INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2006; **51**:719–748
Published online 20 December 2005 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1145

A comparative study of the performance of high resolution advection schemes in the context of data assimilation

S. Akella[‡] and I. M. Navon[*,†]

School of Computational Science and Department of Mathematics, Florida State University, Tallahassee, FL 32306, U.S.A.

SUMMARY

High resolution advection schemes have been developed and studied to model propagation of flows involving sharp fronts and shocks. So far the impact of these schemes in the framework of inverse problem solution has been studied only in the context of linear models. A detailed study of the impact of various slope limiters and the piecewise parabolic method (PPM) on data assimilation is the subject of this work, using the nonlinear viscous Burgers equation in 1-D. Also provided are results obtained in 2-D using a global shallow water equations model. The results obtained in this work may point out to suitability of these advection schemes for data assimilation in more complex higher dimensional models. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: finite volume methods; high resolution schemes; adjoint model; data assimilation; Burgers equation

1. INTRODUCTION

Spatial discretization methods for solving partial differential equations (PDEs) can be broadly classified as finite difference (FD) [1, 2], finite volume (FV) [3], finite element (FE) [4–6] and spectral methods [7] (including the discontinuous Galerkin (DG) methods [8]). All of these methods combined with explicit or implicit time integration schemes can be effectively applied to solve PDEs (of various types such as hyperbolic, parabolic and elliptic).

For numerical solutions of conservation laws, such as the Euler equations in gas dynamics [9] which describe evolution and propagation of flows involving sharp fronts and shocks, several methods have been suggested in the FD, FV, FE, spectral and DG methods literature. Some of the most popular methods in the FV context are Lax–Wendroff, Lax–Friedrichs,

*Correspondence to: I. M. Navon, School of Computational Science, Florida State University, Tallahassee, FL 32306, U.S.A.
†E-mail: navon@csit.fsu.edu
‡E-mail: sakella@math.fsu.edu

Contract/grant sponsor: NSF; contract/grant number: ATM-9731472

Received 6 April 2005
Revised 23 September 2005
Accepted 26 September 2005

Roe's, flux corrected transport (FCT) methods of Boris–Book and Zalesak, slope limited methods of van Leer, piecewise parabolic method (PPM) of Colella and Woodward, essentially non-oscillatory (ENO) schemes of Harten–Shu–Osher (see References [3, 10, 11] for details of these methods), to name a few. In the FD context, please refer Reference [2], FE [6], spectral and DG methods [7, 8], respectively, for details.

In geophysical fluid dynamics problems, discontinuities usually do not develop from smooth initial conditions; except in cases such as the formation of hydraulic jumps that evolve in the shallow-water flows from smooth initial data. For instance in mid-latitudes, fronts can be formed in low-pressure systems, yet these fronts are not entirely discontinuities. Atmospheric fronts (also substances such as chemical pollutants) are transported from one location to another, described very well by a tracer advection model. Due to the deformation (stretching and shearing) of the velocity field that advects the front, discontinuities can be formed on the resolution scale of the (computational) model, see Section 5.3 of Reference [11] for details. As a result of finiteness of clouds, variables such as moisture (density) and temperature are discontinuous (once again, on the scale of the model resolution) across the interface of the cloud [12]. Therefore, from a purely computational stand point, there is a need to apply numerical schemes devised for numerical solutions of conservation laws which support discontinuous solutions, in the geophysical fluid flows. Rood [13] provided a detailed analysis and comparison of various advection schemes for a simple linear atmospheric transport model. Lin *et al.* [14] have analysed the effect of varying the slope limiters using an atmospheric general circulation model. Lin and Rood [15] have compared the first-order upwind, central difference, PPM (modified monotonic and positive definite) and monotonic van Leer schemes [16]. Towards the development of a fully operational atmospheric general circulation model based on FV discretization [17], Lin and Rood [18] have implemented slope limited van Leer schemes and the PPM scheme on a shallow water equations model using a semi-Lagrangian semi-implicit time integration scheme. For a discussion and applications of other popular schemes such as MPDATA of Smolarkiewicz [19, 20] and QUICK of Leonard [21–23], see Reference [24]. MPDATA is also used in the hybrid coordinate ocean model (HYCOM) [25] for advection. Several numerical schemes that are ENO and total variation diminishing (TVD) type have been tested and compared using the rotational and deformational flow-field test cases by Sokol [26]. Iskandarani *et al.* [27] provide a comparison of the continuous Galerkin, discontinuous Galerkin, spectral finite volume (with a FCT limiter) and Taylor Galerkin least square methods using a linear advection mathematical model.

Fusing models with measurements (observations) and finding response of a system to (external) disturbances, all require solving inverse problems and as such the minimization of a goal functional, whose gradient with respect to control variables is efficiently provided via adjoint methods, see References [28–30]. In aerodynamics applications, such as minimization of drag, maximization of lift (as target functionals) are often performed by considering the geometry of the immersed body in the fluid as the control variable, which is called shape optimization [31], for other applications and details, see Reference [28].

Present weather forecasting system primarily utilizes meteorological data collected (temperature, wind velocity, pressure, humidity, etc.) from various sources (land stations, balloons, buoys, ships, aircrafts, satellites, radiosondes, rawinsondes, etc.) and numerical weather prediction (NWP) models' (or, the computer models) forecasts. The NWP models integrate atmospheric models as initial-value problems, i.e. given one day's weather observations, the evolution of the atmosphere in the next few days is obtained by integrating in time the NWP

models (therefore, these are also referred to as *forward* models). Hence a *good* forecast requires that these NWP models represent accurately the dynamics of the atmosphere (including the oceans) and the initial conditions supplied for integration (of the NWP models) be known accurately. Estimation of the state or evolution of the atmosphere using the information provided by NWP model prediction and observations of the atmosphere is carried out by data assimilation (DA) [30]. Current DA methods are based on either using results from estimation theory (such as application of Kalman, extended Kalman, ensemble Kalman filtering, for details, once again, please see Reference [30] and references therein) or variational methods, which are based on minimization of a cost functional which measures the distance (in a suitable norm) between observations and NWP model forecasts. The objective of variational data assimilation is to determine a model trajectory (by adjusting initial conditions used for model integration) that satisfies the model equations as a (strong or weak) constraint while simultaneously minimizing the lack of fit between model predictions and heterogeneous observations in a least-squares sense. Please see References [32, 33] for further details. Large scale minimization algorithms which require availability of gradient of the cost functional with respect to the control variables (provided efficiently by adjoint methods, which are integrated backwards in time) are used for this purpose.

The impact of different discretization techniques for the advection term(s) in the framework of inverse problems and problems related to DA have not been extensively tested, except for work by Vukićević *et al.* [24] and Thuburn and Haine [34]. In Reference [24] the authors performed DA experiments to reveal the relationships between their properties with respect to data assimilation with three different (central difference: LEAPFROG, MPDATA, QUICK) schemes for the advection of a passive tracer in two dimensions using a linear 2-D transport equation. Their results indicate that more accurate advection schemes need to be used to solve both, forward and adjoint models in time to achieve higher accuracy regarding recovery of initial conditions for data assimilation; also the same discretization scheme should be applied consistently both for forward and adjoint model integrations. Thuburn and Haine [34] recall Godunov' s theorem (which states that any linear monotonic advection scheme cannot provide more than first-order accuracy), they studied the affects on adjoint sensitivity computations using a nonlinear, nonoscillatory (QUICK) scheme on a one-dimensional linear advection equation model. They also suggest modifications to advection schemes to obtain adjoint sensitivity results that are meaningful (in the particular physical setting considered by them). In this context, a total variation diminishing (TVD) scheme based on a slope limiter has been suggested.

Since the mathematical models used to study fluid flows and weather prediction are highly nonlinear, as a step towards understanding the effects of using high order advection schemes in DA, we study in this work the impact of using FV methods that are slope limited using van Leer type and PPM for spatial discretization (in 1-D and 2-D). In one dimension, a nonlinear viscous Burgers equation model and in 2-D the spherical global shallow water equations have been used as proxy for more complex NWP models. We show that for a particular smooth initial condition, we obtain a smooth solution for these model problems (in the context of smoothness property of geophysical flows as discussed above), and implement the adjoint method to conduct DA experiments.

The paper is organized as follows. In Section 2 we present the forward model as well as describe numerical solution of the nonlinear Burgers equation using FV discretization. Section 3 describes the test case considered along with results obtained using several slope

limited TVD schemes as described in Section 2. Section 4 describes the derivation of the adjoint and tangent linear models (which are used for DA) and verification of these discrete models. Section 5 provides a brief description of the minimization algorithm used. The performance of the various slope limited and PPM schemes for the minimization of a certain cost functional, in other words, in DA experiments (1-D and 2-D) is presented in Section 6. Finally, in the section of summary and conclusions we discuss the impact of the different advection schemes in the framework of our numerical results.

2. DESCRIPTION OF THE MATHEMATICAL AND NUMERICAL MODELS

The Burgers equation [35] will be used to present detailed formulation of the various slope limiters and the PPM advection scheme in one space dimension. The formulation extends readily to 2-D for the global shallow water equations model, which will be discussed later in this section.

Let us consider the following 1-D (nonlinear) scalar conservation law ($\phi(x,t) \in \mathbb{C}^2$), the space of continuous functions that are at-least twice differentiable)

$$\frac{\partial \phi}{\partial t} + \frac{\partial f}{\partial x} = \frac{\partial S}{\partial x} \tag{1}$$

where f is a convex flux function given by $\phi^2/2$ and S represents the source term(s). Equation (1) is the well known Burgers equation which is a very important fluid dynamical model useful for conceptual understanding of nonlinear waves, shock formation [9, 36] and turbulence [37]. Various numerical schemes (see Fletcher [38] for a detailed numerical analysis) have been suggested and tested on this model equation to efficiently capture shocks.

We will now describe and test a variety of finite volume methods [3] to solve the above equation, all differing in the way the solution, ϕ is reconstructed in each cell using different slope limiters. We will closely follow the approach taken by monotone upstream-centred schemes for conservation laws (MUSCL), see References [16, 39–42].

Let us start by writing the integral form of (1) within the ith cell, $\mathscr{C}_i$,

$$\frac{\partial}{\partial t} \int_{\mathscr{C}_i} \phi(x,t)\,\mathrm{d}x = f[\phi(x_{i-1/2},t)] - f[\phi(x_{i+1/2},t)] + S(x_{i+1/2},t) - S(x_{i-1/2},t) \tag{2}$$

$\mathscr{C}_i : x \in [x_{i-1/2}, x_{i+1/2}]$.

We define ith cell average at time interval t_n ($t \in [t_0, t_{\text{final}}]$ has been discretized into a number of time steps $[t_0, t_1, \ldots, t_n]$) as

$$\Phi_i^n \approx \frac{1}{\Delta x_i} \int_{x_{i-1/2}}^{x_{i+1/2}} \phi(x,t_n)\,\mathrm{d}x \tag{3}$$

where $\Delta x_i = x_{i+1/2} - x_{i-1/2}$ is the length of the ith cell.

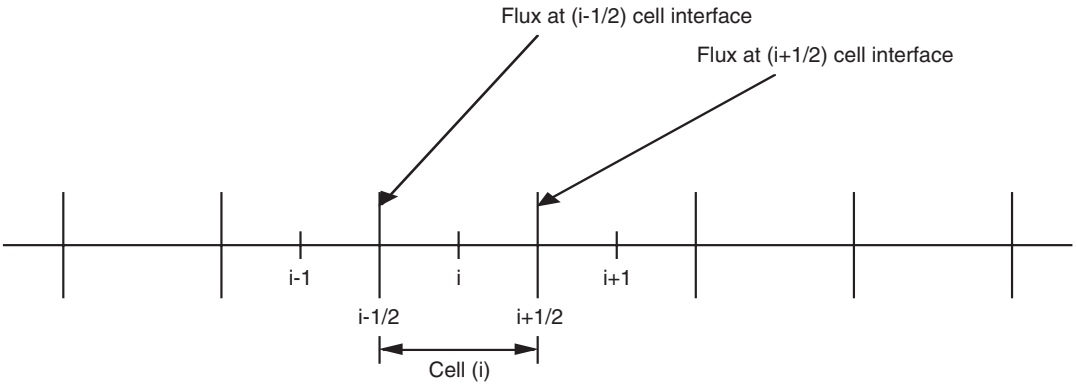

Figure 1. Finite volume discretization.

