THE FLORIDA STATE UNIVERSITY

COLLEGE OF ARTS AND SCIENCES

DETERMINISTIC AND STOCHASTIC ASPECTS OF DATA ASSIMILATION

By

SANTHARAM AKELLA

A Dissertation submitted to the
Department of Mathematics
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Degree Awarded:
Spring Semester, 2006

The members of the Committee approve the Dissertation of Santharam Akella defended on April 03, 2006.

Ionel Michael Navon
Professor Directing Dissertation

James J. O'Brien
Outside Committee Member

Gordon Erlebacher
Committee Member

Qi Wang
Committee Member

Mark Sussman
Committee Member

The Office of Graduate Studies has verified and approved the above named committee members.

ACKNOWLEDGEMENTS

First of all, I would first like to acknowledge my gratitude to Prof. I. M. Navon, my Ph.D. advisor, to have mentored me, patiently, for all these years. He introduced and provided me a valuable training in the exciting field of optimal control of complex systems, for which I thank him from the bottom of my heart. It was a pleasure to have the opportunity to work and learn from him.

I would like to thank Dr. Ricardo Todling for providing valuable comments, remarks and insights into current topics in data assimilation.

Also, I would like to thank Prof. Gordon Erlebacher for always cheering me in the aisles of the School of Computational Science (SCS), early in the mornings, and late in the evenings.

I would also like to thank other doctoral committee members, Professors Qi Wang, Mark Sussman, James O'Brien for having provided me an opportunity to be able learn from their experiences through various courses I signed up with them, and also for having agreed to be on my doctoral committee.

Various system administration staff of the SCS helped kept running the SGIs during all times, I really appreciate all of their help during all of these years. Also I would like to acknowledge the NSF, which supported me during the course of this dissertation work, via grant number ATM-9731472. Drs. S.-J. Lin and R. Rood provided their finite volume shallow water model.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

The principles of optimal control of distributed parameter systems are used to derive a powerful class of numerical methods for solutions of inverse problems, called data assimilation (DA) methods. Using these DA methods one can efficiently estimate the state of a system and its evolution. This information is very crucial for achieving more accurate long term forecasts of complex systems, for instance, the atmosphere. DA methods achieve their goal of optimal estimation via combination of all available information in the form of measurements of the state of the system and a dynamical model which describes the evolution of the system. In this dissertation work, we study the impact of new nonlinear numerical models on DA.

High resolution advection schemes have been developed and studied to model propagation of flows involving sharp fronts and shocks. The impact of high resolution advection schemes in the framework of inverse problem solution/ DA 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 DA is the subject of this work. In 1-D we use a nonlinear viscous Burgers equation and in 2-D a global nonlinear shallow water model has been used.

The results obtained show that using the various advection schemes consistently improves variational data assimilation (VDA) in the strong constraint form, which does not include model error. However, the cost functional included efficient and physically meaningful construction of the background cost functional term, $\mathcal{J}_b$ using balance and diffusion equation based correlation operators. This was then followed by an in-depth study of various approaches to model the systematic component of model error in the framework of a weak constraint VDA. Three simple forms, decreasing, invariant, and exponentially increasing in

time forms of evolution of model error were tested. The inclusion of model error provides a substantial reduction in forecasting errors, in particular the exponentially increasing form in conjunction with the piecewise parabolic high resolution advection scheme was found to provide the best results.

Results obtained in this work can be used to formulate sophisticated forms of model errors, and could lead to implementation of new `VDA` methods using numerical weather prediction models which involve high resolution advection schemes such as the van Leer slope limiters and the PPM.

CHAPTER 1

INTRODUCTION

The potential benefits of more accurate long term forecasts (of atmosphere and oceans) are obvious, particularly in the event of severe weather scenarios, such as hurricanes, tornadoes, etc. Hence the need to better understand and improve various steps involved in weather forecasting.

The evolution of the atmosphere is governed by physical laws of motion and conservation of energy. These laws when expressed by a series of differential equations (usually partial differential equations, or PDEs) give rise to the so-called numerical weather prediction (NWP) models. Present weather forecasting system primarily utilizes NWP models and meteorological data collected (temperature, wind velocity, pressure, humidity, etc.) from various sources (land stations, balloons, buoys, ships, aircrafts, satellites, radiosondes, rawinsondes, etc).

The NWP models solve an initial value problem, i.e., if we know the initial condition of the atmosphere, using the NWP model, we computationally solve the equations for a later time and obtain new values of the variables used to represent the state of the atmosphere. Therefore a *good* quality forecast requires that these NWP models represent accurately the dynamics of the atmosphere (including the oceans) and the initial conditions supplied for integration be known accurately. Efficient estimation of the current state of the atmosphere and its evolution using the information provided by NWP model prediction and observations is carried out by data assimilation (DA) [1]. Due to our lack of understanding of various complicated processes and their interaction, the formulation of the NWP models involves parameterization of various chemical, biological and physiological processes. DA also provides us the framework to optimally estimate those parameters used in modeling of such complex processes.

A DA system consists of three important components: a set of observations or measurements, a dynamical forecast model (or, the NWP model) and a methodology to *meld* the

observations with the model, i.e., the data assimilation scheme itself. It is essential to remember that both the observations and the models have errors. The observational errors consist of instrumental noise, sampling errors, environmental noise, and any possible errors involved in the interpretation of sensor measurements. Regarding models, since no model can ever replicate the actual evolution of nature, all the dynamical models are imperfect; with errors due to approximation of the physics (and, or biology or chemistry) at various length and time scales, due to the discretization of continuous dynamical equations to arrive at the numerical model, etc. Therefore the DA scheme must take into account these errors, and it should not introduce any new errors. Ideally the DA output must be more reliable in terms of quality of the information, when compared to the quality of the information provided by the observations and models.

DA methods considered thus-far are based on either estimation theory (sequential methods such as Kalman, extended Kalman, ensemble Kalman filtering methods) or variational methods (3D-Var, PSAS, 4D-Var) which are based on minimization of a cost functional which measures the distance (in a suitable norm) between observations and NWP model forecasts. Estimation theory comprises of methods to estimate the state of a system by combination (using a statistical approach) of all the available reliable information of the system from measurements and dynamical models. The so-called Kalman filter (KF), derived by Kalman and Bucy in 1960-1961 [2, 3] provided the path-breaking estimation or filtering algorithm that is now used extensively in a wide range of signal processing areas, such as, navigational and control systems. There are two well known estimation procedures: the *filtering* and the *smoothing* processes. If the past and presently available information is used to estimate the current state or predict the state of the system, the procedure is called filtering process. Whereas if we were to estimate the state of the system in past time, using all the currently available information, then the methodology is called smoothing (or interpolation) process. For linear dynamical systems, the KF has been proved to provide a sequential, unbiased, minimum error variance estimate of the state of the system using a linear combination of all available past measurements and dynamics. One of the first applications of the KF theory to arrive at a sequential DA scheme was provided by Ghil *et al.* (1981) [4]. They used a linearized 1-D shallow water model as a surrogate to the complex NWP models, extension to 2-D was accomplished by Todling and Ghil (1992) [5]. For an in-depth development of various statistical, sequential estimation methods, please see [6], the extensions of the KF

theory to nonlinear dynamical systems using the extended KF (EKF) and ensemble KF (EnKF) are reviewed in [7, 1] and references therein.

Now we turn our attention to the other class of DA methods, which are variational in nature, called variational data assimilation (VDA) methods; which in the context of evolution of the atmosphere is the subject of this dissertation (for oceanic DA, please see Wunsch, 1996[8], Yu and O'Brien, 1995[9] and references there-in). The objective of VDA 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 Navon *et al.* (1992)[10], Ledimet and Talagrand (1986)[11] and Lorenc (1986)[12] for further details. In general, fusion of models with measurements (observations) and finding response of a system to (external) disturbances requires solution of inverse problems, by deriving an optimality criteria which involves formulation of a cost functional, derivation of a system of linearized dynamical equations, and an adjoint system of equations. VDA is just a particular case from a variety of methods to solve inverse problems, which is based on optimal control theory. Other examples of inverse problems include those in aerodynamics, such as minimization of drag, maximization of lift (as target functionals) which are often performed by considering the geometry of the immersed body in the fluid as the control variable, called shape optimization [13], for other applications and details, see [14]. Solutions of inverse problems typically involve minimization of a goal functional, using large scale minimization algorithms. These methods often require availability of gradient of the cost functional with respect to the control variables, which is efficiently by adjoint methods (which are integrated backwards in time), see seminar work by Cacuci [15, 16, 17, 18]. Chapter 2 provides a brief introduction to various VDA methods and their mathematical formulation, the derivation of tangent linear and adjoint models is presented and finally a brief summary of the optimization algorithms used in VDA is provided.

During the course of the past few decades a significant improvement in NWP models has been achieved. There have been many reasons for this progress, the most important contributions have are due to increased understanding of the physical and natural process involved, advances in numerical approximation methods, faster, more accurate and larger storage computational resources [1]. Also a significantly larger number of observations are increasingly becoming available from various sources, and from locations previously not

covered. However the usage of these more sophisticated numerical models and larger and better quality observations in DA system is not readily accomplished, in the sense that there is always a lag in time for the latest NWP model and observations to be used for DA. This could be due to the fact that a significant amount of resources in terms of human resources and time are required to *update* the DA scheme (for e.g., derivation of new adjoint model that is compatible with the new NWP model); also simultaneously we need to improve the DA methodology so that it is able to provide better quality estimates than that possible previously, hence there are two possible paths to improvement in the field of DA. One of them is to consider various challenges involved in the implementation of newer numerical models in a DA system and we embarked on such a mission in chapter 3. There we concentrate on the issues of impact of different high order advection schemes for the discretization of the advection term(s) in the framework of inverse problems and problems related to VDA. We addressed these issues using two specific cases, a viscous Burgers equation model in one space dimension and a global shallow water equations model in two space dimensions. The second topic of improving DA schemes is investigated in the chapter 4.

NWP models are imperfect, since they are discretized, dissipative and dispersion errors arise, and, moreover subgrid processes are not included. In addition, most of the physical processes and their interactions in the atmosphere are parametrized, also a complete mathematical modeling of the boundary conditions and forcing terms can never be achieved. Usually all of these modeling drawbacks are collectively addressed by the term, *model error* (ME). Motivated by the results achieved in chapter 3, we attempted to improve the quality of the solutions obtained via VDA, by incorporation of modeling errors using various high resolution advection schemes presented in chapter 3. When the ME is included in VDA, it is usually called *weak constraint* VDA and the formulation of such a methodology requires a reformulation of the numerical model, the cost functional and the adjoint model; details of such are available in chapter 4.

Finally in chapter 5 we summarize the results presented, their scope and significance in the light of development of DA techniques is discussed. Also a brief summary of the current and future work is provided.

CHAPTER 2

VARIATIONAL DATA ASSIMILATION METHODS

Data assimilation (DA) denotes a class of numerical methods which are derived from principles of optimal control of distributed parameter systems. These methods provide us tangible means for estimating the state of a system via combination of information obtained from observational data and a numerical model. Such an estimate of the state can be used for forecasting the system for longer time interval with more accuracy and reliability. The observational data is obtained from heterogeneous measurements of the state of the system under investigation. The other principal ingredient for DA is the numerical model, which is used to forecast the state of the system at a future time, if the present state is supplied as input. Observational data is independent of the information obtained from usage of numerical models. Using mathematical control theory, variational methods are used to reformulate the DA problem as an optimization problem. In this chapter, we provide details on the various ingredients required for DA, a brief summary of the different DA methods (variational methods are described in more detail, than the non-variational algorithms), the *tangent linear* and *adjoint models* are introduced, and finally a class of nonlinear optimization methods are discussed.

2.1 Mathematical description of the various ingredients for DA

2.1.1 State vector

All the independent variables that are used to identify the state of the system at any point in time are collectively called as the *state vector*, denoted by $\mathbf{x}$. If we are studying the evolution of the entire atmosphere, the state vector would comprise of at-least five variables: surface pressure, p_s, temperature, T, moisture, q, and the components of wind along the longitudinal

direction, u, and along the latitudes, v. Usually atmospheric studies are conducted using a discrete representation of the spherical earth. If the discretization consists of n_λ grid points along the longitude (λ), n_θ grid points along the latitude and say, 50 vertical levels (on the sphere), then the total number of variables is equal to $N = 5 \times n_\lambda \times n_\theta \times 50$. Usually all the variables are ordered by grid point and by the variable, forming a single vector of length N. Thus the state vector, $\mathbf{x}$ is of dimension $N \times 1$. To provide an illustration of the huge size of the system, if we were to consider a $1^o \times 1^o$ discretization of the sphere, i.e, $n_\lambda = 360$ grid points along the longitude and $n_\theta = 180$ grid points along the latitude, then $N = 16.2 \times 10^6$.

