Application of Variational Algorithms in Semi-Lagrangian Framework

Wang Yunfeng (王云峰), Wu Rongsheng (伍荣生), Wang Yuan (王 元), and Pan Yinong (潘益农)

Laboratory of Meso-scale Severe Weather, Department of Atmospheric Sciences, Nanjing University, Nanjing 210093

(Received September 4, 1998; revised November 9, 1998)

ABSTRACT

The variational data assimilation scheme (VAR) is applied to investigating the advective effect and the evolution of the control variables in time splitting semi-Lagrangian framework. Two variational algorithms are used. One is the conjugate code method-direct approach, and another is the numerical backward integration of analytical adjoint equation-indirect approach. Theoretical derivation and sensitivity tests are conducted in order to verify the consistency and inconsistency of the two algorithms under the semi-Lagrangian framework. On the other hand, the sensitivity of the perfect and imperfect initial condition is also tested in both direct and indirect approaches.

Our research has shown that the two algorithms are not only identical in theory, but also identical in numerical calculation. Furthermore, the algorithms of the indirect approach are much more feasible and efficient than that of the direct one when both are employed in the semi-Lagrangian framework. Taking advantage of semi-Lagrangian framework, one purpose of this paper is to illustrate when the variational assimilation algorithm is concerned in the computational method of the backward integration, the algorithm is extremely facilitated. Such simplicity in indirect approach should be meaningful for the VAR design in passive model. Indeed, if one can successfully split the diabatic and adiabatic process, the algorithms represented in this paper might be easily used in a more general vision of atmospheric model.

Key words: Variational algorithms, Semi-Lagrangian framework, Indirect approach, Direct approach

1. Introduction

Weather forecasting model is a set of partial differential equations (PDE). It should satisfy certain initial-boundary conditions. Clearly, one of the most important improvements in weather forecasting model is to optimize its initial and boundary conditions. For the simplicity of discussion, we might regard the spatial boundary condition as a kind of initial condition in spatial dimension.

An effective data assimilation procedure is required in order to utilize widely collective data for the weather forecasting model and find an optimal initial condition. The purpose of the comprehensive utilization is to make the errors minimal and to improve the forecasting quality good enough as possibly as it can.

Application of variational assimilation in atmospheric model was best demonstrated and developed by Lewis and Derber (1985), LeDimet and Talagrand (1986), and Talagrand and Courtier (1987). Recently, it has been a popular interesting topic in China (cf. Gao et al., 1995; Zhu et al., 1995; and the references therein).

However, few people had ever linked this technique be applied in semi–Lagrangian framework. Just in recent 10 years, meteorologists begin to rethink semi–Lagrangian framework because it allows the use of large time steps without introducing computational instability (see the review of Staniforth and Côte, 1991). Actually, the computational efficiency of the semi–Lagrangian framework is 6 times faster than that of traditional Eulerian framework (Robert, 1981, 1982). It is imaginable that semi–Lagrangian framework will become increasingly important in numerical weather forecasting models. The research trying to combine various VAR techniques with the semi–Lagrangian framework is therefore very meaningful.

To speak convincingly, the variation algorithm is put place in two categories according to their discretion fashion, namely, direct and indirect approach. The steps for direct approach (it is also called conjugate code method in the paper of Chen et al., 1998) include: the discretion of PDE and the corresponding numeric; then, writing out the numerical tangent linear model (TLM); finally, writing out the numerical adjoint programs (cf. Chao et al., 1992; and Rostating et al., 1993). For indirect approach, the steps are as follows: writing out the partial derivative adjoint equation indirectly from the linear version of PDE; then writing out the numerical adjoint programs (cf. Lewis and Derber, 1985, for example). It is in our interest to compare numerically the efficiency and accuracy of two approaches under semi–Lagrangian framework.

Taking advantage of semi–Lagrangian framework, one purpose of this paper is to illustrate when the variational assimilation algorithm concerns in the computational method of the backward integration, the algorithm is extremely faciliated under semi–Lagrangian framework. The backward integrating process is actually taken place along the semi–Lagrangian trajectories of air parcels; consequently, the numerical codes are almost identical between backward and forward semi–Lagrangian integration.

Since few papers have been dedicated these aspects, so we will provide sensitive tests in an idealized cone test. For advection–dominated airflow, such cone test provides a unique analytical solution so that mainly through the comparison between modeling results and analytical solutions, the statistic significance presents a rigorous benchmark of accuracy in various variational algorithms under semi–Lagrangian framework.