Integration of Equation (2) from t_n to t_{n+1} yields

$$\int_{\mathscr{C}_i} \phi(x,t_{n+1})\,\mathrm{d}x - \int_{\mathscr{C}_i} \phi(x,t_n)\,\mathrm{d}x = \int_{t_n}^{t_{n+1}} f[\phi(x_{i-1/2},t)]\,\mathrm{d}t - \int_{t_n}^{t_{n+1}} f[\phi(x_{i+1/2},t)]\,\mathrm{d}t + \int_{t_n}^{t_{n+1}} [S(x_{i+1/2},t) - S(x_{i-1/2},t)]\,\mathrm{d}t$$

dividing by Δx_i and rearranging

$$\frac{1}{\Delta x_i}\int_{\mathscr{C}_i} \phi(x,t_{n+1})\,\mathrm{d}x = \frac{1}{\Delta x_i}\int_{\mathscr{C}_i} \phi(x,t_n)\,\mathrm{d}x - \frac{1}{\Delta x_i}\int_{t_n}^{t_{n+1}} \{f[\phi(x_{i+1/2},t)] - f[\phi(x_{i-1/2},t)]\}\,\mathrm{d}t + \frac{1}{\Delta x_i}\int_{t_n}^{t_{n+1}} [S(x_{i+1/2},t) - S(x_{i-1/2},t)]\,\mathrm{d}t$$

Assuming a viscous dissipative source $S = v\phi_x$ (v is the kinematic viscosity) and using Equation (3) we obtain

$$\Phi_i^{n+1} = \Phi_i^n - \frac{\Delta t}{\Delta x_i}[(\mathrm{Flux})_{i+1/2}^n - (\mathrm{Flux})_{i-1/2}^n] + v\frac{1}{\Delta x_i}\int_{t_n}^{t_{n+1}} [\phi_x(x_{i+1/2},t) - \phi_x(x_{i-1/2},t)]\,\mathrm{d}t \quad (4)$$

where $(\mathrm{Flux})_{i+1/2}^n \approx \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} f[\phi(x_{i+1/2},t)]\,\mathrm{d}t$ is some approximation of the average flux (described later in this section) along the cell interface at $x_{i+1/2}$, see Figure 1 for an illustration of the grid cells.

2.1. MUSCL limiters

Within each cell if we consider a piecewise constant approximation to the solution (i.e. slope of the reconstruction is equal to zero), then we obtain a first-order method; however if we

use a piecewise linear approximation within each cell, $\mathscr{C}_i$

$$\phi(x \in [x_{i-1/2}, x_{i+1/2}]) = \Phi_i + \Delta\Phi_i(x - x_i)$$

where Φ_i is given by Equation (3), x_i is the coordinate of the ith cell centre and $\Delta\Phi_i$ is equal to the difference between the values of the state at the right and left cell interfaces (it denotes the slope of reconstructed solution in each cell), we obtain a family of second-order approximate schemes.

Conservation laws such as the Euler equations in gas dynamics [10] and the simple Burgers equation (1) support solutions that have discontinuities (or, shocks), expansion fans, contact discontinuities. Apart from ensuring satisfaction of the Courant–Friedrichs–Lewy (CFL) condition [1], unless special treatment is taken, the numerical solutions will lead to excessive dissipation, incorrect phase speeds, spurious oscillations; see Reference [10] for an extensive comparison of many numerical methods applied to solve simple linear and nonlinear advection and Euler equations.

One way to prevent such spurious oscillations and preserve TVD [3, 43, 44] property is by limiting the values of the slopes ($\Delta\Phi_i$). Lin *et al.* [14] listed a number of consistent ways of deriving the limited slopes in various forms and compared their impact on the solution of linear advection equation. We will follow their approach for arriving at various formulations of the slope (now onwards we will assume an uniform grid, i.e. $\Delta x_i = \Delta x \ \forall i$).

1. *Limiter* 1 (first-order scheme):

$$\Delta\Phi_i^n \equiv 0 \quad \forall i \tag{5}$$

2. *Limiter* 2 (unconstrained van Leer scheme):

$$[\Delta\Phi_i^n]_{\text{avg}} = \frac{1}{\Delta x}\frac{\delta\Phi_{i-1/2}^n + \delta\Phi_{i+1/2}^n}{2} \tag{6}$$

   where $\delta\Phi_{i+1/2}^n = \Phi_{i+1}^n - \Phi_i^n$ and 'avg' means the averaging operator in the above equation. This provides us a simple second-order accurate scheme, but the values of the slopes are not limited, in other words, no limiter has yet been applied.

3. *Limiter* 3 (simple positive definite scheme):

$$[\Delta\Phi_i^n] = \frac{1}{\Delta x}\,\text{SIGN}([\Delta\Phi_i^n]_{\text{avg}}) \cdot \text{MIN}[|[\Delta\Phi_i^n]_{\text{avg}}|, 2\,\text{DIM}(\Phi_i^n, \Phi^{\min})] \tag{7}$$

   the value of the slope has been limited using the least value (over all of x_i) of Φ_i^n and $[\Delta\Phi_i^n]_{\text{avg}}$. $\text{DIM}(p,q)$ is defined as the positive difference between p and q

$$\text{DIM}(p,q) = \begin{cases} p - q & \text{if } p > q \\ 0 & \text{otherwise} \end{cases}$$

4. *Limiter* 4 (monotonicity preserving scheme):
   Another form of slope limiter which ensures monotonicity, suggested by van Leer [16, 45] is as follows:

$$[\Delta\Phi_i^n] = \begin{cases} \dfrac{1}{\Delta x}[\delta\Phi_{i-1/2}^n \cdot \delta\Phi_{i+1/2}^n]/[\Delta\Phi_i^n]_{\text{avg}} & \text{if } \text{SIGN}(\delta\Phi_{i-1/2}^n) = \text{SIGN}(\delta\Phi_{i+1/2}^n) \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

5. *Limiter* 5 (constrained van Leer scheme):
We can determine locally defined minimum and maximum values of the solution as

$$\begin{aligned}\Phi_i^{\min} &= \mathrm{MIN}[\Phi_{i-1}^n, \Phi_i^n, \Phi_{i+1}^n] \\ \Phi_i^{\max} &= \mathrm{MAX}[\Phi_{i-1}^n, \Phi_i^n, \Phi_{i+1}^n]\end{aligned} \tag{9}$$

and use them to limit the value of the slope as follows [16, 45] (also ensures monotonicity):

$$[\Delta\Phi_i^n] = \frac{1}{\Delta x}\mathrm{SIGN}([\Delta\Phi_i^n]_{\mathrm{avg}}) \cdot \mathrm{MIN}[|[\Delta\Phi_i^n]_{\mathrm{avg}}|, 2\,\mathrm{DIM}(\Phi_i^n, \Phi_i^{\min}), 2\,\mathrm{DIM}(\Phi_i^{\max}, \Phi_i^n)] \tag{10}$$

6. *Limiter* 6 (global min/max slope limited scheme):
In the above formulation of the limiter, we used the locally computed minimum and maximum values of the solution. Instead if the global minimum and maximum values of Φ_i^n are set to be equal to $\Phi_{\mathrm{global}}^{\min}$ and $\Phi_{\mathrm{global}}^{\max}$, respectively, and replacing these in above limiter formulation, we obtain

$$[\Delta\Phi_i^n] = \frac{1}{\Delta x}\mathrm{SIGN}([\Delta\Phi_i^n]_{\mathrm{avg}}) \cdot \mathrm{MIN}[|[\Delta\Phi_i^n]_{\mathrm{avg}}|, 2\,\mathrm{DIM}(\Phi_i^n, \Phi_{\mathrm{global}}^{\min}), 2\,\mathrm{DIM}(\Phi_{\mathrm{global}}^{\max}, \Phi_i^n)] \tag{11}$$

We will now use these values of slopes and follow the approach of essentially nonoscillatory (ENO) schemes to arrive at an expression for the flux at the cell interfaces.

2.2. *ENO flux*

To calculate the flux at the right cell face $x_{i+1/2}$, we used the ENO [10, 46–48] flux formulation. Using the ith and $i+1$ cell reconstructed values evaluated at $x_{i+1/2}$ (see Reference [10, Chapter 23] for details), we obtain

$$(\mathrm{Flux})_{i+1/2}^n = f^G\left[\left\{\Phi_i^n + \frac{\Delta\Phi_i^n \Delta x}{2}\left(1 - \frac{\Delta t}{\Delta x}\Phi_i^n\right)\right\}, \left\{\Phi_{i+1}^n - \frac{\Delta\Phi_{i+1}^n \Delta x}{2}\left(1 + \frac{\Delta t}{\Delta x}\Phi_{i+1}^n\right)\right\}\right] \tag{12}$$

where

$$f^G[\Phi_i^n, \Phi_{i+1}^n] = \begin{cases} \mathrm{MIN}[f(\Phi_i^n), f(\Phi_{i+1}^n), f(\Phi_*)] & \text{if } \Phi_i^n \leqslant \Phi_{i+1}^n \\ \mathrm{MAX}[f(\Phi_i^n), f(\Phi_{i+1}^n), f(\Phi_*)] & \text{if } \Phi_i^n > \Phi_{i+1}^n \end{cases} \tag{13}$$

where Φ_* is such that the flow speed given by $\partial f/\partial\Phi = \partial\frac{\Phi^2}{2}/\partial\Phi = \Phi = \Phi_* = 0$.

Remark
If the slope in each cell is equal to zero, as in Equation (5), then the above ENO flux form reduces to Godunov flux form [49].

Instead of using a piecewise linear reconstruction within each cell, we can as well apply the piecewise parabolic reconstruction approach of Colella and Woodward [50–52] within each cell.

2.3. PPM reconstruction

We have applied the PPM to reconstruct the state within each cell and to obtain the values of the state at left and right cell interfaces.

$$\phi(x \in [x_{i-1/2}, x_{i+1/2}]) = \Phi_{\mathrm{L},i} + x[\Delta\Phi_i + \Phi_{6,i}(1-x)]$$

$\Phi_{\mathrm{L},i}$ and $\Phi_{\mathrm{R},i}$ are approximations of the state at the left and right cell interface, as in MUSCL piecewise linear extrapolation, $\Delta\Phi_i = \Phi_{\mathrm{R},i} - \Phi_{\mathrm{L},i}$ and $\Phi_{6,i} = 6(\Phi_i - \frac{1}{2}(\Phi_{\mathrm{L},i} + \Phi_{\mathrm{R},i}))$ for details of the above reconstruction procedure, see Reference [50].

The fluxes at the interfaces have been directly evaluated using the calculated values, $\Phi_{\mathrm{L},i}$ and $\Phi_{\mathrm{R},i}$ for every ith cell. We have used a second-order Runge–Kutta (R–K) explicit scheme to integrate in time, described below.

2.4. Integration in time using a second-order optimal TVD R–K method

Using Equations (12) and (13) or the PPM scheme for calculating the flux and forward differencing for the diffusion term, we can write the following simple forward Euler update formula for Φ_i^{n+1}:

$$\Phi_i^{n+1} = \Phi_i^n - \frac{\Delta t}{\Delta x}[(\mathrm{Flux})_{i+1/2}^n - (\mathrm{Flux})_{i-1/2}^n] + \nu\frac{\Delta t}{\Delta x^2}[\Phi_{i+1}^n - 2\Phi_i^n + \Phi_{i-1}^n] \tag{14}$$

The above numerical scheme is at-least second-order accurate (MUSCL schemes: (6)–(9) second-order, whereas PPM being third-order accurate) in space for sufficiently smooth ϕ ($\phi \in \mathbb{C}^2$), but it is only first-order accurate in time, also it does not preserve the TVD property for time integration. In order to overcome these drawbacks, we used a second-order (accurate in time) optimal TVD R–K scheme [53, 54], given by Gottlieb and Shu [55]. Following their notation, let

$$L(\Phi_i^n) = -\frac{1}{\Delta x}[(\mathrm{Flux})_{i+1/2}^n - (\mathrm{Flux})_{i-1/2}^n] + \nu\frac{1}{\Delta x^2}[\Phi_{i+1}^n - 2\Phi_i^n + \Phi_{i-1}^n]$$

then the following sequence of two steps gives us:

$$\Phi_i^{(1)} = \Phi_i^n + \Delta t\, L(\Phi_i^n)$$

$$\Phi_i^{n+1} = \tfrac{1}{2}\Phi_i^n + \tfrac{1}{2}\Phi_i^{(1)} + \tfrac{1}{2}\Delta t\, L(\Phi_i^{(1)})$$

This completes the description of discretization in space (1-D) and time. We have tested these various finite volume methods using the aforementioned advection schemes. Comparison of the numerical results with the exact solution is provided for the following test cases (see Section 3).

2.5. Extension to 2-D: global shallow water equations model

The shallow water (SW) equations on the sphere describe the motion of a shallow (horizontal scales of interest are much larger in comparison to the depth of the fluid) homogeneous incompressible and inviscid fluid layer. The solutions of these equations exhibit some of the important properties of large scale atmospheric flow and the equations have certain important features (such as, horizontal dynamical aspects) in common with more complicated NWP models. NWP models couple such shallow water models vertically, using pressure as the vertical coordinate, see for e.g. References [17, 56] for details. The SW equations in spherical coordinates in the vorticity divergence form can be written as

$$\frac{\partial h}{\partial t} + \nabla \cdot (\mathbf{V}h) = 0 \tag{15}$$

$$\frac{\partial u}{\partial t} = \Omega v - \frac{1}{a\cos\theta}\frac{\partial}{\partial \lambda}[\kappa + \varphi] \tag{16}$$

$$\frac{\partial v}{\partial t} = -\Omega u - \frac{1}{a}\frac{\partial}{\partial \theta}[\kappa + \varphi] \tag{17}$$

where h represents the fluid height (above the surface height, h_s), $\mathbf{V} = (u, v)$, u and v represent the zonal (λ: longitude) and meridional (θ: latitude) wind velocity components, respectively, ω is the angular velocity of the earth, a is the radius of the earth. The free surface potential is given by

$$\varphi = \varphi_s + gh$$

$\varphi_s = gh_s$, $\kappa = \frac{1}{2}\mathbf{V} \cdot \mathbf{V}$ is the kinetic energy, and $\Omega = 2\omega \sin\theta + \nabla \times \mathbf{V}$ is the absolute vorticity. Details on the other forms of writing the SW equations and their development can be found in References [57, 58].