2.1.2 The model

Discrete, computer-based numerical weather prediction (`NWP`) models are used for evolving the state of an atmospheric, oceanic, or coupled system from time t_k to time t_{k+1}, symbolically,

$$\mathbf{x}(t_{k+1}) = \mathcal{M}[\mathbf{x}(t_k)] + \boldsymbol{\eta}(t_k). \tag{2.1}$$

The model' s dynamics are represented in the above equation using the (discrete) operator, $\mathcal{M}$ and $\boldsymbol{\eta}(t_k)$ accounts for the model error at time t_k. Due to our inability to analytically solve the coupled system of `PDEs` that are used to model the evolution of the system, `NWP` models are usually obtained after discretization using numerical methods such as finite difference, finite volume, finite elements or spectral methods of the full partial differential equations (`PDEs`) that are assumed to govern the flow of the atmosphere (see [19] for a description of various discretization methods used in the operational `NWP` models). These computer simulations or prediction models take into account all the available statistical, dynamical and physical knowledge of the atmosphere (as resolved at the level of discretization). Thus these nonlinear models provide only an approximation of the true evolution of the atmosphere, since the *true* evolution of the system may differ from (2.1) by unknown random and, or systematic errors. By comparing model forecasts and the true evolution of the atmosphere over a period of time, different approaches have been used to model the model error $\boldsymbol{\eta}(t_k)$. The mean of the model error is denoted by $E\{\boldsymbol{\eta}_k\}$, where $E\{\cdot\}$ denotes the mathematical expectation operator and the covariance of the model error at t_k is denoted by $\mathbf{Q}_k = E\{\boldsymbol{\eta}_k \boldsymbol{\eta}_k^T\}$. If the numerical grid comprises of N grid points then the discrete state vector, $\mathbf{x}$ and model error vector, $\boldsymbol{\eta}$ are

of length N. The expectation or mean of the model error at any time step, t_k is given by $E\{\boldsymbol{\eta}\} = [E\{\eta_1\}, E\{\eta_2\}, \ldots, E\{\eta_N\}]^T$. The model error covariance is given by the following $N \times N$ matrix,

$$\mathbf{Q} = \begin{pmatrix} E\{\eta_1\eta_1\} & E\{\eta_1\eta_2\} & \ldots & \ldots & \ldots & E\{\eta_1\eta_N\} \\ E\{\eta_2\eta_1\} & \ldots & \ldots & \ldots & \ldots & E\{\eta_2\eta_N\} \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ E\{\eta_N\eta_1\} & \ldots & \ldots & \ldots & \ldots & E\{\eta_N\eta_N\} \end{pmatrix}.$$

Further details and investigation of the various forms of model error are provided in Chapter 4. As evident from the above equation (2.1), NWP models solve an initial value problem, given an estimate of the present state of the atmosphere, say $\mathbf{x}(t_0)$, then the model can be used to forecast the state $\mathbf{x}(t_n)$, for any future time t_n.

2.1.3 Observations

Measurements provide another valuable source of information regarding the state of the system, called *observations* and are collectively denoted by $\mathbf{y}^o$. Due to the complex and dynamic nature of the atmosphere (and oceans), observations can be broadly classified into two types: direct or "conventional" and indirect or "unconventional". Direct observations consist of wind, temperature, and humidity measurements. Weather stations that based on land at different locations, balloons, buoys, ships, aircrafts, radiosondes, rawinsondes, etc provide us direct observations. Indirect observations consist of measurements obtained from various satellites, which measure radiances, various types of imagery (visible, infrared, and water vapor images), which do not explicitly enter in the state vector representation, but are functionals of the state vector. Therefore indirect observations can be used only if we know how to evaluate the functional form. Due to the heterogeneous and non-uniformly distributed (in both space and time) nature of observations, the so-called nonlinear *observation operator*, H is introduced; it is a mapping in both space and time. Observational data is rarely available on a regular grid, usually the locations at which the observations are available differ from the grid point locations on which the estimated field is sought. Therefore all the observations, irrespective of the location where and when they were measured must be present at the NWP model time steps, and grid points, this task is accomplished by the observational operator. In the case of direct observations, H is a linear mapping of observations collected

at various irregularly spaced locations to a regularly spaced numerical model grid. Whereas for indirect observations, H is a complex operator, and usually leads to inverse problems. For example, most of the satellite data is a non-linear integral measurement of the temperature or humidity present in the atmosphere, therefore to obtain the temperature profile, one has to solve an inverse radiative transfer problem where the input is the radiance measured by the satellite.

There are errors associated with the observations, a primary sources of such errors is instrumentation error, which usually has a constant mean and quantifiable variance. Another source of error is through the numerical operations involved in the observation operator. If the observations $\mathbf{y}^o$ at time t_k are defined by

$$\mathbf{y}_k^o = H_k[\mathbf{x}^t(t_k)] + \boldsymbol{\epsilon}_o(t_k), \tag{2.2}$$

where $\mathbf{x}^t$ is the true state of the atmosphere and ϵ_o represents errors in the observations. In the above equation, we assumed that the observational errors are additive in nature. The observational errors are assumed to be uncorrelated with the model errors, i.e,

$$E\{\boldsymbol{\eta}_k \boldsymbol{\epsilon}_{ok}^T\} = E\{\boldsymbol{\epsilon}_{ok} \boldsymbol{\eta}_k^T\} = \mathbf{0},$$

and the observational error covariance is denoted by $\mathbf{R}_k = E\{\boldsymbol{\epsilon}_{ok} \boldsymbol{\epsilon}_{ok}^T\}$.

In general, observations are sparse, at least when compared to the density of the discretized grid of the NWP model on which the state vector is defined. If the dimension of the state vector is $N \times 1$, and the dimension of the observations vector, $\mathbf{y}^o$ is $p \times 1$, then typically, $p \ll N$, p is smaller than N by about one or two orders of magnitude. This lack of information is addressed by introducing the background information, explained in the following section.

2.1.4 Background field

The background field or *first guess* is an a-priori estimate of the state of the atmosphere (denoted by $\mathbf{x}^b$) at all the numerical model grid points is used to supplement for the lack of as many observations as the number of grid points. Ideally speaking, the background field should be our best estimate of the state of the atmosphere before observations are available. Usually a short-range forecast (typically 6 hours) is used to generate $\mathbf{x}^b$. The errors in the background field are quantified by $\boldsymbol{\epsilon}_b = \mathbf{x}^b - \mathbf{x}^t$, and the background error covariance matrix,

B is given by $E\{\boldsymbol{\epsilon}_{bk}\epsilon_{b_k}^T\}$. Further details regarding the role played by the background field in VDA and its formulation are provided in Chapter 4.

2.2 Brief summary of various DA methods

One of the first significant DA algorithm that was implemented was a simple and economical method, which was based on an empirical approach called the **successive corrections method (SCM)**, introduced by Bergthorsson and Doos (1955) [20] and Cressman (1959) [21]. It belongs to a general class of methods called objective analysis (see [22] for more details). Objective analysis is a process of interpolating data from irregularly spaced locations (such as the collected observations) to a fixed grid. SCM is an iterative procedure to estimate the value of a variable (which belongs to a field) on a regular grid, using the background value of the field as the initial value at the zero-th iteration. In the following iterations, value of the variable is computed by successively correcting, using the observational field. If z_i^k denotes the value of a variable at the i^{th} grid point and k^{th} iteration, then the iterative procedure of SCM can be summarized as,

$$z_i^0 = z_i^b, \tag{2.3}$$

$$z_i^{k+1} = z_i^k + \frac{\sum_{j=1}^{N} w_{ij}(z_j^O - z_j^k)}{\sum_{j=1}^{N} w_{ij} + \alpha^2}, \tag{2.4}$$

where z_i^b is the background value of the field, $\mathbf{z}$ at the i^{th} grid point, z_j^O is the j^{th} observation surrounding the i^{th} grid point, z_j^k is the k^{th} iterated value of the field estimate evaluated at the j^{th} observation point (calculated by interpolation from surrounding grid points), α^2 is an estimate of the ratio of the observational to background error variance. The weights w_{ij} are used to quantify the influence of the observed value on the estimated field, as function of the distance (in a suitable norm) between the location of the observation and the i^{th} grid point. Following forms of weights have been used in literature,

$$w_{ij} = \begin{cases} \frac{R^2 - r_{ij}^2}{R^2 + r_{ij}^2}, & \text{if } r_{ij}^2 \leqslant R^2 \\ 0, & \text{otherwise,} \end{cases}$$

$$w_{ij} = \begin{cases} e^{-r_{ij}^2/2R^2}, & \text{if } r_{ij}^2 \leqslant R^2 \\ 0, & \text{otherwise,} \end{cases}$$

where r_{ij} is the euclidean distance between a j^{th} observation point and i^{th} grid point, R is the radius of influence (of any observation), which is allowed to vary from one iteration to another, and its value could range from 300 Km to 1500 Km based on the location of field being estimated (larger values for higher altitudes and lower range of values for fields near the surface of the earth).

Nudging is another empirical method used for DA of small-scale observations, and is not generally used for large-scale assimilation. It is based on a simple idea of *nudging*, or dynamically relaxing the solution of the numerical model towards the observations (interpolated to the model grid) by adding a suitable forcing term to the governing differential equations. For e.g., if the following differential equation describes the evolution a variable, p in time,

$$\frac{\partial p}{\partial t} = g(p, q, t),$$

where $g(.)$ describes the dynamics of evolution of p. Using the method of nudging model predicted values of p to the observed values p^{obs}, we modify the above equation by adding a forcing term to the right hand side,

$$\frac{\partial p}{\partial t} = g(p, q, t) + \frac{p^{obs} - p}{\tau_p}, \tag{2.5}$$

where τ_p is *tuned* based on empirical (and, or heuristic) considerations and may depend on the particular physical variable under consideration. If τ_p (which is also called relaxation time scale) is chosen to be very small then the solution converges to the observations very quickly, and the model dynamics do not have enough time to adjust. Whereas if τ_p is too large, errors in the model could grow before nudging becomes effective; optimal methods to estimate τ_p using parameter estimation have been studied by Zou et al. (1992) [23] and Vidard et al. (2003) [24].

Both SCM and nudging are DA methods that use the availability of observations, background field and numerical model to estimate the state of the atmosphere. Though all the available information is being used, both of these methods do not guarantee that the estimate is "optimal", also, they do not take into account the errors associated with these fields (such as the observation, background and model error covariances). Optimal interpolation (OI) was one of the first operationally implemented method which was formulated for optimally estimating the state of the atmosphere, other variational methods

such as 3D-Var and physical space analysis scheme (PSAS) also provide other approaches towards the same goal. Note all these three methods account for only the spatial distribution of observations, and neglect their temporal distribution (for e.g., 3D-Var denotes variational data assimilation of three dimensional fields). Using a weighted linear combination of the background information and observations, we can write the following expression for an estimate of the state,

$$\mathbf{x} = \mathbf{W}_b \, \mathbf{x}^b + \mathbf{W}_o \, \mathbf{y}^o, \tag{2.6}$$

where $\mathbf{W}_b$ and $\mathbf{W}_o$ are the weighting matrices assigned to the background and observational vector fields respectively. Another way to write the above linear combination is,

$$\mathbf{x} = \mathbf{x}^b + \mathbf{W} \, (\mathbf{y}^o - H[\mathbf{x}^b]), \tag{2.7}$$

where the observational operator H has been used to interpolate the background field on the model grid to the locations of the observations.

