\nu+1}, d_\nu) = 0$. Hence it holds for $i = \nu + 1$ and the proof of (1) is complete.
(2). By the proof of (1), (2) follows immediately.
(3). By (1), $\{d_\nu\}_0^{n-1}$ are linearly independent. Hence $\nabla J(u^{0,(n)}) = 0$ if and only if is

$$(\nabla J(u^{0,(n)}), d_i) = 0, \quad i = 0, \cdots, n-1.$$

This is the same as (2).
(4). Since $\{\mathbb{H}_\nu\}$ has the secant property and (3.46) holds, it follows that

$$\mathbb{H}_{\nu+1}^{-1} Q(\Delta u^{0,(i)}) = \Delta u^{0,(i)}, \quad 0 \le i \le \nu,$$

which is equivalent to

$$\mathbb{H}_{\nu+1}^{-1} Q d_i = d_i, \quad 0 \le i \le \nu.$$

This implies (4) because $\{d_i\}_0^{n-1}$ forms a basis for domain of $\mathbb{H}_n$ and Q. □

One of the Quasi-Newton methods is BFGS [10, 17], where $\mathbb{H}_\nu$ are generated iteratively by

$$\mathbb{H}_{\nu+1} = \mathbb{H}_\nu + \frac{\Delta g_\nu \Delta g_\nu^T}{(\Delta u^{0,(\nu)}, \Delta g_\nu)} - \frac{(\mathbb{H}_\nu \Delta u^{0,(\nu)})(\mathbb{H}_\nu \Delta u^{0,(\nu)})^T}{(\mathbb{H}_\nu \Delta u^{0,(\nu)}, \Delta u^{0,(\nu)})}. \tag{3.48}$$

Generally BFGS will produce local q-superlinear convergence: there exists a positive sequence $\{c_\nu\}$ such that $c_\nu \longrightarrow 0$ and $\frac{\|u^{0,(\nu+1)} - \hat{u^0}\|}{\|u^{0,(\nu)} - \hat{u^0}\|} \le c_\nu$ for any $\nu \in N$. If we choose step size a_ν to be the Armijo step size [17], then

$$a_\nu = \max_{k \in \mathbb{N}} \{\beta^k | J(u^{0,(\nu)} + \beta^k d_\nu) - J(u^{0,(\nu)}) \le \alpha \beta^k (\nabla J(u^{0,(\nu)}), d_\nu)\},$$

where parameters $\alpha \in (0, \frac{1}{2})$ and $\beta \in (0, 1)$ are given. With the Armijo step size and (3.48), it produce the BFGS-Armijo method. There is another modified BFGS method advocated for 4D-Var [22], the L-BFGS (Limited-memory BFGS) method. In L-BFGS, the method is identical to the BFGS method in the first m (m is given) iterations . For $\nu > m$, $\mathbb{H}_\nu$ is obtained by applying m BFGS updates to $\mathbb{H}_0$ using information from the m previous iterations. L-BFGS is an extension of the BFGS-Armijo method. In L-BFGS, additional storage is used to accelerate convergence. It is suitable for large scale problems because the amount of storage required by the algorithms can be controlled by the user. For details of L-BFGS, see [11].

To describe the convergence rate of the modified BFGS, we need to make the following assumption for $J(u^0)$:

Assumption J. (1) The level set $D = \{v \in V : J(v) \leq J(u^{0,(0)})\}$ *is convex. (2) There exist two positive constants* M_1 *and* M_2 *such that* $M_1 \leq \lambda(Q) \leq M_2$*, where* $\lambda(Q)$ *denotes the eigenvalue of* Q *and* $Q = B^{-1} + \int_{t_0}^{T} \bar{S}^* \bar{H}^* R^{-1} \bar{H} \bar{S} ds$*.*

Remark 3.2. *Assumption J assures that J has a unique minimizer* $\hat{u^0}$ *in the convex set D. The first part of Assumption J implies a proper choice for initial guess* $u^{0,(0)}$ *and the second part implies that J is a convex functional and its conditional number is bounded.*

Utilizing the standard optimization techniques [10, 11, 17], we can obtain the following theorem:

Theorem 3.3. *(1) Let Q be positive definite. If there exists* $\delta > 0$ *such that*

$$\|u^{0,(0)} - \hat{u^0}\| \leq \delta \quad \textit{and} \quad \|\mathbb{H}_0 - Q\| \leq \delta,$$

then the $u^{0,(\nu)}$ *produced by BFGS converges q-superlinearly to* $\hat{u^0}$*.*
(2) Let Assumption J hold and $\mathbb{H}_0$ *is symmetric positive and definite. Then* $u^{0,(\nu)}$ *produced by BFGS-Armijo converges q-superlinearly to* $\hat{u^0}$ *globally.*
(3) Let Assumption J hold and $\mathbb{H}_0$ *is symmetric positive and definite. Then the* $u^{0,(\nu)}$ *produced by L-BFGS converges r-linearly to* $\hat{u^0}$ *globally: there exist* $\kappa > 0$ *and* $r \in (0, 1)$ *such that* $\|u^{0,(\nu)} - \hat{u^0}\| \leq \kappa r^{\nu}$*.*

Remark 3.3. *By Theorem 3.3, choosing proper initial data* $u^{0,(0)}$ *and* $\mathbb{H}_0$ *will play important roles for convergence. If we know the conditional number for Q is not far away from 1, then we choose* $\mathbb{H}_0 = I$*. If J is uniformly convex, i.e., Assumption J holds, then BFGS-Armijo and L-BFGS generate a global convergent sequence* $\{u^{0,(\nu)}\}$*.*

If $J(u^0)$ is defined as in (2.2), then we get a nonlinear least square problem

$$J(u^0) = \frac{1}{2}\|\mathbf{f}(u^0)\|^2 = \frac{1}{2}\mathbf{f}(u^0)^T\mathbf{f}(u^0),$$

where

$$\mathbf{f}(u^0) = \begin{pmatrix} B^{-\frac{1}{2}}(u^0 - u^B) \\ R_0^{-\frac{1}{2}}(H_0 u^0 - u_{obs}^0) \\ \vdots \\ R_N^{-\frac{1}{2}}(H_N S_{[t_0,t_N]} u^0 - u_{obs}^N) \end{pmatrix}.$$

This implies that the Gauss-Newton iteration method [10] can be applied to the 4D-Var problem. As for applying Gauss-Newton to the 4D-Var incremental formulation (2.11), see [13].

4 An Example of Nonlinear 4D-Var

In this section, we use a regional atmospheric chemical transport model [3] as an example. This model simulates the pollutants behavior by taking emissions, meteorology (wind, temperature, humidity, precipitation, etc.) and a set of chemical initial and boundary conditions as input. Let $c = \{c_1, \cdots, c_s\}$ denote a high dimension vector ($dim(c) = O(10^6)$) and represents the mole-fraction concentration of many chemical species. Let u denote the wind field vector, K the turbulent diffusivity tensor, ρ the air density, V^{dep} the deposition velocity of species, Q the rate of surface emissions, and E the rate of elevated emissions for this species. Let f be the rate of chemical transformations and it depends on absolute concentration values, the rate at which mole-fraction concentrations change is $\frac{f(\rho c)}{\rho}$. Let the domain Ω cover a region of the atmosphere and n be the outward normal vector on the boundary of Ω, defined by $\partial\Omega = \Gamma^{in} \bigcup \Gamma^{out} \bigcup \Gamma^{gr}$, where $\Gamma^{in} = \{x \in \partial\Omega | u(x) \cdot n \leq 0\}$ is the inflow part of $\partial\Omega$, $\Gamma^{out} = \{x \in \partial\Omega | u(x) \cdot n > 0\}$ is the outflow part of $\partial\Omega$ and Γ^{gr} is the ground level portion of $\partial\Omega$. To avoid introducing many more notations, we use c to denote the concentration of representative chemical species. The governing equation is defined by

$$\begin{cases} D_t c &= -u \cdot \nabla c + \frac{1}{\rho} div(\rho K \nabla c) + \frac{1}{\rho} f(\rho c) + E, \quad t^0 \leq t \leq T \\ c(t_0, x) &= c^0(x) \\ c(t, x) &= c^{in}(t, x) \ \text{ on } \ \Gamma^{in} \\ K \frac{\partial c}{\partial n} &= 0 \ \text{ on } \ \Gamma^{out} \\ K \frac{\partial c}{\partial n} &= V^{dep} c - Q \ \text{ on } \ \Gamma^{gr}. \end{cases} \tag{4.49}$$