The finite volume shallow water equations model of Lin and Rood [18] has been used for integrating the above SW equations. The 1-D advection schemes described thus-far have been implemented in two dimensions by using a sequential operator-split approach, details of which have been provided in Reference [15]. A two grid combination based on C-grid and D-grids has been used while advancing from time step t_n to $t_n + \Delta t$. In the first half of the time step, the advective winds (time centred winds on the C-grid: (u^*, v^*) are updated on the C-grid, and in the other half of the time step, the prognostic variables (h, u, v) are updated on the D-grid (in this study, we will use the same advection scheme on both the grids). The poles have been treated in a similar fashion as that in Reference [59] using a polar Fourier filter.

In particular, it is to be noted that the algorithm conserves total mass (in other words, the height field, h, integrated on the surface of the sphere) for all the time of the numerical integration and, after a 60 day integration of the model, the loss in total energy (total energy is defined as the integral of $\frac{1}{2}h\mathbf{V} \cdot \mathbf{V} + \frac{1}{2}g[(h + h_s)^2 - {h_s}^2]$ on the surface of the sphere) is approximately lost by 0.1%, and the loss in potential enstrophy (potential enstrophy is the integral of $(1/2h)\Omega^2$) is 1%. More details are available in Reference [18]. This represents excellent conservation of integral invariants of the shallow water equations.

3. TEST CASES AND RESULTS

3.1. Case 1: 1-D viscous Burgers equation

In 1-D we will consider the following Burgers equations:

$$\frac{\partial \phi}{\partial t} + \frac{\partial}{\partial x}\left(\frac{\phi^2}{2}\right) = \nu\left(\frac{\partial^2 \phi}{\partial x^2}\right) \tag{18}$$

for $x \in (-\pi, \pi)$ and $t > 0$, with boundary conditions, $\phi(x = \pm\pi, t) = 0$.

Benton and Platzman [60] provide an exact solution for the above Burgers equation (18), with initial condition given by

$$\phi(x, 0) = \phi(x, t = 0) = -R\sin(x) \tag{19}$$

where R is the Reynolds number. It is related to the viscosity via the relationship, $R = UL/\nu$, here the values of (velocity scale) U and (length scale) L have been prescribed to be equal to unity. Then the exact solution assumes the form

$$\phi_{\text{exact}}(x, t) = \frac{4\sum_{n=1}^{\infty} n a_n e^{-n^2 t}\sin(nx)}{a_0 + 2\sum_{n=1}^{\infty} a_n e^{-n^2 t}\cos(nx)} \tag{20}$$

where $a_n = (-1)^n I_n(\frac{1}{2}R)$, I_n is the Bessel function of second kind. For small values of R, viscous dissipation dominates over advection and the solution decays uniformly as time, t increases, as depicted in Figure 2 (which has been generated by setting $R = 1$).

The first-order scheme (limiter 1) and limiter 3 (simple positive definite scheme, which was based on limiting the slope based on the least value of Φ_i^n and $[\Delta\Phi_i^n]_{\text{avg}}$), both undershoot at the peak value of the numerical solution at $x = \pm\pi/2$. It is to be noted that all the numerical solutions have the correct phase speed. In the case of the global min/max limiter 6, we prescribed $\Phi^{\min} = -1$ and $\Phi^{\max} = 1$.

In Table I we show that the numerical solutions converge to the exact solution in both L_2 and L_∞ norms, at $t = 1$. As expected the first-order scheme (limiter 1) has the largest error compared to all other schemes. Lin *et al.* [14] compared limiters $3, 4, 5$ and 6 on a linear advection problem using a rectangular pulse. Based on their study, they concluded that limiter 4 provides the largest implicit diffusion among all the limiters considered, whereas limiter 2 provides the smallest implicit diffusion and the constrained van Leer scheme (limiter 5) is less diffusive than limiter 4.

3.2. Case 2: 1-D inviscid Burgers equation

To further investigate the performance of these limiters on a model problem with no viscosity, we tested them using the following inviscid nonlinear Burgers equation:

$$\frac{\partial \phi}{\partial t} + \frac{\partial}{\partial x}\left(\frac{\phi^2}{2}\right) = 0$$

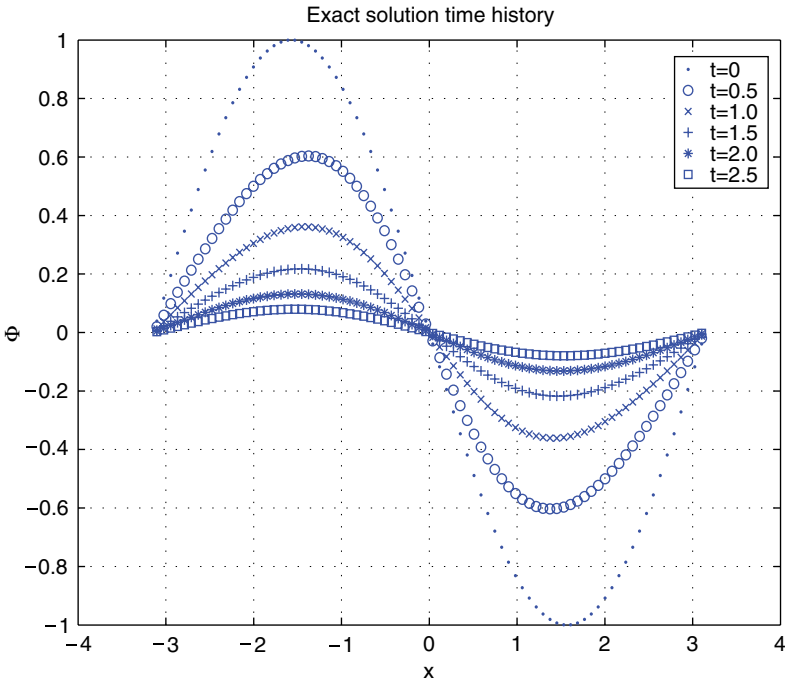

Figure 2. Exact solution at $t = 0, 0.5, 1.0, 1.5, 2.0, 2.5$.

Table I. Errors in L_2 and L_∞ norms for different advection schemes (based on changing the slope limiter, lim.1 indicates limiter 1) in forward mode, with $\Delta t = 1.5708 \times 10^{-3}$ at $t = 1$.

L_2-Error							
N_x	lim.1 $\times 10^{-2}$	lim.2 $\times 10^{-3}$	lim.3 $\times 10^{-3}$	lim.4 $\times 10^{-3}$	lim.5 $\times 10^{-3}$	lim.6 $\times 10^{-3}$	PPM $\times 10^{-3}$
40	3.1357	1.39420	1.39419	1.4882	1.5252	1.3942	1.308409
80	2.2142	0.852385	0.852384	0.79215	0.83126	0.852384	1.037877
160	1.5716	0.70370324	0.70370321	0.69061	0.69817	0.70370317	0.81714
L_∞-Error							
N_x	lim.1 $\times 10^{-3}$	lim.2 $\times 10^{-3}$	lim.3 $\times 10^{-3}$	lim.4 $\times 10^{-3}$	lim.5 $\times 10^{-3}$	lim.6 $\times 10^{-3}$	PPM $\times 10^{-3}$
40	8.0511	0.3200545	0.3200543	0.32797	0.34984	0.32004	0.3833
80	4.0426	0.17541576	0.17541570	0.16975	0.17335	0.1754155	0.1965
160	2.0321	0.087102731	0.087102728	0.086004	0.086741	0.087102723	0.0975

With the following initial condition (on the whole real line) [61]:

$$\phi(x,0) = \begin{cases} 0, & x < -1 \\ \frac{1}{2}, & -1 < x < 0 \\ 0, & x > 0 \end{cases} \tag{21}$$

The solution develops into a shock and an expansion fan (for details of the solution, see Reference [61]), analytically given by (for $t \leqslant 4$, i.e. before the expansion fan meets the shock)

$$\phi(x,t) = \begin{cases} 0, & x < -1 \\ \dfrac{x+1}{t}, & -1 < x < \dfrac{t}{2} - 1 \\ \dfrac{1}{2}, & \dfrac{t}{2} - 1 < x < \dfrac{t}{4} \\ 0, & x > \dfrac{t}{4} \end{cases} \tag{22}$$

As expected, the first-order accurate scheme is diffusive. The solutions obtained by using limiter 2 (unconstrained van Leer scheme) and limiter 3 (simple positive definite scheme), both over shoot, indicating that there is a lack of (implicit) viscosity. Though the solution obtained by using limiter 4 (monotonicity preserving scheme) does not suffer from such problems, it is diffusive, when compared to the computed solutions using limiters 5, 6 (constrained van Leer and global min/max slope limited schemes, respectively) and the PPM scheme.

Following arguments in Section 2 of Reference [15], limiter 3 (positive definite scheme) does not provide satisfactory solutions to 2-D tracer advection equation. Also it requires specification of the minimum values of the solution *a priori*, the same being the case with the global min/max scheme (limiter 6) which requires specification of both minimum and maximum values of the solution *a priori*, which is not accurately possible for complex higher dimensional flows. Limiter 5 (constrained van Leer scheme) has been shown to be better than limiter 4 (monotonicity preserving scheme) in Reference [14] due to the fact than limiter 5 provides less implicit diffusion than limiter 4. Therefore limiter 1 (first-order scheme), limiter 2 (unconstrained van Leer scheme), limiter 5 (constrained van Leer scheme) and PPM advection schemes are of interest to global NWP modelers, such as in Reference [17] (also see the documentation of the *Community Atmosphere Model 3.0* [62]) hence we will restrict our 2-D study to only these schemes.

3.3. Case 3: 2-D global SW equations

The development of a numerical solver for the global spherical SW equations is usually a first step towards the development of a NWP model. A suite of several test cases that have been widely used to compare different algorithmic formulations and numerical schemes for the SW equations was suggested by Williamson *et al.* [57]. Therefore, results obtained from these tests could be used as a guide towards developing more complex models in higher dimensions. The test case number 6, is the Rossby–Haurwitz wave (wavenumber 4), first proposed by Phillips [63]. Although analytical solutions for this case in the global SW equations context are not known, it is a very popular test case in the NWP modelling community for a number of reasons. Haurwitz [64] showed that the Rossby–Haurwitz waves are analytic solutions of the nonlinear barotropic vorticity equation on the sphere. They are characterized by a pattern which moves from west to east without any change in shape.

Figures 3–10 provide results obtained by integration of the finite volume SW equations model of Reference [18] using the different advection schemes for 14 and 30 days, respectively (the initial condition was specified to be a Rossby–Haurwitz wave). The resolution of the

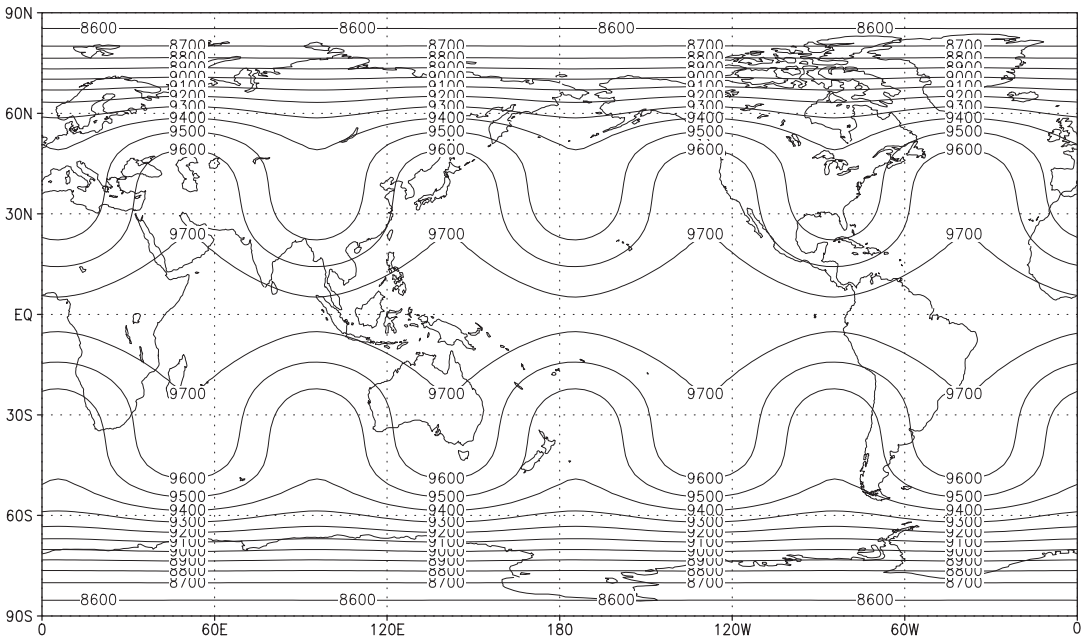

Figure 3. Height field isolines at Day-14 using the first-order advection scheme (lim.1), for the Rossby–Haurwitz wavenumber 4 case using a finite volume global SW equations model. Contour interval is 100 m.

model is the same as in Reference [18], 128 grid cells along the longitude and 64 along the latitude, and a time step of 600 s. The DAY-14 solution in the case of constrained van Leer and PPM schemes is similar, whereas the DAY-30 solution obtained by using the constrained van Leer is more diffused than that of the PPM scheme. The first-order advection scheme is extremely dissipative, as evident from Figures 3 and 4. Therefore, for 2-D DA experiments we will not be using the first-order advection scheme. The unconstrained van Leer scheme is certainly less dissipative than the first-order scheme, but more diffusive when compared to the constrained van Leer and PPM schemes.