2.2.1 Optimal interpolation

For the estimate of the state given by the above equation (2.7) to be optimal in some sense, the weighting matrix, $\mathbf{W}$ must satisfy certain property. In other words, we seek a $\mathbf{W}$ such that the estimate, $\mathbf{x}^*$, among all possible $\mathbf{x}$ is optimal according to some criteria. An obvious optimality criteria is that $\mathbf{x}^*$ should minimize the estimation error with respect to the true state of the atmosphere. We will define the estimation error with respect to the true state $\mathbf{x}^t$ as,

$$\epsilon_a = \mathbf{x} - \mathbf{x}^t. \tag{2.8}$$

Then minimization of the error covariance, $\mathbf{P} = E\{\epsilon_a \epsilon_a^T\}$ provides us the optimal $\mathbf{x}^*$. Let us consider the following residue,

$$\mathbf{y}^o - H[\mathbf{x}^b],$$

we can rewrite the second expression as, $H[\mathbf{x}^b] = H[\mathbf{x}^b - \mathbf{x}^t + \mathbf{x}^t] = H[\mathbf{x}^t] + \mathbf{H}[\mathbf{x}^b - \mathbf{x}^t]$, where $\mathbf{H}$ is obtained by linearization of H, $\mathbf{H} = \frac{\partial H}{\partial x}$.

Recall that $\epsilon_b = \mathbf{x}^b - \mathbf{x}^t$, and $\epsilon_o = \mathbf{y}^o - H[\mathbf{x}^t]$, therefore

$$\mathbf{y}^o - H[\mathbf{x}^b] = \mathbf{y}^o - H[\mathbf{x}^t] - \mathbf{H}[\mathbf{x}^b - \mathbf{x}^t] = \epsilon_o - \mathbf{H}\epsilon_b.$$

Using (2.7), (2.8) and the above relationship, the estimation error can be rewritten as,

$$\epsilon_a = \mathbf{x}^b + \mathbf{W}\left(\mathbf{y}^o - H[\mathbf{x}^b]\right) - \mathbf{x}^t = (\mathbf{x}^b - \mathbf{x}^t) + \mathbf{W}(\epsilon_o - \mathbf{H}\epsilon_b) = \epsilon_b + \mathbf{W}(\epsilon_o - \mathbf{H}\epsilon_b).$$

Therefore the error covariance, $\mathbf{P} = E\{\epsilon_a \epsilon_a^T\} = E\{[\epsilon_b + \mathbf{W}(\epsilon_o - \mathbf{H}\epsilon_b)][\epsilon_b + \mathbf{W}(\epsilon_o - \mathbf{H}\epsilon_b)]^T\}$ Using the definitions of the background and observational error covariances, and the assumption that the observational error is not correlated with the background errors, we obtain the following expression for $\mathbf{P}$,

$$\mathbf{P} = \mathbf{B} + \mathbf{W}[\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^T]\mathbf{W}^T - \mathbf{W}\mathbf{H}\mathbf{B} - \mathbf{B}\mathbf{H}^T\mathbf{W}^T. \tag{2.9}$$

If there exists a $\mathbf{W}$ that yields the least $\mathbf{P}$, then the derivative, $\frac{\partial \mathbf{P}}{\partial \mathbf{W}}$ must vanish, i.e,

$$\mathbf{W}[\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^T] - \mathbf{B}\mathbf{H}^T = 0,$$

therefore the optimal $\mathbf{W}$ is given by,

$$\mathbf{W} = \mathbf{B}\mathbf{H}^T[\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^T]^{-1}. \tag{2.10}$$

Note that using the above result, the error covariance matrix is given by,

$$\mathbf{P} = \mathbf{B} - \mathbf{B}\mathbf{H}^T[\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^T]^{-1}\mathbf{H}\mathbf{B}, \tag{2.11}$$

since both $\mathbf{B}$ and $\mathbf{R}$ are symmetric and positive definite, all the terms in the second expression of the equation are also positive definite, therefore $\mathbf{P} \leqslant \mathbf{B}$, in other words, the uncertainty in estimation of the optimal estimate $\mathbf{x}^*$ decreases after every estimation step. To summarize, in OI, the optimal estimate is given by (using (2.7) and (2.10)),

$$\mathbf{x}^* = \mathbf{x}^b + \mathbf{B}\mathbf{H}^T[\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^T]^{-1}\left(\mathbf{y}^o - H[\mathbf{x}^b]\right), \tag{2.12}$$

and its error covariance is given by equation (2.11), see Eliassen (1954) [25], Gandin (1963) [26] and Daley (1991) [27] for further details on OI.

2.2.2 Variational data assimilation methods: 3D-Var

The objective of variational data assimilation (`VDA`) methods is to identify an estimate of the state that fits simultaneously the background field and the observations, given their respective degree of uncertainty (or degree of confidence or precision), i.e, the inverses of

their error covariances, $\mathbf{B}^{-1}$ and $\mathbf{R}^{-1}$ respectively. Such an estimate is obtained efficiently by minimization of a least squares cost functional.

One of the approaches to derive the VDA methods is based on maximization of the a-priori probability distribution (also called maximum likelihood approach, Edwards, 1984 [28]), assuming Gaussian probability distribution. Given a background field (or a-priori estimate) x^b, the probability of finding the true state given this is, $p_{\mathbf{B}}(\mathbf{x}|\mathbf{x}^b)$, similarly, given the observations, the probability is equal to $p_{\mathbf{R}}(\mathbf{x}|\mathbf{y}^o)$. Assuming a Gaussian distribution of the background and observational errors the respective probabilities are given by,

$$p_{\mathbf{B}}(\mathbf{x}|\mathbf{x}^b) = \frac{1}{(2\pi)^{n/2}\|\mathbf{B}\|^{1/2}} e^{-1/2(\mathbf{x}-\mathbf{x}^b)^T\mathbf{B}^{-1}(\mathbf{x}-\mathbf{x}^b)}, \tag{2.13}$$

$$p_{\mathbf{R}}(\mathbf{x}|\mathbf{y}^o) = \frac{1}{(2\pi)^{n/2}\|\mathbf{R}\|^{1/2}} e^{-1/2(\mathbf{y}^o-H[\mathbf{x}])^T\mathbf{R}^{-1}(\mathbf{y}^o-H[\mathbf{x}])}, \tag{2.14}$$

where $\|\cdot\|$ denotes the norm of a matrix (usually we consider the Frobenius norm). Since the background and observational fields are two independent sources, the most likely estimate of the state of the atmosphere is given by the joint probability of the above two probabilities, i.e, the following product,

$$p_{\mathbf{B}}(\mathbf{x}|\mathbf{x}^b)\; p_{\mathbf{R}}(\mathbf{x}|\mathbf{y}^o) = \frac{1}{(2\pi)\|\mathbf{B}\|^{1/2}\|\mathbf{R}\|^{1/2}} e^{-[1/2(\mathbf{x}-\mathbf{x}^b)^T\mathbf{B}^{-1}(\mathbf{x}-\mathbf{x}^b)+1/2(\mathbf{y}^o-H[\mathbf{x}])^T\mathbf{R}^{-1}(\mathbf{y}^o-H[\mathbf{x}])]}.$$

Maximization of the above joint probability is the same as minimization of the following quantity in the exponent (note the negative sign in the exponentiation),

$$1/2(\mathbf{x}-\mathbf{x}^b)^T\mathbf{B}^{-1}(\mathbf{x}-\mathbf{x}^b) + 1/2(\mathbf{y}^o-H[\mathbf{x}])^T\mathbf{R}^{-1}(\mathbf{y}^o-H[\mathbf{x}]).$$

The above functional represents a sum of least-squared difference between the state and the background and observational fields, weighted by their respective inverse error covariances. We will use the following notation to distinguish between the two different contributions, the *background cost functional* is denoted by, J^b, and the *observational cost functional* is denoted by J^o, so that

$$J(\mathbf{x}) = \underbrace{1/2(\mathbf{x}-\mathbf{x}^b)^T\mathbf{B}^{-1}(\mathbf{x}-\mathbf{x}^b)}_{J^b} + \underbrace{1/2(\mathbf{y}^o-H[\mathbf{x}])^T\mathbf{R}^{-1}(\mathbf{y}^o-H[\mathbf{x}])}_{J^o}. \tag{2.15}$$

In order to minimize the above quadratic cost functional, we would need its gradient and Hessian with respect to the state vector, i.e., $\nabla_{\mathbf{x}}J$ and $\frac{\partial^2 J}{\partial \mathbf{x}^2}$ respectively. At the minima, $\mathbf{x}^*$ of $J(\mathbf{x})$, $\nabla_{\mathbf{x}}J(\mathbf{x}^*) = 0$ and Hessian evaluated at $\mathbf{x}^*$, $\frac{\partial^2 J}{\partial \mathbf{x}^2}|_{\mathbf{x}^*}$ is positive definite.

Let $\mathbf{H}$ denote a linearization of the observational operator, such that $\mathbf{H} = \frac{\partial H}{\partial x}$. Then

$$\mathbf{y}^o - H[\mathbf{x}] = \mathbf{y}^o - H[\mathbf{x}^b + (\mathbf{x} - \mathbf{x}^b)] = \mathbf{y}^o - H[\mathbf{x}^b] - \mathbf{H}(\mathbf{x} - \mathbf{x}^b).$$

Using the above relationship we can J^o as, $J^o =$
$1/2(\mathbf{y}^o - H[\mathbf{x}^b] - \mathbf{H}(\mathbf{x} - \mathbf{x}^b))^T\mathbf{R}^{-1}(\mathbf{y}^o - H[\mathbf{x}^b] - \mathbf{H}(\mathbf{x} - \mathbf{x}^b)) =$
$1/2(\mathbf{y}^o - H[\mathbf{x}^b])^T\mathbf{R}^{-1}(\mathbf{y}^o - H[\mathbf{x}^b]) - 1/2(\mathbf{y}^o - H[\mathbf{x}^b])^T\mathbf{R}^{-1}\mathbf{H}(\mathbf{x} - \mathbf{x}^b)$
$- 1/2(\mathbf{H}(\mathbf{x} - \mathbf{x}^b))^T\mathbf{R}^{-1}(\mathbf{y}^o - H[\mathbf{x}^b]) + 1/2(\mathbf{H}(\mathbf{x} - \mathbf{x}^b))^T\mathbf{R}^{-1}\mathbf{H}(\mathbf{x} - \mathbf{x}^b).$

The above cost functional is quadratic in $(\mathbf{x} - \mathbf{x}^b)$, and

$$\nabla_{\mathbf{x}} J^o = -\mathbf{H}^T\mathbf{R}^{-1}(\mathbf{y}^o - H[\mathbf{x}^b]) + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}(\mathbf{x} - \mathbf{x}^b).$$

Therefore

$$\begin{aligned} \nabla_{\mathbf{x}} J &= \nabla_{\mathbf{x}} J^b + \nabla_{\mathbf{x}} J^o \\ &= \mathbf{B}^{-1}(\mathbf{x} - \mathbf{x}^b) + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}(\mathbf{x} - \mathbf{x}^b) - \mathbf{H}^T\mathbf{R}^{-1}(\mathbf{y}^o - H[\mathbf{x}^b]) \\ &= [\mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}](\mathbf{x} - \mathbf{x}^b) - \mathbf{H}^T\mathbf{R}^{-1}(\mathbf{y}^o - H[\mathbf{x}^b]). \end{aligned} \tag{2.16}$$

For the minima, $\nabla_{\mathbf{x}} J(\mathbf{x}^*) = 0$, which implies that

$$[\mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}](\mathbf{x}^* - \mathbf{x}^b) = \mathbf{H}^T\mathbf{R}^{-1}(\mathbf{y}^o - H[\mathbf{x}^b]).$$

therefore,

$$\mathbf{x}^* = \mathbf{x}^b + [\mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}]^{-1}\mathbf{H}^T\mathbf{R}^{-1}(\mathbf{y}^o - H[\mathbf{x}^b]). \tag{2.17}$$

Using the relationship for the gradient of the cost functional with respect to the state vector, the Hessian is given by,

$$\frac{\partial^2 J}{\partial \mathbf{x}^2}|_{\mathbf{x}^*} = \mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}. \tag{2.18}$$

Since $\mathbf{B}$ and $\mathbf{R}$ are symmetric positive definite, the Hessian evaluated at $\mathbf{x}^*$ is positive definite, which implies that $\mathbf{x}^*$ furnishes a minimum of the cost functional. Though the formal solution of the 3D-Var estimation problem is given by equation (2.17), it is always obtained by direct minimization of the cost functional $J(\mathbf{x})$ using the adjoint model and large scale iterative minimization methods (to be described in detail, late in this chapter); see Vandenberghe and Kuo (1999) [29] for mesoscale model version 5 (MM5) 3D-Var implementation, Courtier *et al.* (1998) [30] and Andersson *et al.* (1998) [31] and references therein for formulation and implementation details of 3D-Var in a complex European center for medium-range-weather forecast (ECMWF) system.