By direct computation, we obtain the tangent linear model equation of (4.49) as

$$\begin{cases} D_t \delta c &= -u \cdot \nabla \delta c + \frac{1}{\rho} div(\rho K \nabla \delta c) + F(\rho c)\delta c, \quad t^0 \leq t \leq T \\ \delta c(t_0, x) &= \delta c^0(x) \\ \delta c(t, x) &= 0 \ \text{ on } \ \Gamma^{in} \\ K \frac{\partial \delta c}{\partial n} &= 0 \ \text{ on } \ \Gamma^{out} \\ K \frac{\partial \delta c}{\partial n} &= V^{dep} \delta c \ \text{ on } \ \Gamma^{gr}, \end{cases} \tag{4.50}$$

where F is the Jacobian of $\{\frac{\partial f_i}{\partial c_j}\}$.

Integration by parts implies the adjoint model is defined by

$$\begin{cases} \frac{\partial \lambda}{\partial t} = -[div(u\lambda) &+ div(\rho K \nabla \frac{\lambda}{\rho}) + F^T(\rho c)\lambda] - (\nabla_{c^k} J(c^0, c)), \quad t = t_k \\ \lambda(T, x) &= 0 \\ \lambda &= 0 \ \text{ on } \ \Gamma^{in} \\ (\lambda u + \rho K \nabla(\frac{\lambda}{\rho})) \cdot n &= 0 \ \text{ on } \ \Gamma^{out} \\ (\rho K \nabla(\frac{\lambda}{\rho})) \cdot n &= V^{dep} \lambda \ \text{ on } \ \Gamma^{gr}, \end{cases} \tag{4.51}$$

where J is defined as in (2.2) by

$$J(c^0) = \frac{1}{2}(c^0 - c^B)^T B^{-1}(c^0 - c^B) + \frac{1}{2}\sum_{k=0}^{N}(H_k c^k - c_{obs}^k)^T R_k^{-1}(H_k c^k - c_{obs}^k). \tag{4.52}$$

The adjoint model contains the term $F(\rho c)$. The adjoint problem (4.51) depends on the states of the forward model (4.49). The adjoint initial condition is posed at the final time T so that the forward model must first be solved forward in time, then the adjoint model is integrated backward from T to t_0.

For simplicity, we set the forward model differential nonlinear operator to be

$$A(c) = -u \cdot \nabla c + \frac{1}{\rho} div(\rho K \nabla c) + \frac{1}{\rho} f(\rho c) + E \tag{4.53}$$

and the adjoint model differential nonlinear operator as

$$(\nabla A(c))^* \lambda = div(u\lambda) + div(\rho K \nabla \frac{\lambda}{\rho}) + F^T(\rho c)\lambda. \tag{4.54}$$

By Theorem 2.2, the optimal control problem (2.14) with constraint defined in (4.51) reduces to

$$\begin{cases} D_t c &= A(c), \quad t^0 \le t \le T \\ c(t = t_0) &= c^0 \\ -D_t \lambda &= (\nabla A(c))^* \lambda - \bar{H}^* R_k^{-1}(H_k c^k - c_{obs}^k), \quad t = t_k \\ \lambda(t = T) &= 0 \\ \lambda(t = t_0) &= B^{-1}(c^0 - c^B), \end{cases} \tag{4.55}$$

where the boundary conditions of c and λ are described in (4.49) and (4.51), respectively.