4. DERIVATION AND VERIFICATION OF THE ADJOINT AND TANGENT LINEAR MODELS

This section details the derivation of the development of the adjoint method, aimed at obtaining the gradient of the cost functional with respect to the control parameters efficiently, closely following [65].

The following form of the cost functional is considered:

$$\mathscr{J}(\mathbf{x}) = \frac{1}{2} \sum_{k=0}^{n} (\mathbf{x}(t_k) - \mathbf{x}^{\text{obs}}(t_k))^{\text{T}} W(t_k) (\mathbf{x}(t_k) - \mathbf{x}^{\text{obs}}(t_k)) \tag{23}$$

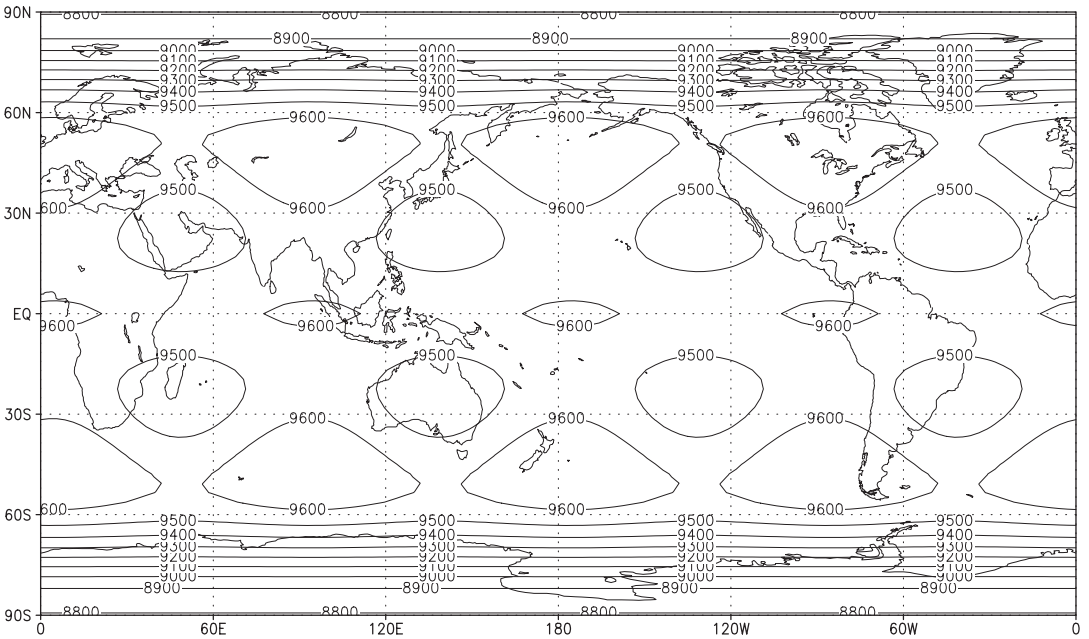

Figure 4. Same as Figure 3, but at Day-30.

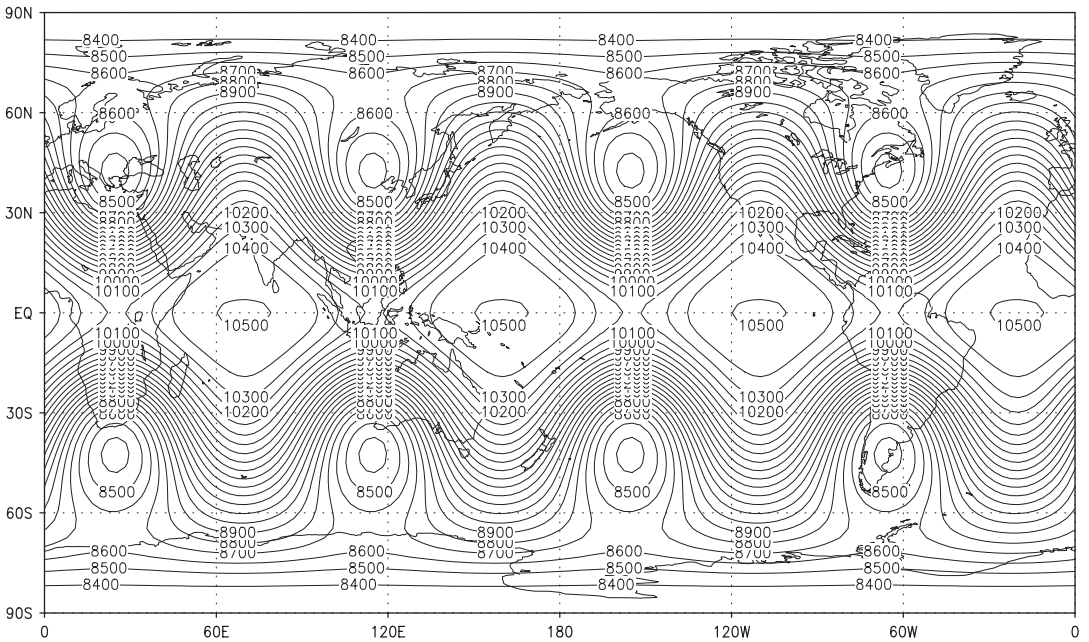

Figure 5. Same as in Figure 3, but using unconstrained van Leer scheme (lim.2).

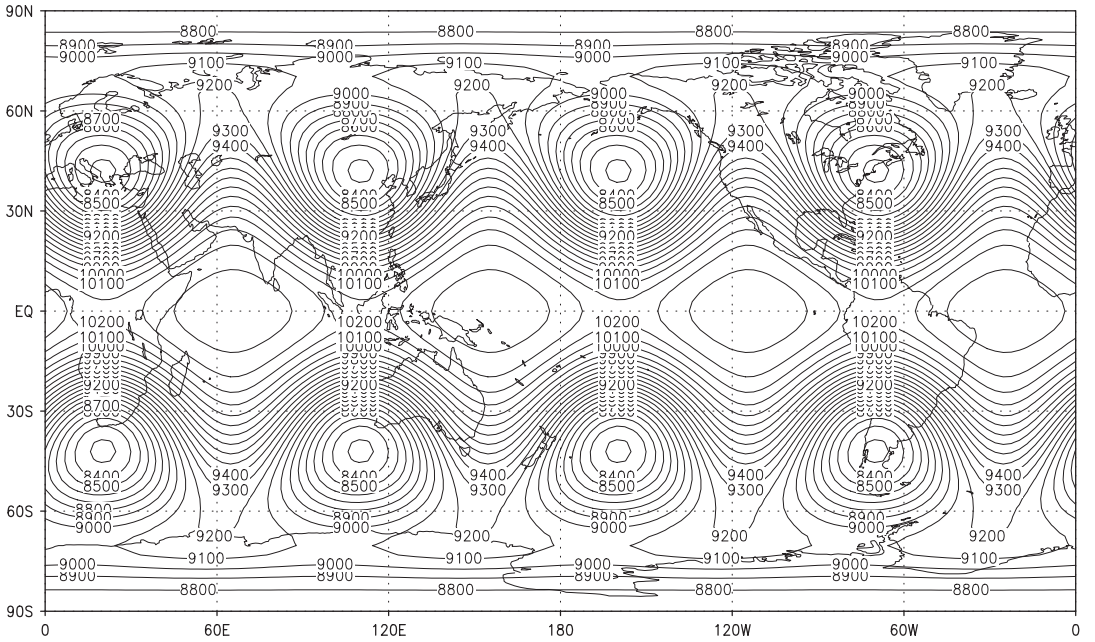

Figure 6. Same as Figure 5, but at Day-30.

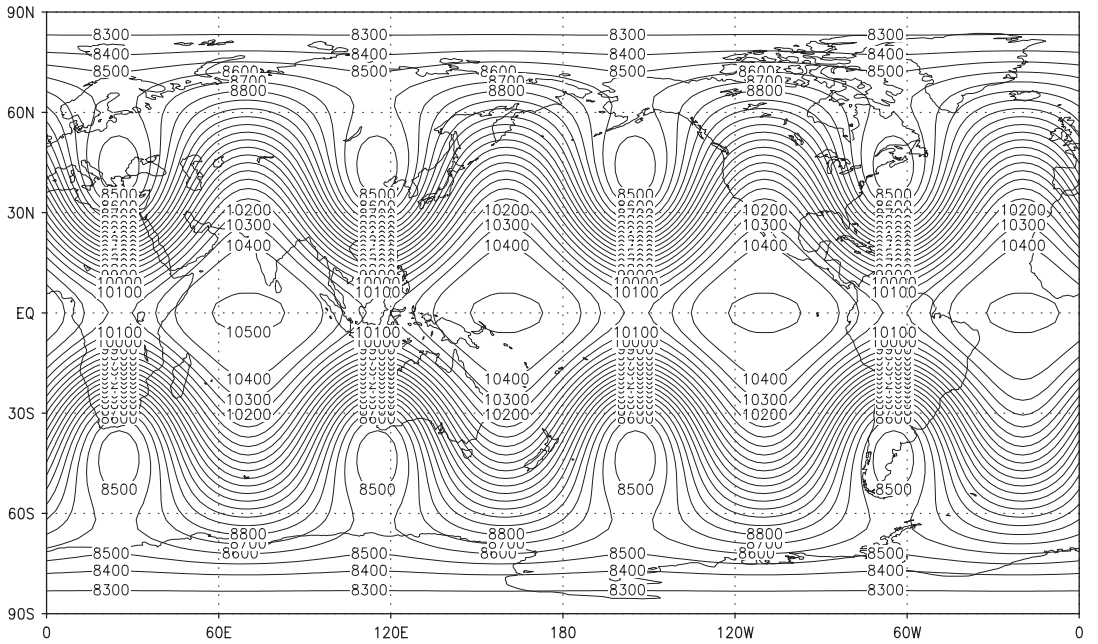

Figure 7. Same as in Figure 3, but using constrained van Leer scheme (lim.5).

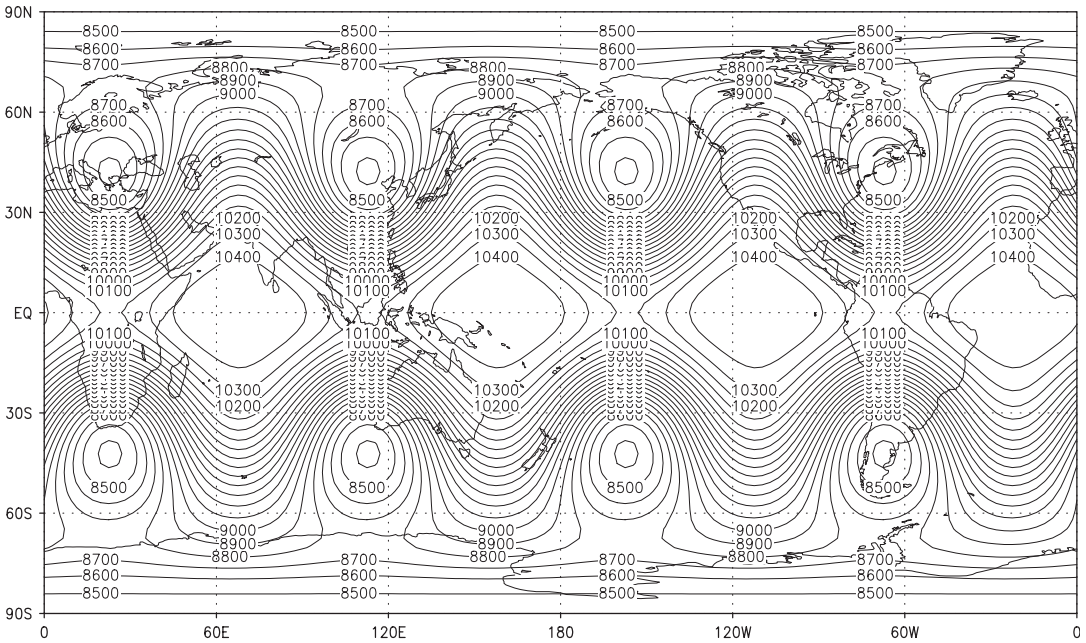

Figure 8. Same as Figure 7, but at Day-30.