2. Principle for variatinal algorithm

The basic PDE equations in forecast model can be written in a compact form

$$\frac{d\mathbf{Y}}{dt} = \mathbf{F}(\mathbf{Y}), \tag{1}$$

where the vector $\mathbf{Y}(t)$ belongs to a Hilbert space Ω, and $\mathbf{F}(\mathbf{Y})$ and represents a regular function of $\Omega \rightarrow \Omega$ into itself. If the initial condition $\mathbf{Y}t_1$ is given, under certain boundary conditions, a time series of the forecast variables such as $\mathbf{Y}(t_1)$, $\mathbf{Y}(t_2)$, $\mathbf{Y}(t_3)$, $\mathbf{Y}(t_n)$ can be obtained from the solutions of Eq. (1). Suppose that the estimate $\hat{\mathbf{Y}}(t)$ of $\mathbf{Y}(t)$ is also available, i.e., $\hat{\mathbf{Y}}(t_1)$, $\hat{\mathbf{Y}}(t_2)$, $\hat{\mathbf{Y}}(t_3)$, $\hat{\mathbf{Y}}(t_n)$ at the corresponding time levels. A cost function J with respect to initial condition $\mathbf{Y}(t_1)$ is thus defined as:

$$J(\mathbf{Y}(t_1)) = \frac{1}{2}\sum_{i=1}^{n} < \mathbf{W}(t_i)(\mathbf{Y}(t_i) - \hat{\mathbf{Y}}(t_i)), \mathbf{Y}(t_i) - \hat{\mathbf{Y}}(t_i) > . \tag{2}$$

The notation " $<,>$" means the inner product in Ω space, and $\mathbf{W}(t_i)$ is weighting matrix.

Notice that we do not assume the estimate $\hat{\mathbf{Y}}(t_i)$ is exactly compatible with a solution of Eq. (1). The idea behind the variational algorithm is to get an optimal solution of Eq. (1) with respect to minimizing the cost function J. In other words, the initial condition has been adjusted in the optimal way to suit model integration. Such adjustment can be expressed as

$$\mathbf{Y}^{\nu+1}(t_1)=\mathbf{Y}^{\nu}(t_1)-\rho^{\nu}\nabla J^{\nu}(\mathbf{Y}(t_1)), \tag{3}$$

where ν is iteration number; ∇J is the gradient of J with respect to the initial condition $\mathbf{Y}(t_1)$ and ρ is convergent speed (or say, optimal step size); ρ in turn can be determined by quasi–Newton scheme (Gill et al., 1982).

2.1 *Indirect approach algorithm*

Given the initial condition $\mathbf{Y}(t_1)$ and for the corresponding solutions of Eq. (1), the first–order variation δJ resulting from a variation $\delta\mathbf{Y}(t_1)$ of $\mathbf{Y}(t_1)$ is equal to

$$\delta J(\mathbf{Y}(t_1))=\sum_{i=1}^{n}<\mathbf{W}(t_i)(\mathbf{Y}(t_i)-\hat{\mathbf{Y}}(t_i)),\delta\mathbf{Y}(t_i)>. \tag{4}$$

Here the first–order perturbations $\delta\mathbf{Y}(t_i)$ are themselves obtained by integration of linear perturbation of Eq. (1), starting from the initial condition $\delta\mathbf{Y}(t_1)$. That is:

$$\frac{d\delta\mathbf{Y}}{dt}=\mathbf{F}'(t)\delta\mathbf{Y}. \tag{5}$$

Where $\mathbf{F}'(t)$ is the linear Jacobian operator obtained by differentiating $\mathbf{F}(t)$ with respect to $\mathbf{Y}$. Since the Eq. (5) (namely of TLM, i.e. tangent linear model) being linear, its solution at a given time t_i depends linearly on the initial condition at time t_1, which can be expressed as

$$\delta\mathbf{Y}(t_i)=\mathbf{L}(t_i,t_1)\delta\mathbf{Y}(t_1). \tag{6}$$

Where $\mathbf{L}(t_i,t_1)$ is called the resolvent of equation on (4) between time t_i and t_1. Substituting Eq. (6) into Eq. (4), we get

$$\begin{aligned}\delta J(\mathbf{Y}(t_1))&=\sum_{i=1}^{n}<\mathbf{W}(t_i)(\mathbf{Y}(t_i)-\hat{\mathbf{Y}}(t_i)),\mathbf{L}(t_i,t_1)\delta\mathbf{Y}(t_1)>\\&=\sum_{i=1}^{n}<\mathbf{L}^{*}(t_i,t_1)\mathbf{W}(t_i)(\mathbf{Y}(t_i)-\hat{\mathbf{Y}}(t_i)),\delta\mathbf{Y}(t_1)>.\end{aligned} \tag{7}$$

Since we have the relation $\delta J=<\nabla J,\delta\mathbf{Y}>$, thus

$$\nabla J(\mathbf{Y}(t_1))=\sum_{i=1}^{n}\mathbf{L}^{*}(t_i,t_1)\mathbf{W}(t_i)(\mathbf{Y}(t_i)-\hat{\mathbf{Y}}(t_i)). \tag{8}$$

Clearly $\mathbf{L}^{*}(t_i,t_1)$ is the adjoint of $\mathbf{L}(t_i,t_1)$ and it is indeed very difficult to evaluate. In order to obtain $\mathbf{L}^{*}(t_i,t_1)$, we introduce at this point the adjoint equation of (5)

$$-\frac{d\delta\mathbf{Y}^{*}}{dt}=\mathbf{F}'^{*}(t)\delta\mathbf{Y}^{*}. \tag{9}$$