For numerical computation, we can use a finite difference, i.e., Crank-Nicholson scheme, for time t and operator splitting methods for advection, diffusion, and reaction part:

$$D_t c_1 = A_1 c_1, \quad D_t c_2 = A_2 c_2, \quad \text{and} \quad D_t c_3 = A_3(c_3),$$

where $A_1 c_1 = -u \cdot \nabla c_1$ is the advection part (linear), $A_2 c_2 = \frac{1}{\rho} div(\rho K \nabla c_2)$ is the diffusion part (linear) and $A_3(c_3) = \frac{1}{\rho} f(\rho c_3) + E$ is the reaction part (nonlinear stiff term). The advantage of the operator splitting method is that different parts can be treated by different numerical schemes. As an example [19], a third order upwind finite difference scheme is applied to advection and a second order central difference scheme is applied to the horizontal and vertical diffusion. An s-stage Rosenbrock method is applied to chemical

reaction, which has proven successful for many different stiff ODE and PDE applications [8]. For convenience, we use a stiff ODE, $D_t c(t) = r(c(t))$, to describe the s-stage one step Rosenbrock method. The process is defined by

$$\begin{cases} c^{n+1} &= c^n + \sum_{i=1}^{s} b_i k_i \\ k_i &= \tau r(c^n + \sum_{j=1}^{i-1} \alpha_{ij} k_j) + \tau G_n \sum_{j=1}^{i} \gamma_{ij} k_j, \end{cases} \tag{4.56}$$

where $G_n = r'(c^n)$ (Jacobian matrix) and coefficients b_i, α_{ij} and γ_{ij} are chosen to obtain a desired order of consistency and stability. For a detailed introduction of the Rosenbrock method, see [8].

Repeating the Marchuk-Strang splitting method introduced in Section 2.3 to the forward model (4.49) over the time interval $[t_k, t_{k+1}]$, we have

$$\begin{cases} D_t c_1 &= A_1 c_1, \quad c_1(t_k) = c_1'(t_k) \\ D_t c_2 &= A_2 c_2, \quad b.c., \quad c_2(t_k) = c_1(t_k + \frac{\Delta t}{2}) \\ D_t c_3 &= A_3(c_3), \quad c_3(t_k) = c_2(t_k + \frac{\Delta t}{2}) \\ D_t c_2' &= A_2 c_2', \quad b.c., \quad c_2(t_k + \frac{\Delta t}{2}) = c_3(t_{k+1}) \\ D_t c_1' &= A_1 c_1', \quad c_1'(t_k + \frac{\Delta t}{2}) = c_2'(t_{k+1}). \end{cases} \tag{4.57}$$

We denote the above procedure by

$$c^{k+1} = S_{[t^k, t^{k+1}]} c^k,$$

where $S_{[t^k,t^{k+1}]} = S_{1,\frac{\Delta t}{2}} S_{2,\frac{\Delta t}{2}} S_{3,\Delta t} S_{2,\frac{\Delta t}{2}} S_{1,\frac{\Delta t}{2}}$, and the S_i are associated to the operators $D_t - A_i$, $i = 1, 2, 3$. Consequently,

$$c^N = \Pi_{k=0}^{N-1} S_{[t^k, t^{k+1}]} c^0.$$

Let $\bar{S}^*_{[t^{k+1},t_k]} = \bar{S}^*_{1,-\frac{\Delta t}{2}} \bar{S}^*_{2,-\frac{\Delta t}{2}} \bar{S}^*_{3,-\Delta t} \bar{S}^*_{2,-\frac{\Delta t}{2}} \bar{S}^*_{1,-\frac{\Delta t}{2}}$, and the $\bar{S}_i^*$ be associated with the operators $D_t + (\nabla A(c))_i^*$, $i = 1, 2, 3$, and $(\nabla A(c))_1^* \lambda = div(u\lambda)$, $(\nabla A(c))_2^* \lambda = div(\rho K \nabla \frac{\lambda}{\rho})$, and $(\nabla A(c))_3^* \lambda = F^T(\rho c)\lambda$. Similarly, the splitting method for adjoint problem (4.51) leads to

$$\lambda^k = \bar{S}^*_{[t^{k+1},t_k]} \lambda^{k+1} + \bar{H}_k^* R_k^{-1} (H c^k - c_{obs}^k).$$

Consequently, λ^0 can be computed by the following iterative algorithm.
Algorithm for λ^k
Step 1: Initial Lambda=0.
Step 2: for $k = N-1, 0, -1$ do Lambda=$\bar{S}^*_{[t^{k+1},t^k]}$(Lambda $+ \bar{H}_k^* R_k^{-1}(H c^k - c_{obs}^k)$).
step 3: λ^0 = Lambda.