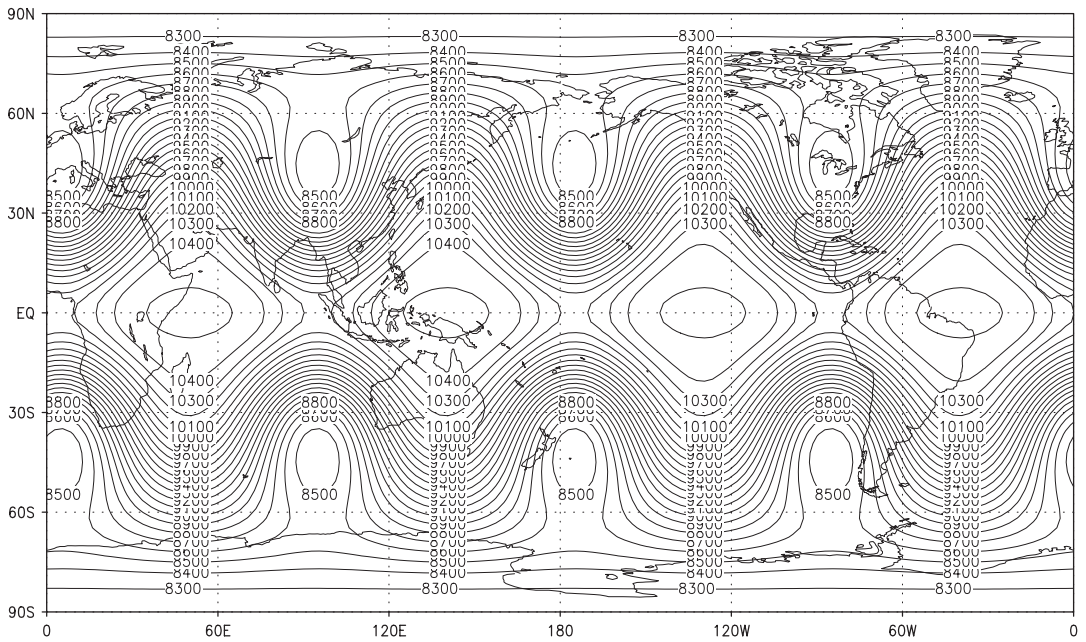

Figure 9. Same as in Figure 3, but using PPM advection scheme.

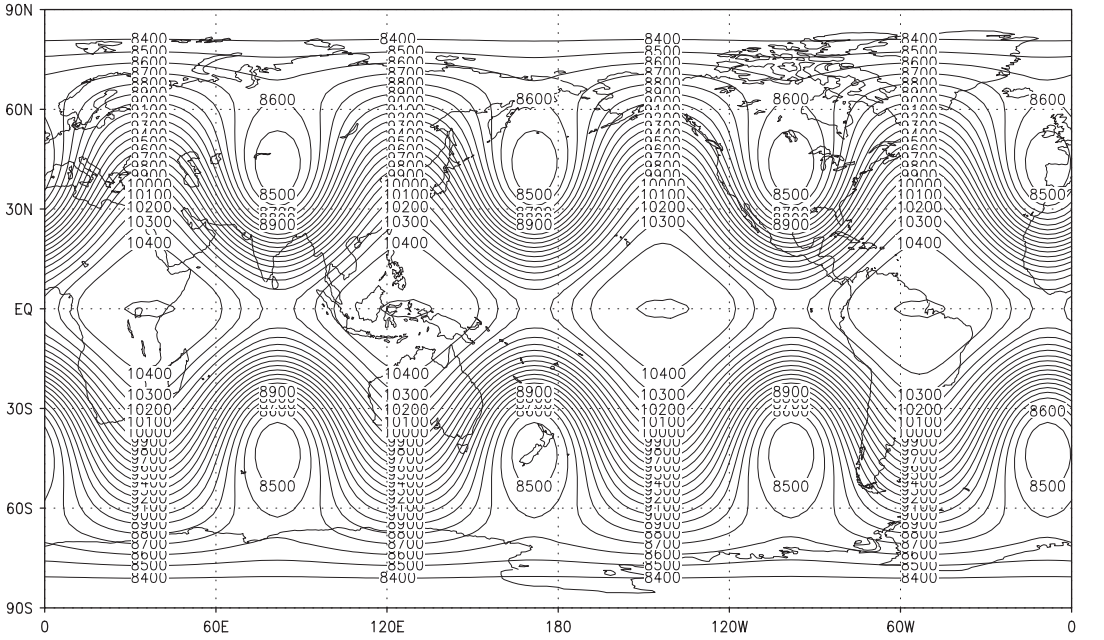

Figure 10. Same as Figure 9, but at Day-30.

where $t \in [t_0, t_n]$ is the (data) assimilation time window comprised of n time steps, $W(t_k)$ is a diagonal weighting matrix, $\mathbf{x}(t_k)$ is the evolving state vector and $\mathbf{x}^{\text{obs}}(t_k)$ is another (evolving) vector, which is made up of the observations that are distributed in both space and time.

The above convex cost functional is minimized (subject to the evolution of the state vector by the nonlinear model as a strong constraint) using a robust unconstrained minimization method described in Section 5. The directional derivative of the above cost functional, in the direction of $\delta\mathbf{x}$ is given by $(\nabla_{\mathbf{x}}\mathscr{J})^{\mathrm{T}}\delta\mathbf{x}$.

From Equation (23)

$$\delta\mathscr{J}(\mathbf{x}) = \sum_{k=0}^{n} (W(t_k)(\mathbf{x}(t_k) - \mathbf{x}^{\text{obs}}(t_k)))^{\mathrm{T}}\delta\mathbf{x}(t_k) \tag{24}$$

where $\delta\mathbf{x}(t_k)$ is the perturbation of the state vector obtained from the perturbation of the model parameters, $\mathbf{x}$. Using the above two equations

$$(\nabla_{\mathbf{x}}\mathscr{J})^{\mathrm{T}}\delta\mathbf{x} = \sum_{k=0}^{n} (W(t_k)(\mathbf{x}(t_k) - \mathbf{x}^{\text{obs}}(t_k)))^{\mathrm{T}}\delta\mathbf{x}(t_k) \tag{25}$$

The evolution of the state vector using the nonlinear model can be symbolically written as

$$\mathbf{x}(t_{k+1}) = F(\mathbf{x}(t_k))$$

Linearizing the model about the current model solution, we obtain the equation for the evolution of perturbations

$$\delta\mathbf{x}(t_{k+1}) = \frac{\partial F(\mathbf{x}(t_k))}{\partial \mathbf{x}}\delta\mathbf{x} \tag{26}$$

Let $L(t)$ represent the Jacobian, $\partial F(\mathbf{x}(t_k))/\partial\mathbf{x}$, then we can rewrite the above equation as

$$\begin{aligned}\delta\mathbf{x}(t_k) &= L(t_k - \Delta t)\delta\mathbf{x}(t_k - \Delta t)\\ &= L(t_k - \Delta t)L(t_k - 2\Delta t)\delta\mathbf{x}(t_k - 2\Delta t)\\ &= L(t_k - \Delta t)L(t_k - 2\Delta t)L(t_k - 3\Delta t)\delta\mathbf{x}(t_k - 3\Delta t)\\ &= \cdots\\ &= M_k\delta\mathbf{x}\end{aligned} \tag{27}$$

Thus $\delta\mathbf{x}(t_k) = M_k\delta\mathbf{x}$, where M_k represents the application of all the linear operators to obtain $\delta\mathbf{x}(t_k)$.

Using Equations (25) and (27) the gradient of the cost functional with respect to the control parameters, $\mathbf{x}$ is given by

$$\nabla_{\mathbf{x}}\mathscr{J} = \sum_{k=0}^{n} M_k^{\mathrm{T}} W(t_k)(\mathbf{x}(t_k) - \mathbf{x}^{\mathrm{obs}}(t_k)) \tag{28}$$

Using $W(t_k)(\mathbf{x}(t_k) - \mathbf{x}^{\mathrm{obs}}(t_k))$ as the (initial) values of the adjoint variables $\mathbf{x}^*(t_k)$ at time t_k, the adjoint equations

$$\mathbf{x}^*(t_0) = M_k^{\mathrm{T}}(\mathbf{x}^*(t_k)) \tag{29}$$

are integrated backwards in time to obtain the values of the adjoint variables at initial time, t_0. Therefore,

$$\nabla_{\mathbf{x}}\mathscr{J} = \sum_{k=n}^{0} \mathbf{x}^*(t_k) \tag{30}$$

Now we will briefly describe the method of programming the adjoint model, in other words, Equations (27) and (29) and implementation of (30).

4.1. Coding the adjoint model

We follow the approach of first discretize and then differentiate (see References [28, 32] for details). Discrete numerical operations in the nonlinear forward model are having unique corresponding operations in the adjoint model. The linear equation (26) is now onwards referred to as the Tangent Linear Model (TLM). The TLM code is programmed by linearizing line by line, the nonlinear forward model code. Following Equation (27), the TLM can be formally viewed as a result of multiplying linear operators: $M_k = L_1, L_2, \ldots, L_k$, where each of the L_k is either a DO-loop or a subroutine in the TLM. Then the adjoint model, M_k^{T} is a product of

the (adjoint) linear operators, $L_k^{\mathrm{T}}, L_{k-1}^{\mathrm{T}}, \ldots, L_2^{\mathrm{T}}, L_1^{\mathrm{T}}$. Hence the adjoint model is the *transpose* of the TLM. This relationship is used to write the adjoint model code, using the TLM code (see References [30, 66] for details), and to verify the same for the transposition property (all our subroutines satisfactorily passed this test). We have used TAMC [67–69] (an automatic differentiation software) to help us derive the TLM and adjoint model codes; however, we would like to emphasize that sufficient caution must be taken while differentiating functions such as the ABS (absolute value function), SIGN (signum function), DIM (dimension function), MIN and MAX (minimum and maximum functions, respectively), these functions frequently arise due to the nature of the formulation of the various slope limiters, such as limiters 3, 5 and 6 (Section 2.1). In Appendix A, we provide a segment of our FORTRAN code which illustrates the differentiation of the MIN function.

The adjoint model is integrated backwards in time to obtain the gradient of the cost functional, $\nabla_{\mathbf{x}} \mathscr{J}$ in the following sequence of three steps:

1. Integrate the adjoint model backwards in time, from time step t_k to t_0 with zero final conditions for the adjoint variables $\mathbf{x}^*$.
2. The *forcing term* $W(t_k)(\mathbf{x}(t_k) - \mathbf{x}^{\mathrm{obs}}(t_k))$ is added to the value of adjoint variables whenever time t_k $(k = 1, 2, \ldots, n)$ is reached.
3. Finally, at t_0 the value of adjoint variables equals the gradient of the cost functional with respect to the control variables.

Using the Taylor series expansion of the cost functional, to first-order

$$\mathscr{J}(\mathbf{x} + \eta \nabla \mathscr{J}) = \mathscr{J}(\mathbf{x}) + \eta (\nabla \mathscr{J})^{\mathrm{T}} \nabla \mathscr{J} + O(\eta^2) \tag{31}$$

where η is a scalar and the gradient, $\nabla \mathscr{J} = \nabla_{\mathbf{x}} \mathscr{J}$, is obtained by using the adjoint model. We can rewrite the above equation as in Reference [32]

$$\Psi(\eta) = \frac{\mathscr{J}(\mathbf{x} + \eta \nabla \mathscr{J}) - \mathscr{J}(\mathbf{x})}{\eta \nabla \mathscr{J}^{\mathrm{T}} \nabla \mathscr{J}} = 1 + O(\eta) \tag{32}$$

Therefore, the gradient provided by the adjoint model is assumed to be accurate up to machine accuracy if $\lim_{\eta \to 0} \Psi(\eta) = 1.0$ The truncation errors dominate for $\eta > 10^{-3}$, whereas for η near machine precision, roundoff errors accumulate. Tables II and III provide values of $\Psi(\eta)$ versus η obtained for the adjoint model using various limiters and the PPM advection scheme case for the 1-D Burgers and 2-D global SW equations models, respectively. See Reference [70] for details of the adjoint model for the SW equations model used.

5. MINIMIZATION

We used an unconstrained limited memory quasi-Newton (L-BFGS) minimization algorithm [71, 72] (available for download at www.netlib.org/opt/lbfgs_um.shar) for minimization of the cost functional $\mathscr{J} = \mathscr{J}(\mathbf{x}_k)$, where $\mathbf{x}_k$ is the n component (control) vector at the kth iteration. $\mathbf{g}_k = \mathbf{g}(\mathbf{x}_k) = \nabla \mathscr{J}_k$ is the gradient vector of size n, and $H_k = \nabla^2 \mathscr{J}_k$ is the $n \times n$ symmetric Hessian matrix of the second partial derivatives of $\mathscr{J}$ with respect to the control vector. The new iterate is given by

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k \tag{33}$$

Table II. Gradient check: values of $\Psi(\eta)$ for different η for slope limiters and PPM advection scheme in adjoint mode for 1-D Burgers equation model.