Where $\delta\mathbf{Y}^{*}$ also belongs to the space Ω, and $\mathbf{F}'^{*}(t)$ represents the adjoint of $\mathbf{F}'(t)$. It is similar with Eq. (6) that

$$\delta \mathbf{Y}^*(t_1) = \mathbf{L}^*(t_i, t_1)\delta \mathbf{Y}^*(t_i). \tag{10}$$

Notice that Eq. (10) is compatible and indeed identical with Eq. (8), if and only if one takes $\delta \mathbf{Y}^*(t_i) = \mathbf{W}(t_i)(\mathbf{Y}(t_i) - \hat{\mathbf{Y}}(t_i))$ as the initial value at t_i time step and integrates Eq. (9) backward from t_i to t_1 and summing the result $\delta \mathbf{Y}^*{}_i(t_1)$ in the period of $[t_n, t_1]$:

$$\nabla J(\mathbf{Y}(t_1)) = \sum_{i=1}^{n} \mathbf{L}^*(t_i, t_1)\mathbf{W}(t_i)(\mathbf{Y}(t_i) - \hat{\mathbf{Y}}(t_i)) = \sum_{i=1}^{n} \delta \mathbf{Y}^*{}_i(t_1). \tag{11}$$

Since Eq. (11) is linear, $\nabla J(\mathbf{Y}(t_1))$ might be obtained through a single integration of adjoint Eq. (9) from time t_i to t_1; and at each time step, the weighting difference $\mathbf{W}(t_i)(\mathbf{Y}(t_i) - \hat{\mathbf{Y}}(t_i))$ is inserted.

The indirect approach algorithm can be expressed as follows: firstly, obtain the partial derivative adjoint Eq. (9) indirectly from the basic Eq. (1); then, discretize the adjoint equation to obtain adjoint code (or call it adjoint model); finally, obtain the gradient of the cost function by the adjoint model.

2.2 *Direct approach algorithm (conjugate code method)*

As we described above, the indirect approach is actually a general way, which is to find adjoint equation analytically, then to express it numerically. Consequently, it leads the complex expression of adjoint equation. On the other hand, the direct approach means that one might obtain discrete adjoint model directly from the code, which is the discrete formulation of model, i.e., Eq. (1). It implies that we view the forward-integrating numerics of Eq. (1) as the result of the multiplication of linear operators, or sy, the multiple subroutines or logical loops. This idea of direct approach can be schematically expressed in an analogue form of Eq. (6):

$$\delta \mathbf{Y}(t_n) = \mathbf{L}_n \mathbf{L}_{n-1} \cdots \mathbf{L}_1 \delta \mathbf{Y}(t_1). \tag{12}$$

Here each operator $\mathbf{L}_i$ symbolically represents either a subroutine or a single logical loop. Then the adjoint model simply turns to

$$\delta \mathbf{Y}^*(t_1) = \mathbf{L}_1^T \mathbf{L}_2^T \cdots \mathbf{L}_n^T \delta \mathbf{Y}^*(t_n). \tag{13}$$

Thus, substitute Eq. (13) into Eq. (11), we get $\nabla J(\mathbf{Y}(t_1))$; which is in turn inserted in Eq. (3), the optimal initial condition is then obtained.

Clearly any $\mathbf{L}_i^T$ is the transpose matrix of $\mathbf{L}_i$ in linear space, and it can be easily obtained if the operator $\mathbf{L}_i$ is readily known (see the example in appendix of Navon et al., 1992). Furthermore, define a multiple linear operator $\mathbf{M} = \mathbf{L}_n \mathbf{L}_{n-1} \cdots \mathbf{L}_1$, and $\mathbf{M}^T = \mathbf{L}_1^T \mathbf{L}_2^T \cdots \mathbf{L}_n^T$, so we have

$$\delta \mathbf{Y}(t_n) = \mathbf{M}\delta \mathbf{Y}(t_1), \tag{14}$$

and

$$\delta \mathbf{Y}^*(t_1) = \mathbf{M}^T \delta \mathbf{Y}^*(t_n). \tag{15}$$

Now it is seen that

$$\begin{aligned} <\delta \mathbf{Y}(t_n), \delta \mathbf{Y}^*(t_n)> &= <\mathbf{M}\delta \mathbf{Y}(t_1), \delta \mathbf{Y}^*(t_n)> = <\delta \mathbf{Y}(t_1), \mathbf{M}^T \delta \mathbf{Y}^*(t_n)> \\ &= <\delta \mathbf{Y}(t_1), \delta \mathbf{Y}^*(t_1)>. \end{aligned} \tag{16}$$

This formula means that the input of an inner production $< \delta\mathbf{Y}(t_1),\delta\mathbf{Y}^*(t_1)>$ at initial time should be equal to the output $< \delta\mathbf{Y}(t_n),\delta\mathbf{Y}^*(t_n)>$ at final step, if the direct approach is correctly constructed.