2.2.3 Equivalence between OI and 3D-Var

Let us recall that the optimal estimate of the state using OI was given by equation (2.12),

$$\mathbf{x}^* = \mathbf{x}^b + \mathbf{B}\mathbf{H}^T[\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^T]^{-1}\,(\mathbf{y}^o - H[\mathbf{x}^b]),$$

and using 3D-Var formulation, the formal solution was given by (2.17),

$$\mathbf{x}^* = \mathbf{x}^b + [\mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}]^{-1}\mathbf{H}^T\mathbf{R}^{-1}(\mathbf{y}^o - H[\mathbf{x}^b]).$$

Though it is not obvious that both of these results are exactly the same, it has been proven in literature that are indeed equivalent. To show that they are equivalent, it is enough to prove that

$$[\mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}]^{-1}\mathbf{H}^T\mathbf{R}^{-1} = \mathbf{B}\mathbf{H}^T[\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^T]^{-1}.$$

For this purpose, we will use the following Sherman-Morrison-Woodbury (SMW) formula [32], which states that for any invertible matrix, Λ and any matrices Γ, Θ,

$$[\Lambda + \Gamma\Theta]^{-1} = \Lambda^{-1} - \Lambda^{-1}\Gamma[I + \Theta\Lambda^{-1}\Gamma]^{-1}\Theta\Lambda^{-1}.$$

In the SMW formula if $\Lambda = \mathbf{B}^{-1}$, $\Gamma = \mathbf{H}^T$, and $\Theta = \mathbf{R}^{-1}\mathbf{H}$, and assuming that $\mathbf{H}$ is unitary, then

$$[\mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}]^{-1} = \mathbf{B} - \mathbf{B}\mathbf{H}(I + \mathbf{R}^{-1}\mathbf{H}\mathbf{B}\mathbf{H}^T)^{-1}\mathbf{R}^{-1}\mathbf{H}\mathbf{B}.$$

Therefore $[\mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}]^{-1}\mathbf{H}^T\mathbf{R}^{-1}$
$= [\mathbf{B} - \mathbf{B}\mathbf{H}(I + \mathbf{R}^{-1}\mathbf{H}\mathbf{B}\mathbf{H}^T)^{-1}\mathbf{R}^{-1}\mathbf{H}\mathbf{B}]\mathbf{H}^T\mathbf{R}^{-1}$
$= \mathbf{B}\mathbf{H}^T\mathbf{R}^{-1} - \mathbf{B}\mathbf{H}^T[I + \mathbf{R}^{-1}\mathbf{H}\mathbf{B}\mathbf{H}^T]^{-1}\mathbf{R}^{-1}\mathbf{H}\mathbf{B}\mathbf{H}^T\mathbf{R}^{-1}$
$= \mathbf{B}\mathbf{H}^T[\mathbf{R}^{-1} - (I + \mathbf{R}^{-1}\mathbf{H}\mathbf{B}\mathbf{H}^T)^{-1}\mathbf{R}^{-1}\mathbf{H}\mathbf{B}\mathbf{H}^T\mathbf{R}^{-1}]$
$= \mathbf{B}\mathbf{H}^T[\mathbf{R}^{-1} - \mathbf{R}^{-1}\mathbf{H}(I + \mathbf{B}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})^{-1}\mathbf{B}\mathbf{H}^T\mathbf{R}^{-1}]$. Another application of the SMW formula using $\Lambda = \mathbf{R}$, $\Gamma = \mathbf{H}$, and $\Theta = \mathbf{B}\mathbf{H}^T$, gives us

$$[\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^T]^{-1} = \mathbf{R}^{-1} - \mathbf{R}^{-1}\mathbf{H}(I + \mathbf{B}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})^{-1}\mathbf{B}\mathbf{H}^T\mathbf{R}^{-1}.$$

Thus

$$[\mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}]^{-1}\mathbf{H}^T\mathbf{R}^{-1} = \mathbf{B}\mathbf{H}^T[\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^T]^{-1},$$

which is the desired result. Also using the above results, note that the inverse of the 3D-Var cost functional Hessian matrix is the same as the OI error covariance matrix in (2.11),

$[\frac{\partial^2 J}{\partial \mathbf{x}^2}|_{\mathbf{x}^*}]^{-1}$

$= [\mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}]^{-1}$

$= \mathbf{B} - \mathbf{BH}(I + \mathbf{R}^{-1}\mathbf{HBH}^T)^{-1}\mathbf{R}^{-1}\mathbf{HB}$

$= \mathbf{B} - \mathbf{BH}[\mathbf{R} + \mathbf{HBH}^T]^{-1}\mathbf{HB}$

$= \mathbf{P}.$

For other ways to derive the same result, please see Lorenc (1986) [12] and Courtier (1997) [33]. Thus the formal solution to the 3D-Var problem is the same as that obtained by using OI. However as described earlier, the procedure followed to obtain the solution using these two methods is very different. Practical implementation of OI requires a number of approximations to be introduced, such as the background error covariance matrix, $\mathbf{B}$ has to be locally approximated. All the available data through observations cannot be used simultaneously. On the other hand, practical implementation of 3D-Var involves minimization of a cost functional using large scale minimization methods (for computational reasons, about 100 minimization iterations are usually performed, Courtier 1997 [34]) and due to this reason many of the simplifying assumptions and approximations required by OI are not needed any more. In particular the background error statistics can be effectively used to obtain a better estimate of the state, and all the available observational data can be used simultaneously; see Kalnay [1] for further discussion on this topic. Therefore 3D-Var replaced OI as the operational scheme of choice at almost all the major weather forecasting centers by the late 1990's (during the 1980s to mid 1990s OI was being used).

2.2.4 Physical space analysis system (PSAS)

The PSAS is another `VDA` method which is closely related to both 3D-Var and OI, and was introduced by Da Silva *et al.* (1995) [35]. Just as in the case of the 3D-Var method, the optimal estimate of the state is obtained by minimization of a cost functional which is defined in the (physical) space of the observations (whereas in 3D-Var the cost functional was defined in the model state space, $\mathbf{x}$). If the size of the observations vector, $\mathbf{y}^o$, is much smaller than that of the state vector, $\mathbf{x}$, then the dimension in which the minimization is carried out to solve the PSAS problem is significantly smaller than that for the 3D-Var problem.

Let $\delta\mathbf{x}$ denote the increment to the background field, $\mathbf{x}^b$ to obtain the optimal estimate of the state vector, $\mathbf{x}$, such that,

$$\mathbf{x} = \mathbf{x}^b + \delta\mathbf{x},$$

and let $\mathbf{d}$ denote the following *residue* in the observational space,

$$\mathbf{d} = \mathbf{y}^o - \mathbf{H}\mathbf{x}^b,$$

where $\mathbf{H}$ is linear approximation of the observation operator in the vicinity of $\mathbf{x}^b$. In the PSAS formulation, we formally minimize the following quadratic cost functional,

$$J(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T \underbrace{[\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^T]}_{\mathbf{C}}\mathbf{w} - \mathbf{w}^T\mathbf{d}, \tag{2.19}$$

note that in the above equation, the cost functional is defined entirely in the observational space (due to the action of $\mathbf{H}$ and $\mathbf{H}^T$, on $\mathbf{B}$, $\mathbf{C}$ is also in the observational space). The gradient of the above cost functional is given by

$$\nabla_{\mathbf{w}} J(\mathbf{w}) = [\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^T]\mathbf{w} - \mathbf{d}.$$

At the minimum, the above gradient must vanish, i.e., $\nabla_{\mathbf{w}} J(\mathbf{w}) = 0$. Which implies that

$$[\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^T]\mathbf{w} = \mathbf{d}. \tag{2.20}$$

In practice it is the above system of linear equations (2.20) that is solved (for example, using preconditioned conjugate gradient method) to obtain $\mathbf{w}$ rather than direct minimization of the cost functional given in equation (2.19). Once $\mathbf{w}$ has been obtained, the increment to the background field, $\delta\mathbf{x}$ is given by,

$$\delta\mathbf{x} = \mathbf{B}\mathbf{H}^T\mathbf{w},$$

thus we obtain the estimate of the optimal state. For further details on PSAS and its relationship to OI and 3D-Var please see Courtier (1997) [33], and for operational implementation details, see da Silva and Guo (1996) [36] and Cohn *et al.* (1998) [37].

2.2.5 4D-Var

Four dimensional (three space dimensions plus time dimension) variational data assimilation, or 4D-Var, can be considered as an extension to the 3D-Var method. All the important `DA` methods considered thus-far (OI, 3D-Var and PSAS) neglected time as a dimension, whereas 4D-Var allows for inclusion of the temporal dimension. In particular, observational data can be considered as distributed within a time interval (t_0, t_n), which is indeed the

case in reality. Thus, rather than having one observational vector, $\mathbf{y}^o$ to *represent* all the observations gathered within an interval of time, in 4D-Var, we have a sequence of temporally distributed observations: $\mathbf{y}^o(t_i),\ t_i \in (t_0, t_n)$. The time interval under consideration, (t_0, t_n) is called the time window of data assimilation. Associated with each of these observations, available at a certain time, t_i, we have the observational error covariance, $\mathbf{R}(t_i) = \mathbf{R}_i$ (therefore we could account for observational errors to evolve in time, this is an important issue and is further discussed in Chapter 4).

The 4D-Var formulation provides an optimal estimate of the trajectory followed by the true evolution of the atmosphere (or ocean) in a time interval, using the available observations, $\mathbf{y}^o(t_i)$ and their respective error covariances, $\mathbf{R}_i$, the background (or, first guess) field, $\mathbf{x}^b$ and the associated background error covariance, $\mathbf{B}$, and the numerical (forecast) model, symbolically represented by $\mathcal{M}$. Recall from equation (2.1), the state at t_{i+1} is obtained using the forecasting model and the state at the previous time step, t_i,

$$\mathbf{x}(t_{i+1}) = \mathcal{M}[\mathbf{x}(t_i)] + \boldsymbol{\eta}(t_i),$$

where $\mathcal{M}[.]$ represented the discrete model' s dynamics operator and $\boldsymbol{\eta}(t_i)$ represented the model error at time step, t_i. The following derivation of the 4D-Var method is provided by neglecting the model error, in other words, assuming that the model is *perfect*. Such a formulation of the 4D-Var is called *strong constraint* 4D-Var; issues related to accounting for model error are very crucial to the subject of `VDA`, and these are addressed in detail, in Chapter 4. A least-squares cost functional similar to that in equation (2.15) is minimized (subject to the forecast model as a constraint) in 4D-Var, where the observations are distributed in time,

$$J[\mathbf{x}(t_0)] = \underbrace{1/2[\mathbf{x}(t_0) - \mathbf{x}^b]^T \mathbf{B}^{-1}[\mathbf{x}(t_0) - \mathbf{x}^b]}_{J^b} + \underbrace{1/2 \sum_{i=0}^{n} [H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)]^T \mathbf{R}_i^{-1} [H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)]}_{J^o}, \tag{2.21}$$

note that the second component of the cost functional, J^o, describes the sum of least-squared differences between each of the observations and the output from the forecast model (considered in the observational space) over the entire data assimilation time window: (t_0, t_n). As in the 3D-Var cost functional, the background cost functional term, J^b, measures the least-squares distance between the background field and model initial state, $\mathbf{x}(t_0)$. The relevance