From **Algorithm for** λ^k, we need to compute $\bar{S}^*_{[t^{k+1},t^k]}v$ from an arbitrary seed vector. Since $\bar{S}^*_1$ and $\bar{S}^*_2$ are linear, the products $\bar{S}^*_1 v$ and $\bar{S}^*_2 v$ can be efficiently computed by automatic adjoint compilers [4, 7]. However, $\bar{S}^*_3$ is highly nonlinear, and the computation of $\bar{S}^*_3$ needs special consideration. The performance of the adjoint model is dominated by the implementation of the direct and adjoint methods used in the chemistry integration which in practice takes as much as 90% of the CPU time [5]. The s-stage (i.e., $s = 2, 3$) Rosenbrock method is an efficient method to solve the chemical reaction part because it provides an efficient implementation for computing $\bar{S}^*_3 v$ [5, 6]. Once we have obtained λ^0, then we evaluate $\nabla J(c^0)$ by Theorem 2.5, which is important in solving the 4D-Var problem by an iterative optimization.

Remark 4.1. *From the operator scheme in (4.57), there exist two errors theoretically. One arises from the Marchuk-Strang operator splitting method itself, which always leads to a second order approximation in time unless advection, diffusion and reaction commute with each other. The second error arises from inconsistencies between the boundary conditions and the initial values prescribed in the intermediate steps, the second equation in (4.57) and the forth equation in (4.57). This may be reduced or removed by using the Rosebrock method with approximation factorization [12]. The same will happen to the splitting process of adjoint problem.*

Numerical results for the 4D-Var chemical transport model can be found in [5, 19, 22]. Authors in [22] have applied L-BFGS, nonlinear conjugate gradient and Hessian free Newton method to implement the chemical transport model. Their numerical results show that L-BFGS converges the fastest of these three methods.

5 Conclusions

In this paper, we presented a framework and an analysis for 4D variational data assimilation. The proposed analysis stems from the rapid theoretical advance of 4D-Var and the desire to translate it into real life applications.

First, we investigated a general framework of 4D-Var in the setting of functional operators. Constructing and solving an adjoint problem gives rise to an efficient approach to evaluate the gradient of the cost functional with respect to an unknown parameter. This framework helps us understand 4D-Var theoretically.

Second, we explored some numerical techniques: symmetrical operator splitting and Quasi-Newton iterative methods, which are used to implement

a 4D-Var problem in practice. It concluded that a large and complicated 4D-Var problem can be split into many subproblems and the Marchuk-Strang symmetrical multi-component splitting method gives rise to a second order splitting error for time step. Nonlinear optimization iterative methods (e.g., Quasi-Newton) lead to fast convergence of the numerical solution in 4D-Var nonlinear optimal problems.

Finally, we used a chemical transport example to demonstrate the procedure of 4D-Var. Rosenbrock method is an efficient method to solve the chemical reaction part when the Marchuk-Strang symmetrical splitting method is applied.

Acknowledgements

The research in this paper was supported in part by National Science Foundation grants EIA-0219627, ACI-0305466, ACI-0324876, CNS-0540136, and CNS-0720454. We would like to thank referees for insightful comments. The first author would like to thank Prof. Efendiev for valuable discussions at Texas A&M University.

References

[1] V. Akcelik, G. Biros, A. Draganescu, O. Ghattas, J. Hill, and B. v. BloemenWaanders, *Inversion of Airborne Contaminants in a Regional Model*, Computational Science - ICCS 2006: 6th International Conference, Reading, UK, May 28-31, 2006, Proceedings, Part III, Vassil N. Alexandrov, Geert Dick van Albada, Peter M.A. Sloot, Jack J. Dongarra (eds.), Lecture Notes in Computer Science series, vol. 3993, pp. 481-488, Springer-Verlag, Heidelberg, 2006.

[2] V. Barbu, *Nonlinear Semigroups and Differential Equations in Banach Spaces*, Editura Academiei, Bucuresti, 1976.

[3] G.R. Carmichael, L.K., Peters, and R.D. Saylor, *The STEM-II regional scale acid deposition and photochemical oxidant model - I. An overview of model development and applications*, Atmospheric Environment 25A (1990), pp. 2077-2090.