On the other hand, as we illustrated in indirect approach, by using Eqs. (5) and (9), one can easily verify that

$$\frac{d}{dt} < \delta\mathbf{Y}(t),\delta\mathbf{Y}^*(t)> = < \frac{d\delta\mathbf{Y}(t)}{dt},\delta\mathbf{Y}^*(t)> + < \delta\mathbf{Y}(t),\frac{d\delta\mathbf{Y}^*(t)}{dt} >$$
$$= < \mathbf{F}'(t)\delta\mathbf{Y}(t),\delta\mathbf{Y}^*(t)> - < \delta\mathbf{Y}(t),\mathbf{F}'^*(t)\delta\mathbf{Y}^*(t)> = 0, \quad (17)$$

thus

$$< \delta\mathbf{Y}(t),\delta\mathbf{Y}^*(t)>_{t=n} = < \delta\mathbf{Y}(t),\delta\mathbf{Y}^*(t)>_{t=1}. \quad (18)$$

The consistency existing in both Eqs. (18) and (16) has demonstrated that indirect and direct approach is theoretically identical. Nevertheless, the inconsistency is generally arisen from the analytical form of the adjoint equation with its imperfect discrete approximation.

3. Forward and backward schemes under semi–Lagrangian framework

Semi–Lagrangian numerical framework is a hybrid idea for traditional Eulerian and Lagrangian scheme. It wants to get best of two schemes: the regular resolution of Eulerian discretion and the enhanced stability of Lagrangian one. This is achieved by using a different set of air particles at each time step, the set of particles being chosen such that they arrive exactly at the points of a regular Cartesian mesh at the end of the time step. In the past decade, the discretion under semi–Lagrangian framework has elicited considerable interest for the efficient integration of weather forecast models, since they allow larger time steps with no loss of accuracy than the Eulerian framework.

Although semi–Lagrangian framework is mainly described for a hierarchy of applications (passive advection, forced advection, and coupled advection) of increasing complexity. However under methodology of time–splitting technique, Eq. (1) can be simply split into two fractional steps (cf. Wang, 1997a):

$$\frac{\partial\mathbf{Y}}{\partial t} = \mathbf{F}(\mathbf{Y}) \qquad \text{(adjustment step)} \quad (19a)$$

$$\frac{\partial\mathbf{Y}}{dt} = 0 \qquad \text{(advection step)} \quad (19b)$$

And the discretion of Eq. (19a) can be implemented by leapfrog time scheme, for instance; and as demonstrated by Talagrand and Courtier (1987), the adjoint of the leapfrog time differencing scheme is readily addressed. Therefore without loss of generality, only the discretion of passive advection problem of Eq. (19b) under semi–Lagrangian framework is of interest here.

For a passive quantity φ, the passive equation without diffusion and source or sink reads

$$\frac{\partial\varphi}{\partial t} = -\nabla\cdot(\mathbf{V}\varphi) - \mathbf{V}\cdot\nabla\varphi - \varphi\nabla\cdot\mathbf{V}. \quad (20)$$

Frequently, Eq. (20) is used in a form where $-\varphi\nabla\cdot\mathbf{V}$ has been neglected, referring $-\mathbf{V}\cdot\nabla\varphi$ as the advection term. Note that $\nabla\cdot\mathbf{V}=0$ is a justified assumption for all meteorological flows, except in the very smallest scale which is not concerned here. Eq. (20) thus

comes to

$$\frac{d\varphi}{dt} = \frac{\partial \varphi}{\partial t} + \frac{d\mathbf{x}}{dt} \cdot \nabla\varphi = \frac{\partial \varphi}{\partial t} + \mathbf{V} \cdot \nabla\varphi = 0, \tag{21}$$

$$\frac{d\mathbf{x}}{dt} = \mathbf{V}(\mathbf{x},t). \tag{22}$$

Here, $\mathbf{x}$ is the position vector (in 1–2– or 3D); ∇ is the gradient operator, and $\mathbf{V}(\mathbf{x},t)$ is the given function. The hyperbolic character of Eqs. (21, 22) states that the transport of φ is constant along a fluid trajectory (or characteristic path). The essence of semi–Lagrangian advective scheme comes to be approximately integrating the Eq. (21) along the approximated trajectory Eq. (22).

Easily, we get the corresponding perturbation and adjoint formula of Eq. (21).

$$\frac{d(\delta\varphi)}{dt} = 0, \tag{23}$$

$$-\frac{d(\delta\varphi^*)}{dt} = 0. \tag{24}$$

And indeed it is the corresponding tangent linear equations of (5) and (9) under the certain condition $\nabla \cdot \mathbf{V} = 0$. Both of Eqs. (23), (24) tell us, the backward integration of the adjoint difference is analytically equivalent to integrating this difference backward in the model equations. As illustrated in Fig. 1(a) and Fig. 1(b), the characteristic curve PQ in forward integration is different with the backward curve QP. Both curves are numerically determined by iterations (cf. Staniforth and Côte, 1991). It is seen that in most situations, the numerical difference exists between the backward and forward way since the imperfect approximation of fluid path, except in the case of $\mathbf{V} = constant$.