$-\log_{10}(\eta)$	lim.1	lim.2	lim.3	lim.4	lim.5	lim.6	PPM
				$\Psi(\eta)$			
2	2.0768853	2.1463904	2.1032286	2.1440140	2.1455097	2.0595244	2.1503010
3	1.1346913	1.1427750	1.1397331	1.1426656	1.1427245	1.1366907	1.1429770
4	1.0137129	1.0144949	1.0143250	1.0144854	1.0144909	1.0141552	1.0145123
5	1.0013705	1.0014515	1.0014379	1.0014508	1.0014513	1.0014242	1.0014536
6	1.0001339	1.0001451	1.0001414	1.0001453	1.0001454	1.0001377	1.0001459
7	1.0000102	1.0000145	1.0000145	1.0000148	1.0000148	1.0000145	1.0000151
8	0.9999978	1.0000014	1.0000014	1.0000017	1.0000017	1.0000014	1.0000020
9	0.9999966	1.0000001	1.0000001	1.0000004	1.0000004	1.0000001	1.0000007
10	0.9999965	0.9999999	1.0000000	1.0000003	1.0000002	1.0000000	1.0000006
11	0.9999965	0.9999999	1.0000002	1.0000008	1.0000006	1.0000000	1.0000006
12	0.9999951	1.0000018	1.0000070	1.0000042	1.0000018	0.9999995	1.0000032

Table III. Gradient check: values of $\Psi(\eta)$ for different η for slope limiters and PPM advection scheme in adjoint mode for 2-D global SW equations model.

$-\log_{10}(\eta)$	lim.1	lim.2	lim.5	PPM
		$\Psi(\eta)$		
1	1.716810	1.894919	1.086659	1.013195
2	1.085421	1.087817	1.007975	1.003011
3	1.009237	1.008724	1.000429	1.001268
4	1.001031	1.001104	0.999922	1.000770
5	1.000126	1.000337	0.999962	1.000825
6	1.000012	1.000304	1.000000	1.000000
7	1.000000	1.000000	1.000705	0.994831
8	0.999999	1.000496	1.001122	0.941435
9	0.999999	1.000872	1.001174	0.422035
10	0.999979	1.000966	1.001313	−4.817312

where $\mathbf{p}_k$ is the descent direction (for instance, $\mathbf{p}_k = -\mathbf{g}_k$ for the steepest descent method and $\mathbf{p}_k = -H_k^{-1}\mathbf{g}_k$ for the quasi-Newton methods), and α_k is the step length.

Iterations are terminated when (using the L_2 norm)

$$\|\mathbf{g}_k\| < \text{EPS} \cdot \text{MAX}(1, \|\mathbf{x}_k\|)$$

Here we specified $\text{EPS} = 10^{-5}$ as our termination criteria.

Given a sequence of two successive iterates, $\mathbf{x}_{k+1}$ and $\mathbf{x}_k$, $\mathbf{g}_k = \nabla \mathscr{J}_k$ and $\mathbf{g}_{k+1} = \nabla \mathscr{J}_{k+1}$. Then $\mathbf{g}_{k+1} - \mathbf{g}_k = H_k \mathbf{p}_k$ which can be rewritten as $\mathbf{q}_k = H_k \mathbf{p}_k$. If the Hessian is constant, then $\mathbf{q}_k = H\mathbf{p}_k$, and we can write the following quasi-Newton condition for $0 \leqslant i \leqslant k$:

$$H_{k+1}^{-1}\mathbf{q}_i = \mathbf{p}_i$$

In general, the evaluation of the Hessian matrix is impractical and costly. Quasi-Newton methods use an approximation of the inverse Hessian matrix. We start with an identity matrix and then iteratively, a better approximation to the inverse Hessian matrix is built up, in such a way that H_k preserves positive definiteness and symmetry.

The Broyden–Fletcher–Goldfarb–Shanno (BFGS) update formula for the B_{k+1} (i.e. H_{k+1}^{-1}) is given by

$$B_{k+1} = B_k + \frac{(1 + \mathbf{q}_k^{\mathrm{T}} B_k \mathbf{q}_k)}{\mathbf{q}_k^{\mathrm{T}} \mathbf{p}_k} \frac{\mathbf{p}_k \mathbf{p}_k^{\mathrm{T}}}{\mathbf{p}_k^{\mathrm{T}} \mathbf{q}_k} - \frac{\mathbf{p}_k \mathbf{q}_k^{\mathrm{T}} B_k + B_k \mathbf{q}_k \mathbf{p}_k^{\mathrm{T}}}{\mathbf{q}_k^{\mathrm{T}} \mathbf{p}_k} \tag{34}$$

this is a symmetric rank two update, constructed using the vectors $\mathbf{p}_k$ and $B_k \mathbf{q}_k$. Thus each minimization iteration proceeds by first checking for termination criteria, finding the direction of descent: $\mathbf{p}_k$ (using the approximation to the inverse Hessian matrix), find an optimal step length (α_k) in the direction of $\mathbf{p}_k$, and finally using Equation (33) find the next $\mathbf{x}_{k+1}$. The limited memory version, L-BFGS is an adaptation of the BFGS algorithm to large problems, achieved by changing the above Hessian update formula, see for details References [71–75] for applications.

6. DATA ASSIMILATION EXPERIMENTS

This section describes results obtained using the adjoint model described in the previous section in order to conduct data assimilation for retrieval of optimal initial conditions which serve as control variables. Following work of Vukićević *et al.* [24], we have consistently used the same advection scheme both in the nonlinear forward and adjoint models. Our goal is to minimize the cost functional given in Equation (23), namely

$$\mathscr{J}(\mathbf{x}) = \frac{1}{2} \sum_{k=0}^{n} (\mathbf{x}(t_k) - \mathbf{x}^{\mathrm{obs}}(t_k))^{\mathrm{T}} W(t_k)(\mathbf{x}(t_k) - \mathbf{x}^{\mathrm{obs}}(t_k))$$

with respect to the initial state $\mathbf{x}(t_0) \equiv \phi(x, 0)$ as the control parameter and we have prescribed $W(t_k) \equiv I$, i.e. the identity matrix for the 1-D Burgers equation case. In the global 2-D SW equations model case, the control vector is given by $\mathbf{x}(t_0) = (h, u, v)$ at the initial time and $W(t_k)$ was prescribed to be equal to a block diagonal matrix with $[10^{-4}\, I, I, I]$ as the diagonal entries.

The framework of *identical twin experiment* has been used in this study, which has been frequently used to compare different methods in developmental stages. In twin experiments, observations are not obtained from reality, but are generated by using a version of the model which is slightly different to the model used in DA. Twin experiments provide a *good* diagnostic tool for determining the quality of the method, since the errors are controlled; thus those methods that perform well in twin experiments are often considered as candidates for conducting DA using real observations.

In our 1-D twin experiments, we used the initial condition given in (19), run the forward model up-to time step t_k to obtain the observations, $\mathbf{x}^{\mathrm{obs}}(t_k)$ and in the 2-D case, the Rossby wave has been used as the initial condition. The initial condition is then randomly perturbed

$$\mathbf{x}^{\mathrm{pert}}(t_0) = \mathbf{x}(t_0) + \varepsilon \cdot \mathrm{RAND} \cdot \mathbf{x}(t_0) \tag{35}$$

where ε has been assigned a value of 0.01 and RAND is a pseudo random number, such that $\text{RAND} \in [-0.5, 0.5]$.

The above perturbed initial condition is used as a first guess to minimize the cost functional, $\mathcal{J}$, and to integrate the nonlinear model to t_k, which yields $\mathbf{x}(t_k)$. Thus the goal is to recover the unperturbed initial condition, $\mathbf{x}$ (now onwards denoted by $\mathbf{x}^{\text{recovered}}$), which is close to $\mathbf{x}(t_0)$ at the conclusion of the minimization process. An assimilation time window of $[0, 2.0]$ seconds has been used in the 1-D case and in the 2-D case, the length of the assimilation window was taken to be 6 h. The same discretization, in space and time, which was used in Section 3 to test and compare the different schemes for the smooth test cases, is used here as well.

Figures 11–14 show the variation of the cost functional, $\mathcal{J}$, and gradient norm (in L_2 norm) versus the number of iterations and in Table IV we compare the values of $\mathbf{x}^{\text{recovered}}$ with $\mathbf{x}(t_0)$ for different advection schemes (limiters 1–6 and the PPM) in the 1-D case. For the 2-D case, please see Figures 15 and 16 and Table V, respectively. The cost functional has been successfully reduced by about nine orders of magnitude in the 1-D case and in the 2-D case by about four orders of magnitude. Whereas the gradient norm was reduced by about five orders of magnitude (in Section 4, we described the termination criteria for the minimization process) for all the limiters in the 1-D case, and by about three for the 2-D case. The fact that all of these schemes (in the 1-D case) achieve the same convergence criteria for successful termination in about 45–50 minimization iterations (limiter 1: 51 iterations, limiter 2: 47, limiter 3: 43, limiter 4: 42, limiter 5: 52, limiter 6: 37), except for the PPM scheme, which took 65 iterations indicates that the approximation to the Hessian matrix that is constructed by the L-BFGS minimization algorithm does not differ from one advection scheme to the other (the spectrum of the eigenvalues of the Hessian matrix influences the minimization process [73]). In the 2-D case, limiter 2: 577, limiter 5: 589 and PPM scheme took 575 minimization iterations to achieve the prescribed convergence criteria. It is to be

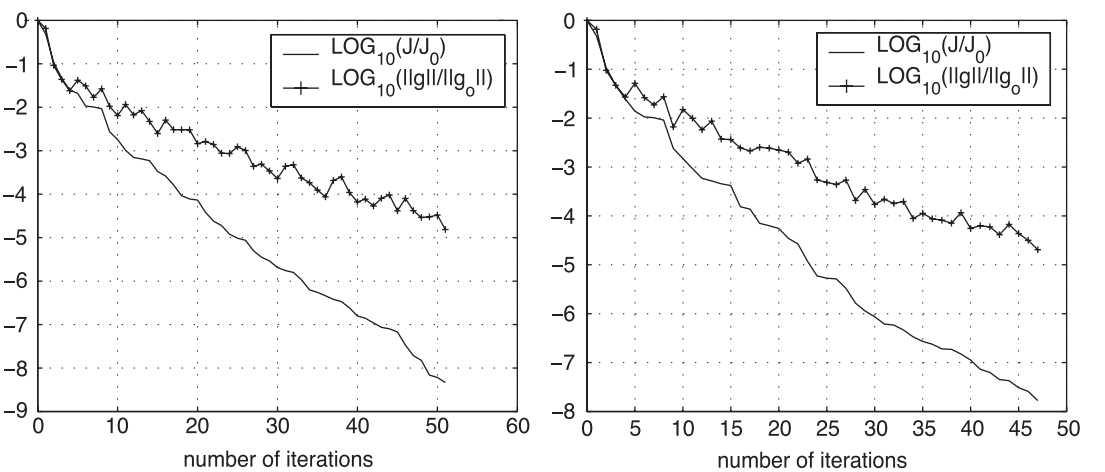

Figure 11. Variations of the normalized cost function $\mathcal{J}/\mathcal{J}_0$ and normalized gradient $\|\mathbf{g}\|/\|\mathbf{g}_0\|$ versus the number of minimization iterations using slope limiters 1 and 2 in forward and adjoint models for the 1-D Burgers equation model (in log scale).

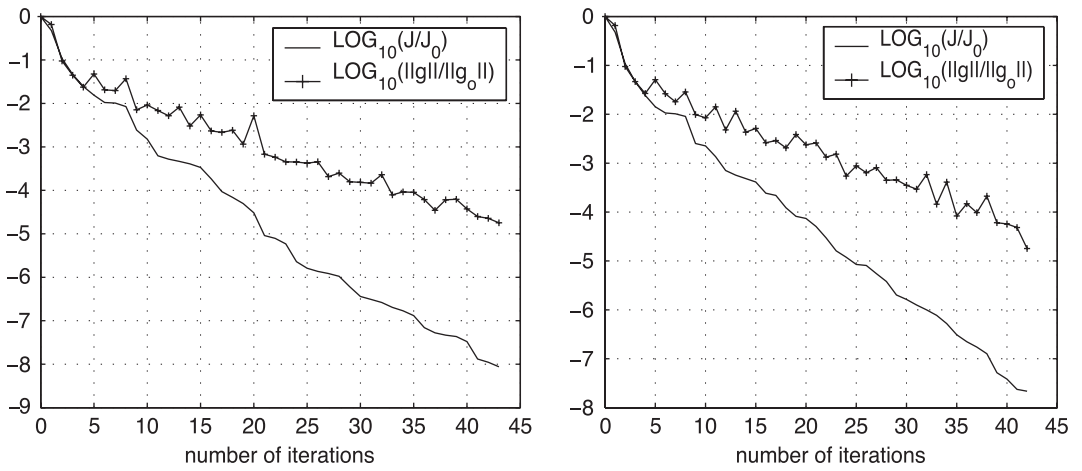

Figure 12. Same as Figure 11, but with limiters 3 and 4.