[4] F. T. Coleman, F. Santosa and A. Verma, *Effcient Calculation of Jacobian and Adjoint Vector Products in Wave Propagational*

Inverse Problems Using Automatic Differentiation, J. Comput. Phys., 157 (2000), pp. 234–255.

[5] D. Daescu, G. R. Carmichael and A. Sandu, *Adjoint implementation of Rosenbrock methods applied to variational data assimilation problems*, J. Comput. Phys., 165 (2000), pp. 496-510

[6] G. Dimitriu and R. Cuciureanu, *Mathematical Aspects of Data Assimilation for Atmospheric Chemistry Models*, Advances in Air Pollution Modeling for Environmental Security, Vol. 54 (2005), pp. 93-104.

[7] R. Giering, *Tangent linear and Adjoint Model Compiler*, http://www.autodiff.com/tamc, visit on Dec. 1, 2007.

[8] W. Hundsdorfer and J. G. Verwer, *Numerical Solutions of Time-Dependent Advection-Diffusion-Reaction Equations*, Springer-Verlag, New York, 2003.

[9] K. Ide, P. Courtier, M. Ghil, and A. C. Lorenc, *Unified Notation for Data Assimilation: Operational, Sequential and Variational*, JMSJ, vol. 75 (1B) (1997), pp. 181-189.

[10] M.C. Joshi and K. M. Moudgalya, *Optimization: Theory And Practice*, Alpha Science Intl Ltd, Harrow, 2004.

[11] D. Liu and J. Nocedal, *On the limited memory BFGS method for large scale optimization*, Math. Programming 45 (3) (1989), pp 503–528.

[12] D. Lanser and J. G. Verwer, *Analysis of operator splitting for advection-diffusion-reaction problems from air pollution modelling*, J. Comput. Appl. Math. 111 (1999), pp. 201–216.

[13] A. S. Lawless, S. Gratton, N. K. Nichols, *An Investigation of Incremental 4D-Var Using Non-tangent Linear Models*, Q. J. R. Meteorol. Soc., Vol. 131 (no.606) (2006) , Pp. 459– 476.

[14] P. Lax, *Functional Analysis*, Wiley Interscience, New York, 2002.

[15] G. Marchuk, *Adjoint Equations and Analysis of Complex System*, Mathematics and Its Applications Series 29, Springer-Verlag, New York, 1995.

[16] G. Marchuk, V. Shutyaev and G. Bocharov, *Adjoint Equations and Analysis of Complex System: Application to Virus Infection Modelling*, J. Computational and Applied Math, 184 (2005), pp. 177-204.

[17] E. Polak, *Optimization: Algorithms and Consistent Approximations*, Springer-Verlag, New York, 1997.

[18] F. A. Rihan and C. G. Collier, *Four-Dimensional Data Assimilation and Numerical Weather Prediction*, CERS Data Assimilation Technical Report No. 1 (2003), University of Salford, UK.

[19] A. Sandu, D.N., Daescu, G.R. Carmichael and T. Chai, *Adjoint Sensitivity Analysis of Regional Air Quality Models*, J. Comput. Phys., Vol. 204 (2005), pp. 222-252.

[20] G. Söderlind, *The logarithmic norm. History and modern theory*, Bit., Vol. 46 (no.3) (2006), pp. 631-652.

[21] G. Strang, *On the construction and comparison of difference schemes*, SIAM J. Numer. Anal. 5 (1968), pp. 506–517.

[22] L. Zhang and A. Sandu, *Data Assimilation in Multiscale Chemical Transport Models*, Computational Science - ICCS 2007: 7th International Conference, Beijing, China, May 27-30, 2007, Proceedings, Part 1, Y. Shi, Geert Dick van Albada, Jack J. Dongarra, and Peter M. A. Sloot (eds.), Lecture Notes in Computer Science series, vol. 4487, pp. 1026-1033, Springer-Verlag, Heidelberg, 2007.

Lijian Jiang
Institute for Mathematics and its Applications
University of Minnesota
Minneapolis, MN 55455-0134
U.S.A.
lijiang@ima.umn.edu

Craig C. Douglas
Departments of Mathematics, University of Wyoming
1000 E. University Ave.
Laramie, WY 82071
U.S.A.
cdougla6@uwyo.edu