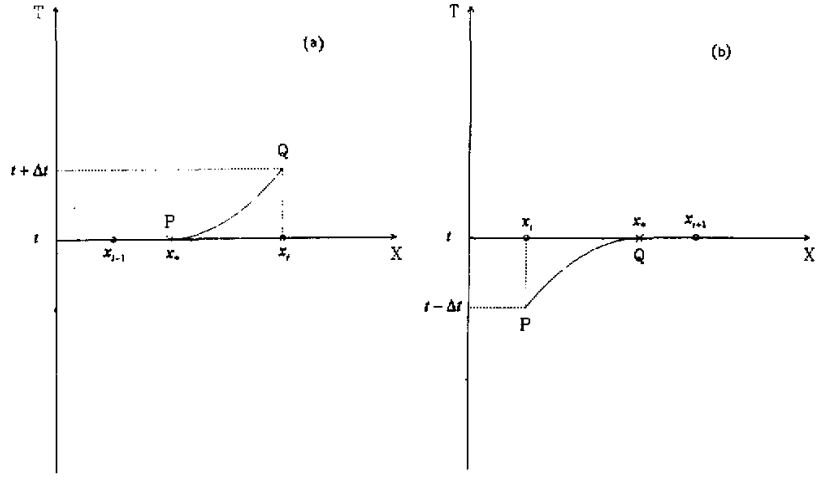

Fig. 1. (a) Forward semi–Lagrangian advection scheme; (b) Backward semi–Lagrangian advection scheme.

As shown in Fig. 1(a) and Fig. 1(b), the forward numerical integration can be expressed as $\varphi(Q,t+\Delta t)=\varphi(P,t)$. And the first guess fluid path is taken $PQ=\mp|\mathbf{V}|dt$. On the other hand, for backward numerical integration, we have $\varphi^*(P,t-\Delta t)=\varphi^*(Q,t)$ and $QP=\pm|\mathbf{V}|dt$ as the guess path.

As mentioned before, the main idea in semi-Lagrangian advection scheme is that the arriving point (i.e., the position Q in forward or P in backward integration, respectively) is always designed at the grid of mesh. And the departing point (i.e. the position P in forward and Q in backward integration) is generally located in somewhere in the interval of mesh. Therefore, $\varphi(P,t)$ and $\varphi^*(Q,t)$ have to be determined by interpolation between the intervals. In our case, the optimal cubic spline (it means the cubic spline with natural boundary condition) is used (Wang, 1997a). In order to keep the shape-preserving interpolation function, the proposed monotone constraint is also employed in the spline (see Wang, 1997b). Such constrained cubic spline interpolation has provided very high accuracy in the numerical advection integration and the spurious numerical dissipation and dispersion are well eliminated to minimum. Such high-accuracy offers an opportunity to perform the advection scheme without engagement of any kind numerical smoothness / diffusion / filter and so on.

Conclusively, the semi-Lagrangian advection scheme provides a unique platform to benchmark the pronounced two approaches. It has been shown that the indirect approach in semi-Lagrangian framework comes to be extremely simple. For both of the forward and backward integration, the possible inconsistency arising from smoothness (such inconsistency due to the smoothness can increase assimilating error to higher level, see the Fig. 8 in the paper of Navon et al., 1992) completely disappears. Another inconsistency from the derivation of the adjoint equation following by the discrete approximation is negligible, because the exactly same analytical form persists in the adjoint equation and its corresponding TLM, as well as the high accuracy of the numerical approximation in the proposed semi-Lagrangian backward and forward scheme.

In summary, the crucial indirect approach of backward integration in semi-Lagrangian framework is outlined here:

Step 1: Determining the backward fluid path by using the first guess value of $QP=\pm|\mathbf{V}|dt$, and then integration Eq. (22) backward.

Step 2: Integrating Eq. (24) following to the backward path, i.e., $\delta\varphi^*(P,t-\Delta t)=\delta\varphi^*(Q,t)$.

Step 3: Inserting $\delta\varphi^*(t_1)$ into Eq. (11) in order to obtain the cost function.

The direct way begins to represent numerical formalism of Eq. (I) as we have already mentioned. In fact the consequent construction of the codes is no difference with the scheme proposed by Rostating et al. (1993).

Clearly, indirect approach in semi-Lagrangian framework is a very simple way in particular; the direct approach is, as it was, a general way in natural. The two approaches will be numerically tested in the following section and the statistic error-level comparison will be given.

4. Numerical test

In order to assess the numerical accuracy and efficiency in direct and indirect approach under the semi-Lagrangian framework, the so-called cone test is introduced by rotating a "consine hill" passive scalar distribution in two dimensions. The numerical dissipation and

dispersion of each approach can be easily visualized in such simple test: A 2D underlying function is one sonsine–hill distribution of concentration with a peak value of 3.87 (Figs. 2a,c). Then this is advected in a rotational flow field, in which the angle velocity is a constant in a clockwise direction (Fig. 2d). The peak value and distribution should remain constant throughout the rotation. Therefore, a measure of the accuracy of a method lies in its ability to transport the distribution without change.