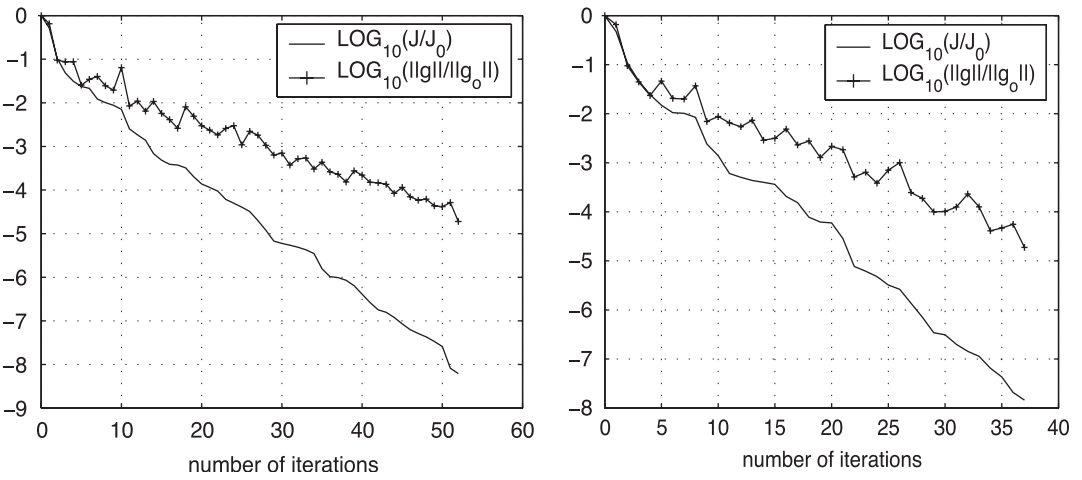

Figure 13. Same as in Figure 11, but with limiters 5 and 6.

noted that though the PPM scheme is well known to be a very accurate scheme (third-order accurate), it requires more CPU time when compared to that required by other schemes (both in forward and adjoint modes, since the adjoint model performs forward computations as well, this problem becomes compounded). We would like to mention that limiter 3 (simple positive definite scheme), the local and global min/max (limiters 5 and 6, respectively) slope limited and PPM schemes all have *switches*, in other words, involve computation of min and (or) max of certain variables to evaluate the slope limiter (see Equations (7), (9) and (11)). Programming these switches in the adjoint model proves to be a very tedious and time consuming task.

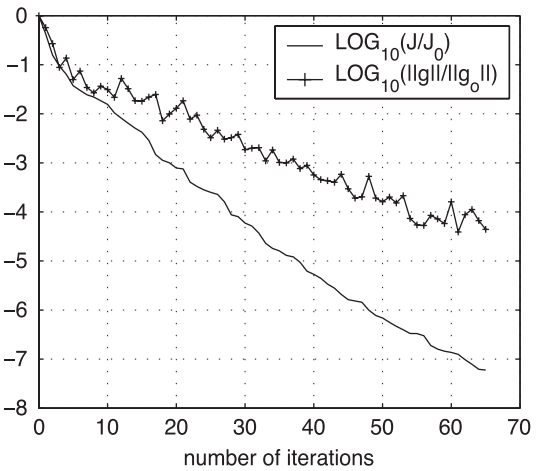

Figure 14. Same as Figure 13, but with the PPM scheme.

Table IV. Comparison of the $\phi^{\text{recovered}}$ for different advection schemes based on 1-D data assimilation experiments, $\|\phi^{\text{pert}}(x,0) - \phi(x,0)\|_2 = 1.3004 \times 10^{-2}$ for all the schemes; forecast time, $T = 2.2$ s.

Advection scheme	$\|\phi^{\text{recovered}} - \phi(x,0)\|_2$	$\|\phi^{\text{pert}}(x,T) - \phi(x,T)\|_2$	$\|\phi^{\text{recovered}}(T) - \phi(x,T)\|_2$
Lim.1	3.1063×10^{-6}	1.2362×10^{-3}	7.7479×10^{-8}
Lim.2	7.1633×10^{-6}	1.245593×10^{-3}	1.3876×10^{-8}
Lim.3	4.9715×10^{-6}	1.245587×10^{-3}	1.0756×10^{-8}
Lim.4	8.3664×10^{-6}	1.24545×10^{-3}	1.1044×10^{-8}
Lim.5	4.1360×10^{-6}	1.24579×10^{-3}	1.2286×10^{-8}
Lim.6	6.3637×10^{-6}	1.245584×10^{-3}	6.7988×10^{-8}
PPM	1.3140×10^{-5}	1.24583×10^{-3}	9.7266×10^{-8}

The quality of the optimal initial conditions has often been compared by using them to forecast for a time period longer than the time window of DA. A comparison of such a forecast using $\mathbf{x}^{\text{pert}}$ and $\mathbf{x}^{\text{recovered}}$ to ($T = 2.2$ s in 1-D case and $T = 7$ h in the 2-D case) is provided in Tables IV and V, respectively. The forecast errors are reduced for all the schemes, in both 1-D and 2-D, when the $\mathbf{x}^{\text{recovered}}$ is used as the optimal initial condition. As evident, in the 1-D case, though the first-order scheme (limiter 1) yields the closest $\mathbf{x}^{\text{recovered}}$ (to $\mathbf{x}(t_0)$), the forecast obtained by using the $\mathbf{x}^{\text{recovered}}$ is inferior to that obtained by using other limiters. Limiters 3 and 5 show the least errors in recovering the initial conditions and forecasting. In the 2-D case, the PPM scheme provides the best recovery of the optimal initial conditions and least forecasting errors as well, when compared to limiters 2 and 5.

7. SUMMARY AND CONCLUSIONS

We have studied the impact of various high resolution TVD, FV (which use MUSCL slope limiters and PPM) schemes on data assimilation for two nonlinear model problems, namely

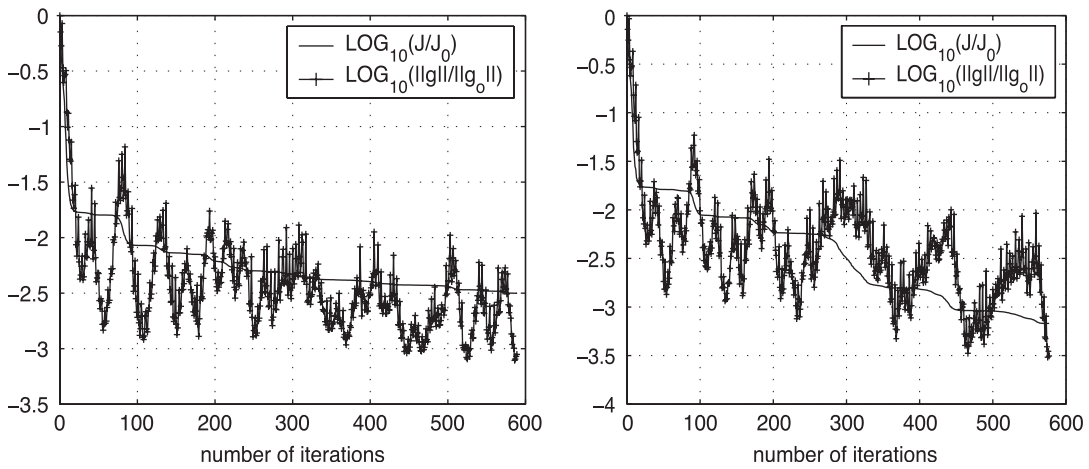

Figure 15. Variations of the normalized cost function $\mathcal{J}/\mathcal{J}_0$ and normalized gradient $\|\mathbf{g}\|/\|\mathbf{g}_0\|$ versus the number of minimization iterations using limiters 2 and 5 in forward and adjoint models for the 2-D global spherical SW equations model (in log scale).

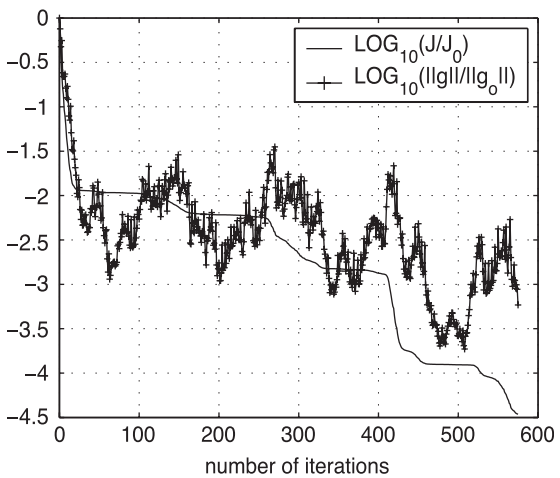

Figure 16. Same as Figure 15, but with the PPM scheme.

the viscous Burgers equation in one space dimension and the global SW equations model in two space dimensions. In both cases, smooth solutions (Section 3) have been considered. To the best of our knowledge, the PPM scheme has not been used for data assimilation in adjoint model in any previous research work, thus-far. Using the recovered initial conditions for forecasting and the closeness of the recovered optimal initial conditions to the unperturbed initial conditions as important criteria in DA, our preliminary twin experiment results indicate that limiter 5 (constrained van Leer limiter) in 1-D and the PPM in 2-D yield better results, when compared to all other schemes. Following the results obtained here, future research

Table V. Comparison of the $\mathbf{x}^{\text{recovered}}$ for different advection schemes based on data assimilation experiments, for slope limiters and PPM advection scheme in adjoint mode for 2-D global SW equations model. RMS errors $h(\text{pert}) - h(\text{unpert}) = 2.5993$, $u(\text{pert}) - u(\text{unpert}) = 0.1115$ and $v(\text{pert}) - v(\text{unpert}) = 8.0081 \times 10^{-2}$ for all the schemes; $T = 7\,\text{h}$.

RMS errors for u-wind			
Advection scheme	$u^{\text{recovered}} - u(x,0)$	$u^{\text{pert}}(x,T) - u(x,T)$	$u^{\text{recovered}}(T) - u(x,T)$
Lim.2	2.33491594196759075E-4	3.3630019845121123E-2	1.12476527351364601E-4
Lim.5	5.63382077564010195E-4	3.34702468842524234E-2	3.32149072340977508E-4
PPM	1.06535525977395802E-4	4.94900419576975101E-2	8.87413582912270193E-5
RMS errors for v-wind			
Advection scheme	$v^{\text{recovered}} - v(x,0)$	$v^{\text{pert}}(x,T) - v(x,T)$	$v^{\text{recovered}}(T) - v(x,T)$
Lim.2	2.34466472857029914E-4	3.21712340658954488E-2	1.07142814969051091E-4
Lim.5	7.40685987310806394E-4	3.24462344953588347E-2	3.08314384762099097E-4
PPM	1.70534278238826311E-4	5.51154561503572893E-2	8.69239404161232235E-5
RMS errors for height field			
Advection scheme	$h^{\text{recovered}} - h(x,0)$	$h^{\text{pert}}(x,T) - h(x,T)$	$h^{\text{recovered}}(T) - h(x,T)$
Lim.2	0.52938283634642536	3.0605303764849494	7.38576308190367625E-2
Lim.5	1.1119488545031126	3.1776765891059209	0.19626515182457319
PPM	0.13969535459251051	3.70379538782958	2.0670192049105484E-2

will focus on further investigation of validity of the above findings for a higher dimensional system with real observations. Another aspect of interest we aim to address is to compare how the error covariance matrices are propagated by these various limiters in the context of sequential variational DA, in particular, in an ensemble Kalman filtering framework. Of interest to the computational fluid dynamics community is the issue of fluid flow control, where the numerical model used for solving the governing equations are solved by FV methods, using various slope limiters and, or the PPM scheme.

APPENDIX A

In this appendix, we illustrate the differentiation of functions which require special care, such as the ABS, SIGN, DIM, MIN, MAX functions, etc. Following is an example which shows a section from the forward code to obtain

```
phi_local_min = MIN(phi_old(i-1),phi_old(i),phi_old(i+1))
```

is rewritten as

```
IF(phi_old(i-1) .LE. phi_old(i))THEN
  IF(phi_old(i-1) .LE. phi_old(i+1))THEN
     phi_local_min = phi_old(i-1)
```



```
      ELSE
         phi_local_min = phi_old(i+1)
      END IF
    ELSE IF(phi_old(i) .LE. phi_old(i+1))THEN
      phi_local_min = phi_old(i)
    ELSE
      phi_local_min = phi_old(i+1)
    END IF
```

The linearization of the above segment is give by

```
if (phi_old(i-1) .le. phi_old(i)) then
  if (phi_old(i-1) .le. phi_old(i+1)) then
    g_phi_local_min = g_phi_old(i-1)
    phi_local_min = phi_old(i-1)
  else
    g_phi_local_min = g_phi_old(i+1)
    phi_local_min = phi_old(i+1)
  endif
else if (phi_old(i) .le. phi_old(i+1)) then
  g_phi_local_min = g_phi_old(i)
  phi_local_min = phi_old(i)
else
  g_phi_local_min = g_phi_old(i+1)
  phi_local_min = phi_old(i+1)
endif
```

the corresponding adjoint statements are as follows:

```
if (phi_old(i-1) .le. phi_old(i)) then
  if (phi_old(i-1) .le. phi_old(i+1)) then
    adphi_old(i-1) = adphi_old(i-1)+adphi_local_min
    adphi_local_min = 0.d0
  else
    adphi_old(i+1) = adphi_old(i+1)+adphi_local_min
    adphi_local_min = 0.d0
  endif
else if (phi_old(i) .le. phi_old(i+1)) then
  adphi_old(i) = adphi_old(i)+adphi_local_min
  adphi_local_min = 0.d0
else
  adphi_old(i+1) = adphi_old(i+1)+adphi_local_min
  adphi_local_min = 0.d0
endif
```

It should be noted that in order to compute the adjoint variables in the backward direction, we require forward states to be available (as evident from the above piece of adjoint code)

in memory or recompute them, see research on checkpointing [76, 77] for discussion on the trade-off between storing in memory and recomputation.