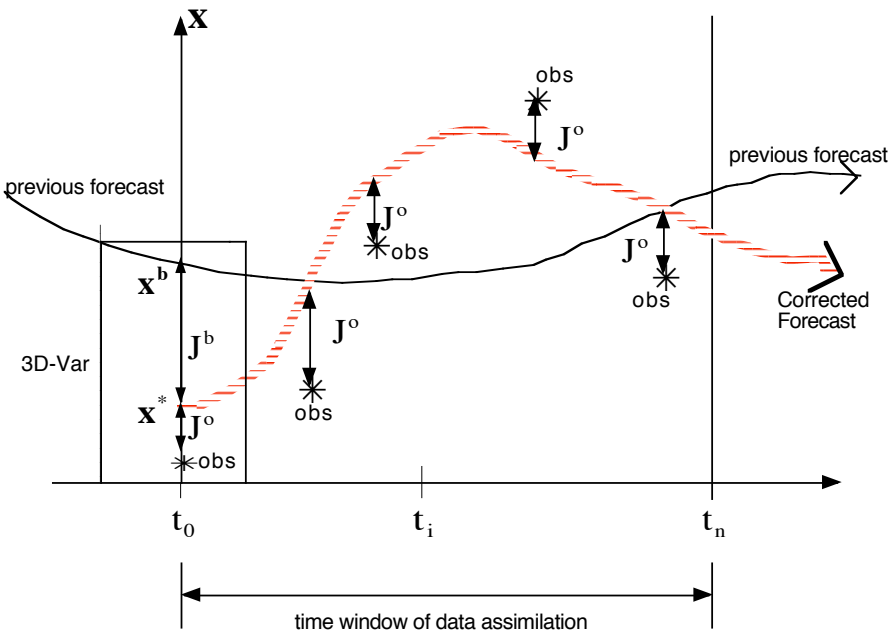

Figure 2.1: Illustration of 4D-Var in the phase plane. Using information from the previous forecast ($\mathbf{x}^b$) and most recently available observations within a time interval, a new *optimal* $\mathbf{x}^*$ is obtained via 4D-Var so that the corrected forecast (new model trajectory) closely fits the available and subsequent observations. Also see that neglecting the time dimension reduces 4D-Var to 3D-Var.

of such a quadratic cost functional is shown in the fig. 2.1. The *control variable*, or the the variable with respect to which the above cost functional is minimized is the initial state of the model, $\mathbf{x}(t_0)$, with in the data assimilation time window. Through minimization of $J[\mathbf{x}(t_0)]$, the goal of strong constraint 4D-Var is to obtain an *optimal initial state* $\mathbf{x}^*$ (and other model parameters), which when used as initial conditions for the forecast model, the model output is able to fit not only the observations within (t_0, t_n), but also for time, $t > t_n$, i.e., forecast for longer periods of time with more reliability and accuracy. This procedure of implementing 4D-Var within each data assimilation time window (which is typically 12- 24 hours long) is carried out cyclically at major weather forecasting centers [1, 38]; see flowchart in appendix A).

In order to minimize the cost functional in (2.21) with respect to $\mathbf{x}(t_0)$, we need to be able to calculate the gradient of the cost functional with respect to the control variable, i.e., $\nabla_{\mathbf{x}(t_0)} J$, and ensure that the Hessian of the cost functional, evaluated at the minimum is

positive definite. Since the cost functional is convex by construction and the background and observational error covariances, $\mathbf{B}$ and $\mathbf{R}$ are symmetric positive definite (by definition), the Hessian of the cost functional is always positive definite (if there were to be discontinuities in the state variables, then we would have to convexify such a 4D-Var cost functional), for further details, please see [39] and references therein (here onwards, unless otherwise specified, it is assumed that the observational error covariance matrices are *fixed* in time, i.e., $\mathbf{R}_i = \mathbf{R}$).

The gradient of the 4D-Var cost functional with respect to the control variables is efficiently provided by the *adjoint model*, following derivation details its mathematical formulation (another approach to obtain the gradient will be described in Chapter 4). Let us consider a first order variation of the above cost functional, by introducing a small change $\delta\mathbf{x}(t_0)$, in the control variable, $\mathbf{x}(t_0)$,

$$\delta J = J[\mathbf{x}(t_0) + \delta\mathbf{x}(t_0)] - J[\mathbf{x}(t_0)] \approx [\frac{\partial J}{\partial \mathbf{x}(t_0)}]^T \delta\mathbf{x}(t_0) = [\nabla_{\mathbf{x}(t_0)} J]^T \delta\mathbf{x}(t_0),$$

where $[\nabla_{\mathbf{x}(t_0)} J]^T \delta\mathbf{x}(t_0)$ is the directional derivative of J in the direction of $\delta\mathbf{x}(t_0)$. It is also the Gâteaux derivative (note that in the finite-dimensional case, such as the one we have here, the Fréchet derivative is equivalent to the gradient, whereas the Gâteaux derivative is equivalent to the directional derivative [40]). The gradient, $\nabla_{\mathbf{x}(t_0)} J$ is given by,

$$\nabla_{\mathbf{x}(t_0)} J = \nabla_{\mathbf{x}(t_0)} J^b + \nabla_{\mathbf{x}(t_0)} J^o, \tag{2.22}$$

and gradient $\nabla_{\mathbf{x}(t_0)} J^b$ is obtained trivially,

$$\nabla_{\mathbf{x}(t_0)} \mathcal{J}^b = \mathbf{B}^{-1}(\mathbf{x}(t_0) - \mathbf{x}^b), \tag{2.23}$$

whereas obtaining $\nabla_{\mathbf{x}(t_0)} J^o$ is relatively involved, because model state at time, t_i is obtained by integrating the forecast model from initial time to t_i,

$$\mathbf{x}(t_i) = \mathcal{M}(t_0, t_i)[\mathbf{x}(t_0)],$$

and therefore J^o implicitly depends on $\mathbf{x}(t_0)$. However, using the chain rule, for a functional $K = \frac{1}{2}\mathbf{aDa}$, where the vector $\mathbf{a}$ is dependent on the vector, $\mathbf{b}$, via $\mathbf{a} = \mathbf{a}(\mathbf{b})$, the gradient of K with respect to $\mathbf{b}$ is given by,

$$\nabla_{\mathbf{b}} K = [\frac{\partial \mathbf{a}}{\partial \mathbf{b}}]^T (D\mathbf{a}).$$

In the case of the observational cost functional, J^o, $\mathbf{b} = \mathbf{x}(t_0)$, $\mathbf{D} = \mathbf{R}^{-1}$, and $\mathbf{a} = [H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)]$. Therefore we first set out to find

$$\frac{\partial \mathbf{a}}{\partial \mathbf{b}} = \frac{\partial [H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)]}{\partial \mathbf{x}(t_0)}.$$

Once again, let us consider a small perturbation $\delta\mathbf{x}(t_0)$, of the initial state, $\mathbf{x}(t_0)$. Then

$$\mathbf{x}'(t_i) = \mathcal{M}(t_0, t_i)[\mathbf{x}(t_0) + \delta\mathbf{x}(t_0)],$$

which can be expanded to yield, $\mathbf{x}'(t_i)$
$= \mathcal{M}(t_0, t_i)[\mathbf{x}(t_0) + \delta\mathbf{x}(t_0)]$
$= \mathcal{M}(t_0, t_i)[\mathbf{x}(t_0)] + [\frac{\partial \mathcal{M}}{\partial \mathbf{x}(t_0)}(t_0, t_i)]\delta\mathbf{x}(t_0) + O[\delta\mathbf{x}(t_0)^T\, \delta\mathbf{x}(t_0)]$
$= \mathbf{x}(t_i) + [\frac{\partial \mathcal{M}}{\partial \mathbf{x}(t_0)}(t_0, t_i)]\delta\mathbf{x}(t_0) + O[\delta\mathbf{x}(t_0)^T\, \delta\mathbf{x}(t_0)]$.
If we consider an approximation up-to first order only and using the above result, we obtain,

$$\delta\mathbf{x}(t_i) = \mathbf{x}'(t_i) - \mathbf{x}(t_i) \approx [\frac{\partial \mathcal{M}}{\partial \mathbf{x}(t_0)}(t_0, t_i)]\delta\mathbf{x}(t_0).$$

Therefore we have obtained an equation to describe the evolution of the perturbed state, $\delta\mathbf{x}$ from t_0 to final time, t_n. The linearized operator: $\frac{\partial \mathcal{M}}{\partial \mathbf{x}(t_0)}$ is denoted by $\mathbf{L}$ and is called the tangent linear model (TLM). When the TLM is integrated from $t_0 \rightarrow t_n$ using $\delta\mathbf{x}(t_0)$, as initial condition, yields, $\delta\mathbf{x}(t_n)$,

$$\delta\mathbf{x}(t_n) = \mathbf{L}(t_0, t_n)[\delta\mathbf{x}(t_0)]. \tag{2.24}$$

The TLM is also the Jacobian of the nonlinear model with respect to the initial conditions, therefore it can be utilized to study the evolution of *uncertainties* in the initial state, $\mathbf{x}(t_0)$. Therefore we obtain the following result,

$$\delta\mathbf{x}(t_n) = \frac{\partial \mathbf{x}(t_n)}{\partial \mathbf{x}(t_0)}.$$

Computationally it is obtained by linearization of every nonlinear statement (or every segment of the programmed code) in the nonlinear (forecast) model, further details on the TLM are provided in Chapter 3.

Using the above derived result and equation (2.24),

$$\frac{\partial [H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)]}{\partial \mathbf{x}(t_0)} = \frac{\partial H}{\partial \mathbf{x}(t_i)}\frac{\partial \mathbf{x}(t_i)}{\partial \mathbf{x}(t_0)} = \frac{\partial H}{\partial \mathbf{x}(t_i)}\mathbf{L}(t_0, t_i). \tag{2.25}$$

Let $\mathbf{H}_i$ be a linearization of the observations operator around the state, $\mathbf{x}(t_i)$ (in other words, let $\mathbf{H}_i$ be the linearized Jacobian, $\frac{\partial H}{\partial \mathbf{x}(t_i)}$) then

$$\frac{\partial[H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)]}{\partial \mathbf{x}(t_0)} = \mathbf{H}_i \mathbf{L}(t_0, t_i).$$

Since the TLM, $\mathbf{L}$, is a linear operator, we can write $\mathbf{L}(t_0, t_i)$ as a continuous product of a series of linear operators, each operating from one time step to the next, i.e.,

$$\mathbf{L}(t_0, t_i) = \mathbf{L}(t_0, t_1)\mathbf{L}(t_1, t_2) \ldots \mathbf{L}(t_{i-1}, t_i) = \prod_{j=0}^{j=i-1} \mathbf{L}(t_j, t_{j+1}). \tag{2.26}$$

Thus we now have all the ingredients needed to obtain the gradient of the observational cost, J^o with respect to $\mathbf{x}(t_0)$,

$$\nabla_{\mathbf{x}(t_0)} J^o = \sum_{i=0}^{n} [\frac{\partial[H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)]}{\partial \mathbf{x}(t_0)}]^T \mathbf{R}^{-1} [H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)].$$

Using the above result and equations (2.25 and 2.26), we obtain

$$\begin{aligned} \nabla_{\mathbf{x}(t_0)} J^o &= \sum_{i=0}^{n} [\mathbf{H}_i \, \mathbf{L}(t_0, t_i)]^T \mathbf{R}^{-1} [H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)] \\ &= \sum_{i=0}^{n} [\mathbf{L}(t_0, t_i)]^T \mathbf{H}_i^T \mathbf{R}^{-1} [H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)]. \end{aligned} \tag{2.27}$$

From (2.26), $[\mathbf{L}(t_0, t_i)]^T = [\mathbf{L}(t_0, t_1)\mathbf{L}(t_1, t_2) \ldots \mathbf{L}(t_{i-1}, t_i)]^T$
$= \mathbf{L}^T(t_{i-1}, t_i)\mathbf{L}^T(t_{i-2}, t_{i-1}) \ldots \mathbf{L}^T(t_1, t_2)\, \mathbf{L}^T(t_0, t_1)$.
It is obvious from the above equation that we have a product of a sequence of linear operators, which are transposes of their corresponding operators in the TLM. Also the time index in the above continuous product is starting at the final time, t_i and ending at the initial time, t_0. The above transposed operator, $[\mathbf{L}(t_0, t_i)]^T$ is the so-called adjoint model (ADM), and it is computationally obtained by transposing each and every linear statement in the TLM, written *backwards* in time (once again, please see Chapter 3 for further details). We can rewrite the ADM as a sequence of product of linear operators, each of which is *flowing* backwards in time,

$$[\mathbf{L}(t_0, t_i)]^T = \mathbf{L}^T(t_i, t_{i-1})\mathbf{L}^T(t_{i-1}, t_{i-2}) \ldots \mathbf{L}^T(t_2, t_1)\, \mathbf{L}^T(t_1, t_0) = \prod_{j=i-1}^{0} \mathbf{L}^T(t_{j+1}, t_j) = \mathbf{L}^T(t_i, t_0). \tag{2.28}$$