Meanwhile a noisy consine–hill distribution is constructed by its underlying contribution perturbed by an artificial two–grids–noise maker with a reduced peak value to 2.0; this is introduced for the sensitive test of initial conditions, and here it is taken as an imperfect initial field (Fig. 2b). Both the imperfect and perfect initial concentrations, in the two approaches, are advected by same rotational wind field, and then compared with the corresponding analytical solutions in a statistic fashion.

As we list in Table 1, four tests are conducted. That is: Case (I) is the indirect approach

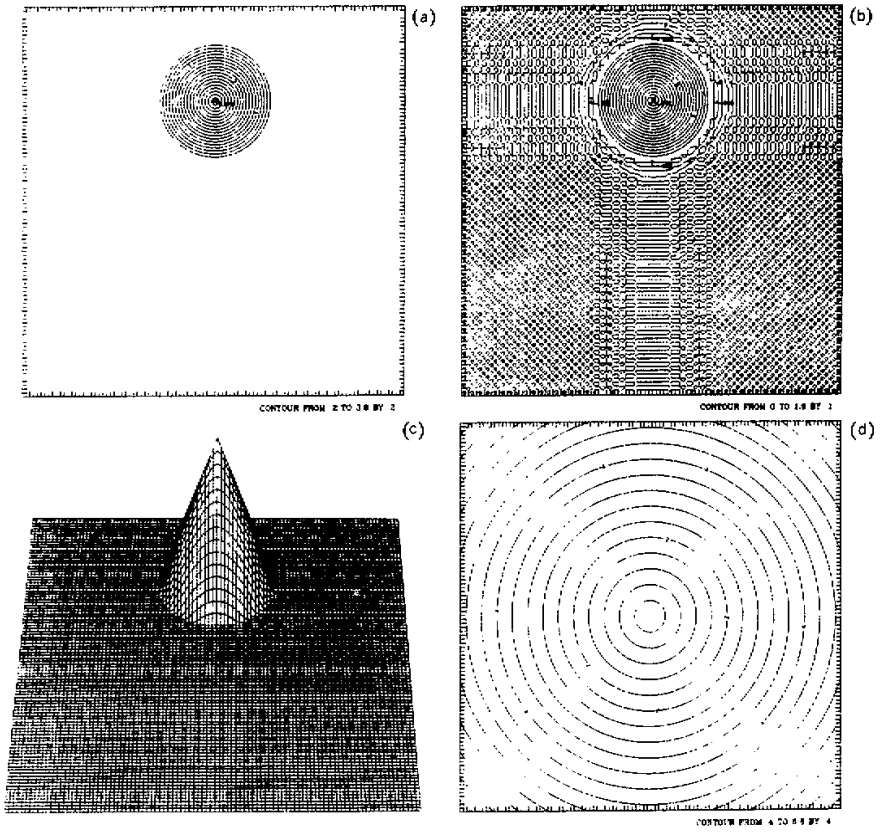

Fig. 2. (a) The 2D perfect initial distribution of cone; (b) The 2D imperfect initial distribution of cone; (c) The 3D perspective view of Fig. 2a; (d) The horizontal distribution of the background wind field.

with the perfect initial condition; Case (II) is indirect approach with the imperfect initial condition; Case (III) is the direct approach with perfect approach; Case (IV) is the direct approach with imperfect approach.

Table 1. Statistic assessment for various cases in cone test under Semi-Lagrangian framework

		Case 1	Case 2	Case 3	Case 4
Initial	MAX	3.87	2.0	3.87	2.0
state	MIN	0.0	−0.05	0.0	−0.05
Model	MAX	2.544	1.313	2.544	1.313
results	MIN	5.39E−9	2.06E−9	5.39E−9	2.06E−9
without	RMSE	7.65E−2	0.239	7.65E−2	0.239
VAR	J	88.545	1746.61	88.545	1746.61
Model	MAX	2.755	2.756	2.596	2.339
results	MIN	2.32E−6	2.075E−6	2.27E−5	2.11E−5
with	RMSE	5.35E−2	5.29E−2	6.96E−2	9.89E−2
VAR	J	43.39	42.13	72.16	197.84

In Table 1, the numerical results after six rotating cycles are also given. The number of the time steps in every cycle is 6284. There are total 6 rotating cycles, and therefore 37707 time steps. If one time step stands for 10 seconds, the total time will thus reach 4.4 days, which represents well the characteristic time-scale for a typical weather-scale phenomenon. After each cycle the model numerical solutions $\varphi_i (I= 1,2,\cdots,7)$ are stored. The corresponding estimate of solutions $\hat{\varphi}_i (I= 1,2,\cdots,7)$ of φ_i is all equal, in those ideal cases, to the analytical values derived from the underlying function. Therefore the cost function $J= \frac{1}{2}\sum_{i=1}^{7}(\varphi_i - \hat{\varphi}_i)^2$ here denotes the discrepancy between the analytical solutions and the model solutions. In our iteration procedure, since the " quasi-Newton" scheme is used, so we have $\rho= \frac{1}{\hat{c}\nabla J / \partial\varphi}$. However, because the term is difficult to solve numerically, it is thus replaced by $\rho= \frac{1}{J}$. And in order to accelerate the convergent rate, it can be multiplied by some certain coefficients. The iteration is keeping until some criteria are satisfied. For example, it satisfies $\nabla J< \varepsilon$, where ε is a small constant.