ACKNOWLEDGEMENTS

The authors acknowledge the support from the NSF grant number ATM-9731472 managed by Dr Linda Peng whom we would like to thank, for her support.

We would like to thank Drs S.-J. Lin and R. Rood for providing us their finite volume shallow water equations model.

REFERENCES

1. Thomas JW. *Numerical Partial Differential Equations*: *Finite Difference Methods*. Springer: Berlin, 1998.
2. Thomas JW. *Numerical Partial Differential Equations*: *Conservation Laws and Elliptic Equations*. Springer: Berlin, 1999.
3. Le Veque RJ. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press: Cambridge, MA, 2002.
4. Zienkiewicz OC, Taylor RL. *Finite Element Method*: *Volume 1*, *The Basis* (5th edn). Butterworth-Heinemann: Stoneham, MA, 2000.
5. Zienkiewicz OC, Taylor RL. *Finite Element Method*: *Volume 2*, *Solid Mechanics* (5th edn). Wiley: New York, 2000.
6. Zienkiewicz OC, Taylor RL. *Finite Element Method*: *Volume 3*, *Fluid Mechanics* (5th edn). Butterworth-Heinemann: Stoneham, MA, 2000.
7. Canuto C, Hussaini MY, Quarteroni A, Zang TA. *Spectral Methods in Fluid Dynamics* (3rd edn). Springer: Berlin, 1991.
8. Karniadakis G, Sherwin SJ. *Spectral/hp Element Methods for Computational Fluid Dynamics* (*Numerical Mathematics and Scientific Computation*) (2nd edn). Oxford University Press: Oxford, 2005.
9. Le Veque RJ. *Numerical Methods for Conservation Laws*. Birkhäuser: Basel, 1990.
10. Laney CB. *Computational Gasdynamics*. Cambridge University Press: Cambridge, MA, 1998.
11. Durran DR. *Numerical Methods for Wave Equations in Geophysical Fluid Dynamics*. Springer: New York, 1999.
12. Zhang S, Zou X, Ahlquist J, Navon IM. Use of differentiable and nondifferentiable optimization algorithms for variational data assimilation with discontinuous cost functions. *Monthly Weather Review* 2000; **128**(12): 4031–4044.
13. Rood RB. Numerical advection algorithms and their role in atmospheric transport and chemistry models. *Reviews of Geophysics* 1987; **25**(1):71–100.
14. Lin S-J, Chao WC, Sud YC, Walker GK. A class of the van Leer transport schemes and its applications to the moisture transport in a general circulation model. *Monthly Weather Review* 1994; **122**:1575–1593.
15. Lin S-J, Rood RB. Multidimensional flux-form semi-Lagrangian transport schemes. *Monthly Weather Review* 1996; **124**:2046–2070.
16. van Leer B. Towards the ultimate conservative difference scheme. IV: a new approach to numerical convection. *Journal of Computational Physics* 1977; **23**:276–299.
17. Lin S-J. A vertically Lagrangian finite-volume dynamical core for global models. *Monthly Weather Review* 2004; **132**:2293–2307.
18. Lin S-J, Rood RB. An explicit flux-form semi-Lagrangian shallow-water model on the sphere. *Quarterly Journal of the Royal Meteorological Society* 1997; **123**:2477–2498.
19. Smolarkiewicz PK. A fully multidimensional positive definite advection algorithm with small implicit diffusion. *Journal of Computational Physics* 1984; **54**:325–362.
20. Smolarkiewicz PK, Margolin LG. MPDATA: a finite-difference solver for geophysical flows. *Journal of Computational Physics* 1998; **140**:459–480.
21. Leonard BP. A stable and accurate convective modeling procedure based on quadratic upstream interpolation. *Computer Methods in Applied Mechanics and Engineering* 1979; **19**:59–98.
22. Leonard BP. The ULTIMATE conservative difference scheme applied to unsteady one-dimensional advection. *Computer Methods in Applied Mechanics and Engineering* 1991; **88**(1):17–74.
23. Leonard BP. Order of accuracy of QUICK and related convection–diffusion schemes. *Applied Mathematical Modelling* 1995; **19**(11):640–653.
24. Vukićević T, Steyskal M, Hecht M. Properties of advection algorithms in the context of variational data assimilation. *Monthly Weather Review* 2001; **129**(5):1221–1231.

25. HYCOM consortium for Data-Assimilative Ocean Modeling. http://hycom.rsmas.miami.edu (4 September 2005).
26. Sokol Z. Comparison of several numerical schemes applied to advection equations. *Quarterly Journal of the Royal Meteorological Society* 1999; **125**:213–224.
27. Iskandarani M, Levin JC, Choi B-J, Haidvogel DB. Comparison of advection schemes for high-order h-p finite element and finite volume methods. *Ocean Modelling* 2005; **10**:233–252.
28. Gunzburger MD. *Perspectives in Flow Control and Optimization*. SIAM: Philadelphia, PA, 2003.
29. Cacuci DG. *Sensitivity and Uncertainty Analysis*. Chapman & Hall/CRC: London, 2003.
30. Kalnay E. *Atmospheric Modeling*, *Data Assimilation and Predictability*. Cambridge University Press: Cambridge, MA, 2002.
31. Mohammadi B, Pironneau O. *Applied Shape Optimization for Fluids*. Oxford University Press: Oxford, 2001.
32. Navon IM, Zou X, Derber J, Sela J. Variational data assimilation with an adiabatic version of the NMC spectral model. *Monthly Weather Review* 1992; **120**:1433–1446.
33. Ledimet FX, Talagrand O. Variational algorithms for analysis and assimilation of meteorological observations–theoretical aspects. *Tellus Series A—Dynamic Meteorology and Oceanography* 1986; **38**:97–110.
34. Thuburn J, Haine TWN. Adjoint of nonoscillatory advection schemes. *Journal of Computational Physics* 2001; **171**:616–631.
35. Burgers JM. *A Mathematical Model Illustrating the Theory of Turbulence*. Advances in Applied Mechanics, vol. 1. Academic Press: New York, 1948; 171–199.
36. Whitham G. *Linear and Nonlinear Waves*. Wiley-Interscience: New York, 1974.
37. Zhang DS, Wei GW, Kouri DJ, Hoffman DK. Burgers' equations with high Reynolds number. *Physics of Fluids* 1997; **9**(6):1853–1855.
38. Fletcher CAJ. *Computational Techniques for Fluid Dynamics*. Springer: Berlin, 1988.
39. van Leer B. *Towards the Ultimate Conservative Difference Scheme. I*: *The Quest for Monotonicity*. Springer Lecture Notes in Physics, vol. 18. Springer: Berlin, 1973; 163–168.
40. van Leer B. Towards the ultimate conservative difference scheme. II: monotonicity and conservation combined in a second-order scheme. *Journal of Computational Physics* 1974; **14**:361–370.
41. van Leer B. Towards the ultimate conservative difference scheme. III: upstream-centered finite difference schemes for ideal compressible flow. *Journal of Computational Physics* 1977; **23**:263–275.
42. Osher S. Convergence of generalized MUSCL schemes. *SIAM Journal on Numerical Analysis* 1985; **22**: 947–961.
43. Harten A. High resolution schemes for hyperbolic conservations laws. *Journal of Computational Physics* 1983; **49**:357–393.
44. Harten A. On a class of high resolution total variation stable finite difference schemes. *SIAM Journal on Numerical Analysis* 1984; **21**:1–23.
45. van Leer B. Towards the ultimate conservative difference scheme. V: a second-order sequel to Godunov's method. *Journal of Computational Physics* 1979; **32**:101–136.
46. Harten A, Osher S. Uniformly high-order accurate nonoscillatory schemes. I. *SIAM Journal on Numerical Analysis* 1987; **24**:279–309.
47. Harten A, Engquist B, Osher S, Chakravarthy SR. Uniformly high-order accurate essentially non-oscillatory schemes, III. *Journal of Computational Physics* 1987; **71**:231–303.
48. Harten A. ENO schemes with subcell resolution. *Journal of Computational Physics* 1987; **83**:148–184.
49. Harten A, Lax PD, van Leer B. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM Review* 1983; **25**(1):35–61.
50. Colella P, Woodward P. The piecewise-parabolic method (PPM) for gas-dynamical simulations. *Journal of Computational Physics* 1984; **54**:174–201.
51. Woodward PR, Colella P. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of Computational Physics* 1984; **54**:115–173.
52. Carpenter RL, Droegemeier KK, Woodward PW, Hanem CE. Application of the piecewise parabolic method (PPM) to meteorological modeling. *Monthly Weather Review* 1990; **118**:586–612.
53. Shu C-W. Total-variation-diminishing time discretizations. *SIAM Journal of Scientific Statistical Computation* 1988; **9**:1073–1084.
54. Shu C-W, Osher S. Efficient implementation of essentially non-oscillatory shock capturing schemes. *Journal of Computational Physics* 1988; **77**:439–471.
55. Gottlieb S, Shu C-W. Total variation diminishing Runge–Kutta schemes. *Mathematics of Computation* 1998; **67**(221):73–85.
56. Lin S-J. A finite-volume integration method for computing pressure gradient force in general vertical coordinates. *Quarterly Journal of the Royal Meteorological Society* 1997; **123**:1749–1762.
57. Williamson DL, Drake JB, Hack JJ, Jakob R, Swarztrauber PN. *Journal of Computational Physics* 1992; **102**:211–224.
58. Haltiner GJ, Williams RT. *Numerical Prediction and Dynamic Meteorology* (2nd edn). Wiley: New York, 1980.

59. Suarez MJ, Takacs LL. Documentation of the ARIES/GEOS dynamical core: version 2. *Technical Report*, *NASA Technical Memorandum 104606*, vol. 5, 1996.
60. Benton ER, Platzman GW. A table of solutions of the one-dimensional Burgers equation. *Quarterly of Applied Mathematics* 1972; **30**:195–212.
61. Numerical schemes applied to the Burgers and Buckley–Leverett equations. http://www.extra.rdg.ac.uk/Maths/Research/Publications/Msc_dissertations/rakhib_ahmed.pdf [30 March 2005].
62. The community atmosphere model (CAM). http://www.ccsm.ucar.edu/models/atm-cam (11 September 2005).
63. Phillips NA. Numerical integration of the primitive equations on the hemisphere. *Monthly Weather Review* 1959; **87**:333–345.
64. Haurwitz B. The motion of atmospheric disturbances on the spherical earth. *Journal of Marine Research* 1940; **3**:254–267.
65. Homescu C, Navon IM, Li Z. Suppression of vortex shedding for flow around a circular cylinder using optimal control. *International Journal for Numerical Methods in Fluids* 2002; **38**:43–69.
66. Yang W, Navon IM. Documentation of the tangent linear model and its adjoint of the adiabatic version of the NASA GEOS-1 C-grid GCM (Version 5.2). *NASA Technical Memorandum 104606*. Edited by Max Suarez. *NASA Technical Memorandum Report Series on Global Modeling and Data Assimilation*, 1996.
67. Tangent and Adjoint Model Compiler (TAMC). http://www.autodiff.com/tamc [3 April 2005].
68. Giering R, Kaminski T. Recipes for adjoint code construction. *ACM Transactions on Mathematical Software* 1998; **24**:437–474.
69. Griewank A. *Evaluating Derivatives*: *Principles and Techniques of Algorithmic Differentiation*. SIAM: Philadelphia, PA, 2000.
70. Documentation of the TLM and adjoint models of the Lin–Rood spherical shallow water finite volume model. http://www.csit.fsu.edu/navon/publ.html [20 September 2005].
71. Liu DC, Nocedal J. On the limited memory BFGS method for large scale minimization. *Mathematical Programming* 1989; **45**:503–528.
72. Nash SG, Nocedal J. A numerical study of the limited memory BFGS method and the truncated-Newton method for large scale optimization. *SIAM Journal on Optimization* 1991; **1**:358–372.
73. Nocedal J, Wright SJ. *Numerical Optimization*. Springer: New York, 1999.
74. Daescu D, Navon IM. An analysis of a hybrid optimization method for variational data assimilation. *International Journal of Computational Fluid Dynamics* 2003; **17**(4):231–233.
75. Zou X, Navon IM, Berger M, Phua MK, Schlick T, LeDimet FX. Numerical experience with limited-memory, quasi-Newton methods for large-scale unconstrained nonlinear minimization. *SIAM Journal on Optimization* 1993; **3**(3):582–608.
76. Restrepo JM, Leaf GK, Griewank A. Circumventing storage limitations in variational data assimilation studies. *SIAM Journal on Scientific Computing* 1998; **19**(5):1586–1605.
77. Griewank A, Walther A. An implementation of checkpointing for the reverse or adjoint mode of computational differentiation. *ACM Transactions on Mathematical Software* 2000; **26**(1):19–45.