Thus

$$\nabla_{\mathbf{x}(t_0)} J^o = \sum_{i=0}^{n} \mathbf{L}^T(t_i, t_0)\, \mathbf{H}_i^T\, \mathbf{R}^{-1}\, [H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)], \tag{2.29}$$

and

$$\nabla_{\mathbf{x}(t_0)} J = \nabla_{\mathbf{x}(t_0)} J^b + \nabla_{\mathbf{x}(t_0)} J^o = \mathbf{B}^{-1}(\mathbf{x}(t_0) - \mathbf{x}^b) + \sum_{i=0}^{n} \mathbf{L}^T(t_i, t_0)\, \mathbf{H}_i^T\, \mathbf{R}^{-1}\, [H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)]. \tag{2.30}$$

The above equation shows that in every iteration of 4D-Var cost functional (2.21) minimization, we need to calculate the gradient (2.30), i.e., compute the increments $[H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)]$ at the observation times, t_i during a forward integration of the forecast model, operate on them by $\mathbf{H}_i^T\, \mathbf{R}^{-1}$ and integrate these weighted increments backwards in time, to the initial time, t_0 using the ADM. Since many parts of the backward ADM integration are common to several time intervals, various implementations of the above procedure (obtained by rearranging the above equation) have been suggested in literature (such as the incremental form of 4D-Var, cycling 4D-Var, also the method of *representers*), see [1, 34, 41, 42] and references therein for further details on such methods. Therefore implementation of the 4D-Var algorithm involves a sequence of calls to a minimization algorithm, which uses the functional and gradient information, and in an iterative mechanism, yields the optimum value of $\mathbf{x}^* = \mathbf{x}(t_0)$ which provides the least value of the cost functional, $J(\mathbf{x}^*)$.

Due to the complex nonlinear nature of the forecast models, the development of the tangent linear and adjoint models is extremely complicated, time consuming and laborious (these issues are elaborated in Chapter 3). A few *automatic differentiation* (AD) tools are now available to help speed-up the task of generating the tangent linear and adjoint models. Some of the popular AD softwares are: tangent linear and adjoint model compiler (TAMC [43]), automatic differentiation of Fortran (ADIFOR [44]), Odyssée and Tapenade [45] (we have used TAMC in our studies). Coupled with a suitable large scale nonlinear minimization method, and using the background field as the first guess for minimization, the method of 4D-var provides an optimal estimate of evolution of the true atmosphere inside the time window of assimilation. In the next and final section of this chapter, we will briefly introduce the main concepts of minimization methods.

2.3 An overview of nonlinear minimization methods

Here we are interested in the following minimization problem,

$$\min_{\mathbf{x}} \; f(\mathbf{x}),$$

where $\mathbf{x}$ is the vector of control variables, f is the *objective* or cost functional, which we desire to minimize with respect to the control variables. Some times we have equality and, or inequality constraints imposed on all or a few components $\mathbf{x}$. In that case, we have a constrained minimization problem. Since in this dissertation work, we deal with unconstrained minimization problems only, here onwards, unless otherwise specified, there are no constraints on the control variables (only a comprehensive summary of the minimization methods has been provided over here, for exhaustive details, please see Nocedal and Wright (1999) [46], Nash and Sofer (1996) [47]).

If the functional f is linear in $\mathbf{x}$, such as, $f(\mathbf{x}) = \mathbf{a}^T\mathbf{x} + b$, where $\mathbf{a}$ and b are two constants, then the minimization method used to obtain the minimum of f is called a linear optimization or linear minimization method, for example, the simplex method attempts to minimize such linear functionals. On the other hand, here we will deal with nonlinear functionals of the form, $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\,\mathbf{A}\,\mathbf{x} + \mathbf{b}^T\,\mathbf{x} + c$, where $\mathbf{A}, \mathbf{b}$ and c are constants. Such an objective functional is convex (or, quadratic) in $\mathbf{x}$. We will assume that the functional is continuously differentiable all over the domain of $\mathbf{x}$ (usually, $\mathbf{x} \in \mathbb{R}^n$).

In order to be able to obtain the minimum of the above functional, we first need to characterize nature of the minimum we are interested in finding.

Def.: Global Minimum

A point $\mathbf{x}^*$ is a global minimizer, if $f(\mathbf{x}^*) \leqslant f(\mathbf{x}),\ \forall \mathbf{x}$.

Not all functions (or functionals) have a finite global minimizer, and if even there exists a global minimizer, there is no guarantee that we should be able to calculate it in a practical manner, with the available computational resources in realistic time (particularly for the complex functionals that we come across in real life applications). Hence it would be very satisfying, at-least theoretically, and certainly for practical reasons too, if were able to find global minimizers. However, many of the popular nonlinear optimization methods devised thus-far are based on the Taylor series expansion, hence they are based on the information about the functional, available at a single point and approximation of this information within

a small neighborhood of this point. Therefore using such methods it, without any additional information or assumptions about the problem, it would not be possible to identify global minimizers using such methods which are based on Taylor series expansions. Fortunately, by construction, the 4D-Var cost functional in equation (2.21) was shown to be convex, hence we should be able to find such global minimizers, still using a method based on Taylor series approximations; in-fact for such a cost functional, the local minimizer is the same as the global minimizer (as explained below). Following are some of the used formal definitions and proof of the above statement.

Def.: Local Minimum

A point $\mathbf{x}^*$ is a local minimizer if there is a neighborhood, $\mathcal{S}$ of $\mathbf{x}^*$ such that $f(\mathbf{x}^*) \leqslant f(\mathbf{x}),\ \forall \mathcal{S}$, more precisely, $\|\mathbf{x} - \mathbf{x}^*\| \leqslant \epsilon$.

The necessary and sufficient conditions for optimality are based on the assumption that $\mathbf{x}^*$ is a local minimizer of an objective functional.

Def.: First-order necessary condition

If $\mathbf{x}^*$ is a local minimizer and f is continuously differentiable in an open neighborhood of $\mathbf{x}^*$, then $\nabla f(\mathbf{x}^*) = 0$.

Any local minimizer, $\mathbf{x}^*$, must also be a stationary point, i.e., a point where the gradient of the functional, $\nabla f(\mathbf{x}^*) = 0$.

Def.: Second-order criteria

For $\mathbf{x}^*$ to be a local minimizer of f, assuming that $\nabla^2 f$ is continuous in an open neighborhood of $\mathbf{x}^*$, it is necessary that $\nabla f(\mathbf{x}^*) = 0$ and $\nabla^2 f(\mathbf{x}^*)$ be positive semi-definite; for sufficiency we require $\nabla f(\mathbf{x}^*) = 0$ and $\nabla^2 f(\mathbf{x}^*)$ be positive definite.

We will often be using the following form of Taylor series, in n-dimensions.

Taylor' s theorem:

$$f(\mathbf{x}_0 + \mathbf{p}) = f(\mathbf{x}_0) + \mathbf{p}^T \nabla f(\mathbf{x}_0) + \frac{1}{2}\mathbf{p}^T \nabla^2 f(\mathbf{x}_0)\mathbf{p} + O(\|\mathbf{p}\|^3). \tag{2.31}$$

Lemma 1:

When f is a convex function(al), any local minimizer $\mathbf{x}^*$ is a global minimizer of f.

Proof : We will prove the above lemma by contradiction. Suppose $\mathbf{x}^*$ be a local minimum point, but *not* a global minimizer. Then we can find a point, $\mathbf{z} \in \mathbb{R}^n$, such that $f(\mathbf{z}) < f(\mathbf{x}^*)$.

Now let us consider a line segment that connects $\mathbf{x}^*$ and $\mathbf{z}$,

$$\mathbf{x} = c_1\mathbf{z} + (1 - c_1)\mathbf{x}^*, \text{ for any } c_1 \in (0, 1].$$

Using the convexity property of the functional, f, we have,

$$f(\mathbf{x}) \leqslant c_1 f(\mathbf{z}) + (1 - c_1)f(\mathbf{x}^*),$$

but it has already been assumed that $f(\mathbf{z}) < f(\mathbf{x}^*)$ and $c_1 \in (0, 1]$. Therefore

$$f(\mathbf{x}) < f(\mathbf{x}^*),$$

which is not possible because $\mathbf{x}^*$ is already assumed to be the local minimizer, hence our supposition is incorrect, and hence, $\mathbf{x}^*$ is also the global minimum of the convex functional, f.

2.3.1 Basic components of large scale minimization algorithms

Following is a generic description of any implementation of an optimization algorithm. Starting at an initial guess point, $\mathbf{x}_0$, optimization algorithms generate a sequence of iterates $\{\mathbf{x}_i\}_{i=1}^{i=IterMax}$ which terminate when either no more progress can be made or when it seems that a solution point (either a local or global minimum) has been achieved with sufficient accuracy (usually it is prescribed by the user, and is called termination criteria). One iteration of the optimization algorithm, i.e., to move from $\mathbf{x}_i$ to $\mathbf{x}_{i+1}$, involves usage of information about the functional f at $\mathbf{x}_i$, also possibly the information accrued from previous iterates: $\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_{i-1}$. The goal being, as we move from one iteration to the next, the functional is lower than the functional value in the previous iteration, i.e., $f(\mathbf{x}_{i+1}) < f(\mathbf{x}_i)$. Following is an algorithmic representation of the above description,

Start with a specified initial guess of the solution $\mathbf{x}_0$.

For $i = 1, 2, \ldots, IterMax$,

If $\mathbf{x}_i$ is optimal (according to some specified termination criteria), stop.

Determine an improved estimate of the solution: $\mathbf{x}_{i+1} = \mathbf{x}_i + \alpha_i \mathbf{p}_i$, such that $f(\mathbf{x}_{i+1}) < f(\mathbf{x}_i)$.

Where $\mathbf{p}_i$ is a direction of descent, i.e., a direction following along which the value of the functional decreases and the scalar, α_i is the step length, i.e., the amount of *distance* we would like to go along the direction of descent to arrive at the new iterate, $\mathbf{x}_{i+1}$.

Hence optimization methods provide us the framework to minimize an objective functional to a prescribed satisfaction, in a tangible way. To achieve this goal, we need a recipe to proceed from iteration to another, i.e., find α_i and $\mathbf{p}_i$, check for termination criteria after every iteration. First we will address the issue of finding the step length, which also called *line search*. Because the optimization algorithm chooses a direction $\mathbf{p}_i$ and searches along this direction, from the current iterate $\mathbf{x}_i$ for a new iterate with a lower functional value. This process of finding α_i can be cast into another problem of solving the following 1-D minimization problem (moving along the direction of descent $\mathbf{p}_i$),

$$\min_{\alpha>0} f(\mathbf{x}_i + \alpha\, \mathbf{p}_i).$$

One can in principle solve the above equation exactly, but an exact minimization is expensive an practically infeasible. Therefore most of the line search algorithms generate a limited number of trial step lengths until it finds one that closely approximates the minimum of the above one-dimensional minimization problem. At the new iteration point $\mathbf{x}_{i+1}$, a new search direction and step length are computed (after checking for termination) and the process is repeated until convergence criteria are satisfied.

2.3.2 Brief description of a few nonlinear minimization algorithms

One of the simplest approaches to obtain the direction of descent is based upon using the gradient of the objective functional alone, such the steepest-descent method (because of its simplicity, steepest descent method converges very slowly). Where $\mathbf{p}_i = -\nabla f_i$, i.e., the direction of descent at the i-th. iteration is simply equal to the negative gradient of the objective functional at the i-th. iteration. Note that when $\mathbf{p}_i$ is a downhill direction, the angle θ_i between $\mathbf{p}_i$ and ∇f_i has a $\cos\theta_i < 0$, therefore

$$\mathbf{p}_i^T\, \nabla f_i = \|\mathbf{p}_i\|\, \|\nabla f_i\|\, \cos\theta_i \ < 0,$$

which implies that for all positively but sufficiently small ϵ, $f(\mathbf{x}_i + \epsilon \mathbf{p}_i) < f(\mathbf{x}_i)$, thus guaranteeing decrease in f. In-fact the above criteria, $\mathbf{p}_i^T\, \nabla f_i < 0$, is a necessary condition for any $\mathbf{p}_i$ to be a descent direction.