As shown in Table 1, for four cases, the model results with both indirect and direct approaching VAR technique are superior than the results without VAR. The effect of optimal initial conditions due to the VAR adjustment is thus in evidence. And we also can find that the indirect approach can make the cost function or RMSE smaller than the direct approach one.

As demonstrated by comparison of Cases (I) and (II) or Cases (II) and (IV) in Fig. 3, we find that it converges to the prescribed criterion more feasible with the perfect initial conditions than with the imperfect one. Nevertheless, the convergent rate of the imperfect initial conditions is larger than the perfect one, particularly at the beginning steps of the iteration. Such sensitivity of initial conditions indicates that the perfect initial conditions possibly are not the optimal initial conditions to the model. Indeed, the perfect conditions (i.e., $J=0$ since it is exactly equal to the analytical solution) have to be adjusted to minimize the errors from the numerical approximation of model.

To compare indirect and direct approach, i.e. the comparison of the Cases (I) (II) against the Cases (III) and (IV), we found that the simulation results for two approaches are nearly identical in a numerical sense (the plotted results have not been shown), although the indirect approach has shown somewhat superiority in the term of numerical accuracy. Furthermore, the big differences between the cost function J, the gradient $|\nabla J|$ in indirect and direct approach indicate that the indirect way is a faster way and therefore it is a time efficient for semi–Lagrangian framework. This becomes much clear as shown in Fig. (3). It is seen that the varying of the cost function J, the gradient $|\nabla J|$, the root mean square error (RMSE) with iteration steps illustrated all the cost function J, the gradient $|\nabla J|$ and RMSE are gradually reduced with the increasing of the iteration times.

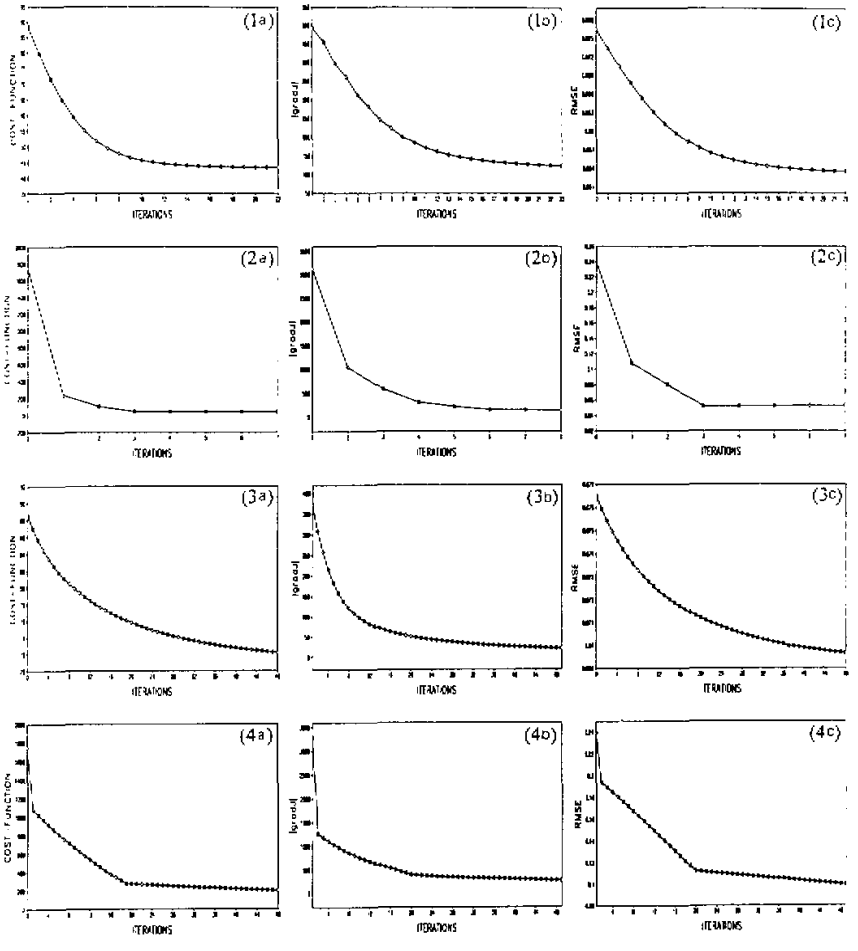

Fig. 3. The varying of the cost function J, the gradient $|\nabla J|$, and the root mean square error (RMSE) with the iteration steps. (1a,1b,1c) in Case (I); (2a,2b,2c) in Case (II); (3a,3b,3c) in Case (III); (4a,4b,4c) in Case (IV) (see text).