Let us consider the following second-order Taylor series approximation to $f(\mathbf{x}_i + \mathbf{p})$,

$$f(\mathbf{x}_i + \mathbf{p}) \approx f(\mathbf{x}_i) + \mathbf{p}^T\, \nabla f(\mathbf{x}_i) + \frac{1}{2}\mathbf{p}^T\, \nabla^2 f(\mathbf{x}_i)\, \mathbf{p},$$

we can write the right hand side of the above equation as a function of $\mathbf{p}$ alone,

$$K(\mathbf{p}) = f(\mathbf{x}_i) + \mathbf{p}^T \nabla f(\mathbf{x}_i) + \frac{1}{2}\mathbf{p}^T \nabla^2 f(\mathbf{x}_i)\, \mathbf{p}.$$

Newton' s method or simply, the Newton direction is obtained by requiring that the vector $\mathbf{p}$ minimize the functional $K(\mathbf{p})$, which is itself a second-order approximation of the original objective functional in the neighborhood of $\mathbf{x}_i$. In order to minimize $K(\mathbf{p})$, using the first order necessary conditions alone, the gradient of $\nabla_{\mathbf{p}} K(\mathbf{p}) = \nabla f(\mathbf{x}_i) + \nabla^2 f(\mathbf{x}_i)\, \mathbf{p} = 0$, which implies that

$$\mathbf{p} = -[\nabla^2 f(\mathbf{x}_i)]^{-1} \nabla f(\mathbf{x}_i). \tag{2.32}$$

It is easy to see that such a direction of descent does indeed guarantee descent (unless the gradient, $\nabla f(\mathbf{x}_i)$ is equal to zero). In the vicinity of the minimum, Newton' s method is characterized by a fast rate of convergence, typically quadratic. However, the biggest bottleneck involved in the practical implementation of Newton' s method is the need for the Hessian of the objective functional, or at-least the Hessian-vector product, whose computation is usually extremely expensive, if not practically impossible.

Quasi-Newton (QN) methods alleviate such a need for the Hessian information by constructing suitable approximations of the Hessian, using the gradient of the objective functional available at successive iterates. One of such methods is the Broyden, Fletcher, Goldfarb, Shanno (BFGS) method, which is described in Chapter 3. Another class of methods called Newton-conjugate gradient (or Newton-CG), also called truncated Newton (TN) methods are also based on the principal idea of Newton methods, equation (2.32). Just like the QN methods, the TN methods also approximately solve the Newton equations by solving it in the following way,

$$[\nabla^2 f(\mathbf{x}_i)]\, \mathbf{p} \approx -\nabla f(\mathbf{x}_i),$$

for $\mathbf{p}$. The Hessian vector product, i.e., the left hand side of the above equation is obtained by taking finite differences of two successive gradients, and the right hand side is just the current gradient itself and finally, using a conjugate gradient method the above system of linear equations is solved to obtain the search direction, $\mathbf{p}_i$.

Nonlinear conjugate gradient (CG) methods are another important class of minimization methods that provide a suitable direction of descent [48] at the i.-th iteration,

$$\mathbf{p}_i = -\nabla f(\mathbf{x}_i) + \gamma_i \mathbf{p}_{i-1}, \tag{2.33}$$

where γ_i is a scalar which ensures that the vectors, $\mathbf{p}_{i-1}$ and $\mathbf{p}_i$ are conjugate. The property of conjugacy of two vectors is defined as following. Any two non-zero vectors, $\mathbf{a}$ and $\mathbf{b}$ are conjugate with respect to a symmetric positive definite matrix $\mathbf{A}$, if

$$\mathbf{a}^T \mathbf{A}\, \mathbf{b} = 0, \forall\ \mathbf{a} \neq \mathbf{b}.$$

If we denote the gradient of the objective functional at the i.-th iteration, $-\nabla f(\mathbf{x}_i)$, by $\mathbf{r}_i$, then in order to ensure that the vector $\mathbf{p}_i$ given by equation (2.33) is conjugate to $\mathbf{p}_{i-1}$, we calculate γ_i using

$$\gamma_i = \frac{\mathbf{r}_i^T\, \mathbf{r}_i}{\mathbf{r}_{i-1}^T\, \mathbf{r}_{i-1}}.$$

This form of nonlinear CG method is known as the Fletcher-Reeves [49] CG method. In another popular implementation, known as the Polak-Ribiére CG method [50], γ_i is obtained as,

$$\gamma_i = \frac{\mathbf{r}_i^T\, (\mathbf{r}_i - \mathbf{r}_{i-1})}{\|\mathbf{r}_{i-1}\|_2^2}.$$

In the 4D-Var minimization applications we will consider in the following chapters, we will be using a version of the BFGS quasi-Newton method called, the limited memory version of the BFGS method. As detailed earlier, the adjoint method is used to compute the gradient of the cost functional, which is in-turn supplied to the minimization algorithm to minimize the 4D-Var objective functional.

CHAPTER 3

IMPACT OF HIGH RESOLUTION ADVECTION SCHEMES ON VDA

The numerical weather prediction (NWP) models are based on a set partial differential equations (PDEs). Due to the complex nature of these PDEs, arising from nonlinearities, boundary conditions, etc, NWP models are discretized (in space and time) and solved using high speed digital computers. Spatial discretization methods for solving these PDEs can be broadly classified as finite difference (FD) [51, 52], finite volume (FV) [53], finite element (FE) [54, 55, 56] and spectral methods [57] (including the discontinuous Galerkin (DG) methods [58]). 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). 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 the Euler equations in gas dynamics [59] 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, 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 [53, 60, 61] for details of these methods), to name a few. In the FD context, please refer [52], FE [56], spectral and DG methods [57] and [58] respectively for details.

In atmospheric 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 [61] 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 [62]. 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 [63] provided a detailed analysis and comparison of various advection schemes for a simple linear atmospheric transport model. Lin et. al [64] have analysed the effect of varying the slope limiters using an atmospheric general circulation model. Lin and Rood [65] have compared the first order upwind, central difference, PPM (modified monotonic and positive definite) and monotonic van Leer schemes [66]. Towards the development of a fully operational atmospheric general circulation model based on FV discretization [67], Lin and Rood [68] 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 [69, 70] and QUICK of Leonard [71, 72, 73], see [74]. MPDATA is also used in the hybrid coordinate ocean model (HYCOM) [75] 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 [76]. Iskandarani et al. [77] 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.

The impact of different discretization techniques for the advection term(s) in the framework of inverse problems and problems related to `DA` has not been extensively tested, except for work by Vukićević et al. [74] and Thuburn and Haine [78]. In [74] 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 [78] 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.

In this chapter we study 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. The solution of these nonlinear `PDEs` has been achieved using FV discretization. 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. Finally the performance of the various slope limited and PPM schemes for the minimization of a certain cost functional is presented in the context of `DA` (in both 1-D and 2-D).

3.1 Description of the mathematical and numerical models

The Burgers equation [79] 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)

$$\frac{\partial \phi}{\partial t} + \frac{\partial f}{\partial x} = \frac{\partial S}{\partial x}, \tag{3.1}$$

where the f is a convex flux function given by $\frac{\phi^2}{2}$ and S represents the source term(s). Equation (3.1) is the well known Burgers equation which is a very important fluid dynamical model useful for conceptual understanding of nonlinear waves, shock formation [59, 80] and turbulence [81]. Various numerical schemes (see Fletcher [82] 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 [53] to solve the above equation, all differing in the way which we reconstruct the solution, ϕ, in each cell using different slope limiters. We will closely follow the approach taken by Monotone Upstream-centered Schemes for Conservation Laws (MUSCL), see [83, 84, 85, 66, 86].

Let us start by writing the integral form of (3.1) within the i-th. cell, $\mathcal{C}_i$,

$$\frac{\partial}{\partial t}\int_{\mathcal{C}_i}\phi(x,t)dx = f[\phi(x_{i-\frac{1}{2}},t)] - f[\phi(x_{i+\frac{1}{2}},t)] + S(x_{i+\frac{1}{2}},t) - S(x_{i-\frac{1}{2}},t) \tag{3.2}$$

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

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

$$\Phi_i{}^n \approx \frac{1}{\Delta x_i}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}\phi(x,t_n)\,dx, \tag{3.3}$$

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

Integrating equation (3.2) from t_n to t_{n+1} yields,

$$\begin{aligned}\int_{\mathcal{C}_i}\phi(x,t_{n+1})dx - \int_{\mathcal{C}_i}\phi(x,t_n)dx &= \int_{t_n}^{t_{n+1}} f[\phi(x_{i-\frac{1}{2}},t)]dt - \int_{t_n}^{t_{n+1}} f[\phi(x_{i+\frac{1}{2}},t)]dt \\ &\quad + \int_{t_n}^{t_{n+1}} [S(x_{i+\frac{1}{2}},t) - S(x_{i-\frac{1}{2}},t)]dt,\end{aligned}$$

dividing by Δx_i and rearranging,

$$\begin{aligned}\frac{1}{\Delta x_i}\int_{\mathcal{C}_i}\phi(x,t_{n+1})dx &= \frac{1}{\Delta x_i}\int_{\mathcal{C}_i}\phi(x,t_n)dx \\ &\quad -\frac{1}{\Delta x_i}\int_{t_n}^{t_{n+1}}\{f[\phi(x_{i+\frac{1}{2}},t)] - f[\phi(x_{i-\frac{1}{2}},t)]\}dt + \\ &\quad +\frac{1}{\Delta x_i}\int_{t_n}^{t_{n+1}}[S(x_{i+\frac{1}{2}},t) - S(x_{i-\frac{1}{2}},t)]dt.\end{aligned}$$

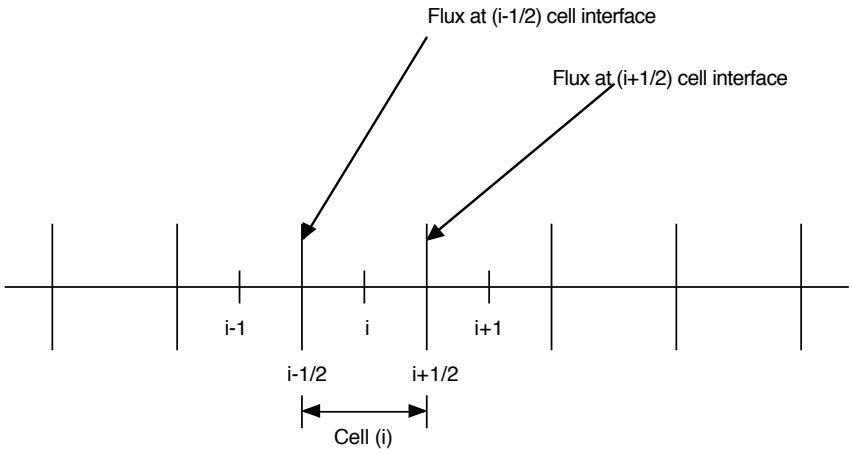

Figure 3.1: Finite volume discretization

Assuming a viscous dissipative source $S = \nu\phi_x$ (ν is the kinematic viscosity) and using equation (3.3) we obtain,

$$\begin{aligned} {\Phi_i}^{n+1} &= {\Phi_i}^n - \frac{\Delta t}{\Delta x_i}[(Flux)^n_{i+\frac{1}{2}} - (Flux)^n_{i-\frac{1}{2}}] + \\ & \nu\frac{1}{\Delta x_i}\int_{t_n}^{t_{n+1}}[\phi_x(x_{i+\frac{1}{2}},t) - \phi_x(x_{i-\frac{1}{2}},t)]dt, \end{aligned} \tag{3.4}$$

where $(Flux)^n_{i+\frac{1}{2}} \approx \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} f[\phi(x_{i+\frac{1}{2}},t)]dt$ is some approximation of the average flux (described later in this section) along the cell interface at $x_{i+\frac{1}{2}}$, see figure (3.1) for an illustration of the grid cells.