Nevertheless, the decrease of the cost function J and the gradient $|\nabla J|$ in direct approach is much slower than the indirect approach is. Meanwhile the time varying of RMSE for both approaches did not show a big difference.

5. Conclusion

The time splitting semi-Lagrangian framework was suggested by Leslie and Purser (1991). It has been proved that the framework is one of the most fast algorithms (cf. the review of Staniforth and Côte, 1991). It also offers the flexibility for grid configuration; for instance, the framework can been used in Arakawa-A grid mesh (Leslie et al., 1981), and therefore it greatly decreases the numerical complexity.

In such framework, we have discussed the two approaches of VAR technique. It has been verified that the direct approach (conjugate code method) and the indirect approach (numerical approximation of analytical form of adjoint equation) are not only identical in theoretical, they are also identical in numerical. However, the indirect approach is much efficient than the direct approach, since the inconsistency between the backward and forward integration in the indirect way is negligible under the semi-lagrangian framework.

The proposed semi-Lagrangian backward scheme in fact can be directly used in passive models, such as the dispersion (air pollution) and budget model. Generally, we think that if one can successfully split the diabatic and adiabatic terms in weather forecast model, the algorithms of the indirect approach should be much more feasible and efficient than that of the direct approach. This requires, of course, intensive sensitivity tests in future.

We wish to thank Mr. Wang Xingbao, Ms. Fang Juan, and Ms. Wei Ming for helpful discussion and useful comments. The research is financially supported by the National Natural Science Foundation of China, the President Foundation of Nanjing University, the Project Sponsored by the Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry, and "973" National Research Project.

REFERENCES

Chao W. C., and L.P. Chang, 1992: Development of a four-dimensional variational assimilation system using the adjoint method at GLA. Part I: Dynamics. *Mon. Wea. Rev.*, **120**, 1661–1673.

Chen Zitong, Shen Tongli, Ding Yihui, and Sun Linping, 1998: Data assimilation system for mesoscale models part I: Layout of an adjoint model. *J. Nanjing Institute of Meteorology*, **21(2)**, 165–172.

Gao Jidong, and Chou Jifan, 1995: The effects of the model sensitivity to initial condition upon the variational four-dimensional assimilation — the study based on Lorenz model. *Acta. Meteorologica. Sinica.*, **53(4)**, 471–479.

Gill, P. E., W. Murray, and M. E. Schlesinger, 1982: *Practical optimization.* Academic Press, London.

Leslie, L. M., G. A. Mills, and D. J. Gauntlett, 1981: The impact of FGGE data coverage and improved numerical techniques in numerical weather prediction in the Australian region. *Quart. J. R. Met. Soc.*, **107**, 629–642.

Leslie, L. M., and G. A. Purser, 1991: High-order numerics in an unstaggered three-dimensional time-split semi-Lagrangian forecast model. *Mon. Wea. Rev.*, **119**, 1612–1623.

Lewis J. W., and J. C. Derber, 1985: The use of adjoint equation to solve a variational adjustment problem with advective constraints. *Tellus*, **37A**, 309–322.

Le Dimet F., and O. Talagrand, 1986: Variational algorithms for analysis and assimilation of meteorological observations: Theoretical aspects. *Tellus*, **38A**, 97–110.

Navon, I. M., X. Zou, J. Derber, and J. Sela, 1992: Variational data assimilation with an adiabatic version of the NMC spectral model. *Mon. Wea. Rev.*, **120**, 1433–1446.

Robert, A., 1981: A stable numerical integration scheme for the primitive meteorological equations. *Atmos. Ocean*, **19**, 35–46.

——, 1982: A semi-Lagrangian and semi-implicit numerical integration scheme for the primitive meteorological equations. *J. Meteor. Soc. Japan*, **60**, 319–324.

Rostating N., S. Dalmas, and A. Galligo, 1993: Automatic differentiation in Odyssee. *Tellus*, **45(A)**, 558–568.

Staniforth, A. and J. Côte, 1991: Semi-Lagrangian integration schemes for atmospheric models—A review. *Mon. Wea. Rev.*, **119**, 2206–2223.

Talagrand O., and P. Courtier, 1987: Variational assimilation of meteorological observations with the adjoint vorticity equation. Part I: Theory. *Quart. J. Roy. Meteor. Soc.*, **113**, 1311–1328.

Wang Yuan, 1997a: High-resolution evaluation of meso-scale flow driven by topographic irregularities on telescopic-meshed domain. Ph.D. Dissertation, ASTR, University Catholique De Louvain. Belgique.

——, 1997b: Quasi-monotonicity constraints can be used for shape-preserving interpolation. Ph. D. Dissertation Annexe, ASTR, Universite Catholique De Louvain, Belgique.

Zhu Jiang, 1995: Four-dimensional data quality control: variational method. *Acta. Meteorologica. Sinica.* **53(4)**, 480–487.