3.1.1 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, $\mathcal{C}_i$,

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

where Φ_i is given by equation (3.3), x_i is the coordinate of the i-th. cell center 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 [60] and the simple Burgers equation (3.1) support solutions that have discontinuities (or, shocks), expansion fans, contact discontinuities. Apart from ensuring satisfaction of the CFL (Courant-Friedrichs-Lewy) condition [51], unless special treatment is taken, the numerical solutions will lead to excessive dissipation, incorrect phase speeds, spurious oscillations; see Laney [60] 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 [87, 88, 53] property is by limiting the values of the slopes ($\Delta\Phi_i$). Lin et al. [64] 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 (from now onwards we will assume an uniform grid, i.e, $\Delta x_i = \Delta x \, \forall \, i$).

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

$$\Delta\Phi_i^n \equiv 0, \forall \, i. \tag{3.5}$$

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

$$[\Delta\Phi_i^n]_{avg} = \frac{1}{\Delta x} \frac{\delta\Phi^n_{i-\frac{1}{2}} + \delta\Phi^n_{i+\frac{1}{2}}}{2}, \tag{3.6}$$

   where $\delta\Phi^n_{i+\frac{1}{2}} = \Phi^n_{i+1} - \Phi^n_i$ 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):

$$[\Delta\Phi_i^n] = \frac{1}{\Delta x} \, SIGN([\Delta\Phi_i^n]_{avg}) \cdot MIN[|[\Delta\Phi_i^n]_{avg}|, 2DIM(\Phi_i^n, \Phi^{min})], \tag{3.7}$$

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

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

4. *Limiter 4* (monotonicity preserving scheme):

   Another form of slope limiter which ensures monotonicity, suggested by van Leer [66, 89] is the following,

$$[\Delta\Phi_i^n] = \begin{cases} \frac{1}{\Delta x}[\delta\Phi^n_{i-\frac{1}{2}} \cdot \delta\Phi^n_{i+\frac{1}{2}}]/[\Delta\Phi_i^n]_{avg}, \text{if } SIGN(\delta\Phi^n_{i-\frac{1}{2}}) = SIGN(\delta\Phi^n_{i+\frac{1}{2}}), \\ 0, \text{ otherwise.} \end{cases} \tag{3.8}$$

5. *Limiter 5* (constrained van Leer scheme):

   We can determine locally defined minimum and maximum values of the solution as,

$$\begin{aligned} \Phi_i^{min} &= MIN[\Phi^n_{i-1}, \Phi^n_i, \Phi^n_{i+1}] \\ \Phi_i^{max} &= MAX[\Phi^n_{i-1}, \Phi^n_i, \Phi^n_{i+1}] \end{aligned} \tag{3.9}$$

   and use them to limit the value of the slope as following [66, 89] (also ensures monotonicty),

$$[\Delta\Phi_i^n] = \frac{1}{\Delta x} SIGN([\Delta\Phi_i^n]_{avg}) \cdot MIN[|[\Delta\Phi_i^n]_{avg}|, 2DIM(\Phi_i^n, \Phi_i^{min}), 2DIM(\Phi_i^{max}, \Phi_i^n)]. \tag{3.10}$$

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 Φ_{global}^{min} and Φ_{global}^{max} respectively, and replacing these in above limiter formulation, we obtain:

$$[\Delta\Phi_i^n] = \frac{1}{\Delta x} SIGN([\Delta\Phi_i^n]_{avg}) \cdot MIN[|[\Delta\Phi_i^n]_{avg}|, 2DIM(\Phi_i^n, \Phi_{global}^{min}), 2DIM(\Phi_{global}^{max}, \Phi_i^n)]. \tag{3.11}$$

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

3.1.2 ENO flux

To calculate the flux at the right cell face $x_{i+\frac{1}{2}}$, we used the ENO [90, 91, 92, 60] flux formulation. Using the i-th. and $i+1$ cell reconstructed values evaluated at $x_{i+\frac{1}{2}}$ (see Laney [60] chapter 23 for details), we obtain

$$(Flux)^n_{i+\frac{1}{2}} = f^G[\{\Phi^n_i + \frac{\Delta\Phi^n_i \Delta x}{2}(1 - \frac{\Delta t}{\Delta x}\Phi^n_i)\}, \{\Phi^n_{i+1} - \frac{\Delta\Phi^n_{i+1}\Delta x}{2}(1 + \frac{\Delta t}{\Delta x}\Phi^n_{i+1})\}], \quad (3.12)$$

where

$$f^G[\Phi^n_i, \Phi^n_{i+1}] = \begin{cases} MIN[f(\Phi^n_i), f(\Phi^n_{i+1}), f(\Phi_*)] \text{if } \Phi^n_i \leqslant \Phi^n_{i+1}, \\ MAX[f(\Phi^n_i), f(\Phi^n_{i+1}), f(\Phi_*)] \text{if } \Phi^n_i > \Phi^n_{i+1}. \end{cases} \quad (3.13)$$

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

Remark: If the slope in each cell is equal to zero, as in the equation (3.5), then the above ENO flux form reduces to Godunov flux form [93].

Instead of using a piecewise linear reconstruction within each cell, we can as well apply the piecewise parabolic reconstruction approach of Colella and Woodward [94, 95, 96] within each cell.

3.1.3 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.

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

$\Phi_{L,i}$ and $\Phi_{R,i}$ are approximations of the state at the left and right cell interface, as in MUSCL piecewise linear extrapolation, $\Delta\Phi_i = \Phi_{R,i} - \Phi_{L,i}$ and $\Phi_{6,i} = 6(\Phi_i - \frac{1}{2}(\Phi_{L,i} + \Phi_{R,i}))$ for details of the above reconstruction procedure, see [94].

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

3.1.4 Integration in time using a second order optimal TVD R-K method

Using equations (3.12) and (3.13) 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},

$$\begin{aligned} {\Phi_i}^{n+1} &= {\Phi_i}^{n} - \frac{\Delta t}{\Delta x}[(Flux)^n_{i+\frac{1}{2}} - (Flux)^n_{i-\frac{1}{2}}] + \\ & \nu\frac{\Delta t}{\Delta x^2}[\Phi^n_{i+1} - 2\Phi^n_i + \Phi^n_{i-1}]. \end{aligned} \tag{3.14}$$

The above numerical scheme is at-least second order accurate (MUSCL schemes: (3.6)-(3.9) 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 [97, 98], given by Gottlieb and Shu [99]. Following their notation, let

$$L(\Phi^n_i) = -\frac{1}{\Delta x}\left[(Flux)^n_{i+\frac{1}{2}} - (Flux)^n_{i-\frac{1}{2}}\right] + \nu\frac{1}{\Delta x^2}[\Phi^n_{i+1} - 2\Phi^n_i + \Phi^n_{i-1}],$$

then the following sequence of two steps gives us,

$$\Phi^{(1)}_i = \Phi^n_i + \Delta t\, L(\Phi^n_i),$$

$${\Phi_i}^{n+1} = \frac{1}{2}\Phi^n_i + \frac{1}{2}\Phi^{(1)}_i + \frac{1}{2}\Delta t\, L(\Phi^{(1)}_i).$$

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).

3.1.5 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., [100] and [67] for details. The SW equations in spherical coordinates in the vorticity divergence form can be written as,

$$\frac{\partial h}{\partial t} + \nabla \cdot (\mathbf{V}h) = 0 \tag{3.15}$$

$$\frac{\partial u}{\partial t} = \Omega v - \frac{1}{a\ cos\theta}\frac{\partial}{\partial \lambda}[\kappa + \varphi] \tag{3.16}$$

$$\frac{\partial v}{\partial t} = -\Omega u - \frac{1}{a}\frac{\partial}{\partial \theta}[\kappa + \varphi] \tag{3.17}$$

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 geopotential is given by

$$\varphi = \varphi_s + g\,h,$$

$\varphi_s = gh_s$, $\kappa = \frac{1}{2}\mathbf{V} \cdot \mathbf{V}$ 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 [101] and [102].

The finite volume shallow water equations model of Lin and Rood [68] 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 [65]. 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 centered 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 [103] 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 $\frac{1}{2h}\Omega^2$) is 1%. More details are available in Lin and Rood, 1997 [68]. This represents excellent conservation of integral invariants of the shallow water equations.

3.2 Test cases and results

3.2.1 Case 1: 1-D viscous Burgers equation

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

$$\frac{\partial\phi}{\partial t}+\frac{\partial}{\partial x}(\frac{\phi^2}{2})=\nu(\frac{\partial^2\phi}{\partial x^2}), \tag{3.18}$$

for $x\in(-\pi,\pi)$ and $t>0$,

with boundary conditions, $\phi(x=\pm\pi,t)=0$.

Benton and Platzman [104] provide an exact solution for the above Burgers equation (3.18), with initial condition given by,

$$\phi(x,0)=\phi(x,t=0)=-R\sin(x), \tag{3.19}$$

where R is the Reynolds number. It is related to the viscosity via the relationship, $R=\frac{UL}{\nu}$, 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:

$$\phi_{exact}(x,t)=\frac{4\sum_{n=1}^{\infty}na_ne^{-n^2t}\sin(nx)}{a_0+2\sum_{n=1}^{\infty}a_ne^{-n^2t}\cos(nx)}, \tag{3.20}$$

where $a_n=(-1)^nI_n(\frac{1}{2}R)$, 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 (3.2) (which has been generated by setting $R=1$).

In figures (3.3)- (3.4) we have plotted the exact solution along with solutions obtained using the various numerical schemes (MUSCL schemes, with limiters 1- 6, and the PPM) at $t=1$. Since the Bessel functions of second kind are exponentially decreasing functions, to compute ϕ_{exact} we have used $n=10$ for the summation in equation (3.20). The numerical solutions have been computed using a resolution of $N_x=40$ grid cells ($\Delta x=\frac{2\pi}{N_x}$) and a time

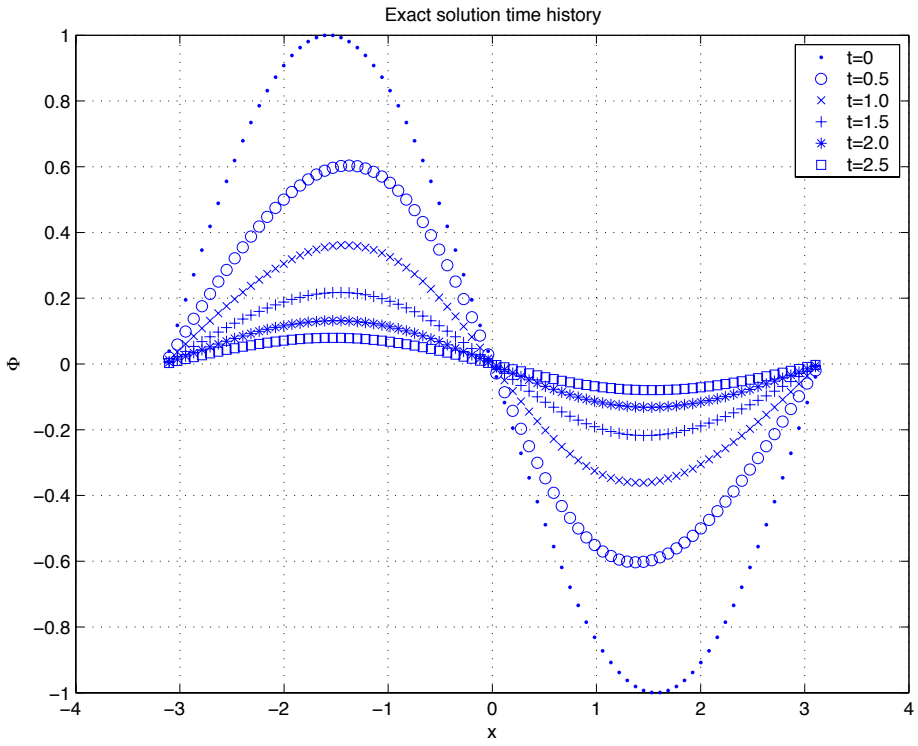

Figure 3.2: Exact solution at $t = 0, 0.5, 1.0, 1.5, 2.0, 2.5$

step (Δt) given by the CFL criteria, $CFL = \frac{U\Delta t}{\Delta x}$, where the CFL number was assigned a value of 0.01.

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 $[\Delta\Phi_i^n]_{avg}$), 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 (3.1) 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 [64] 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.

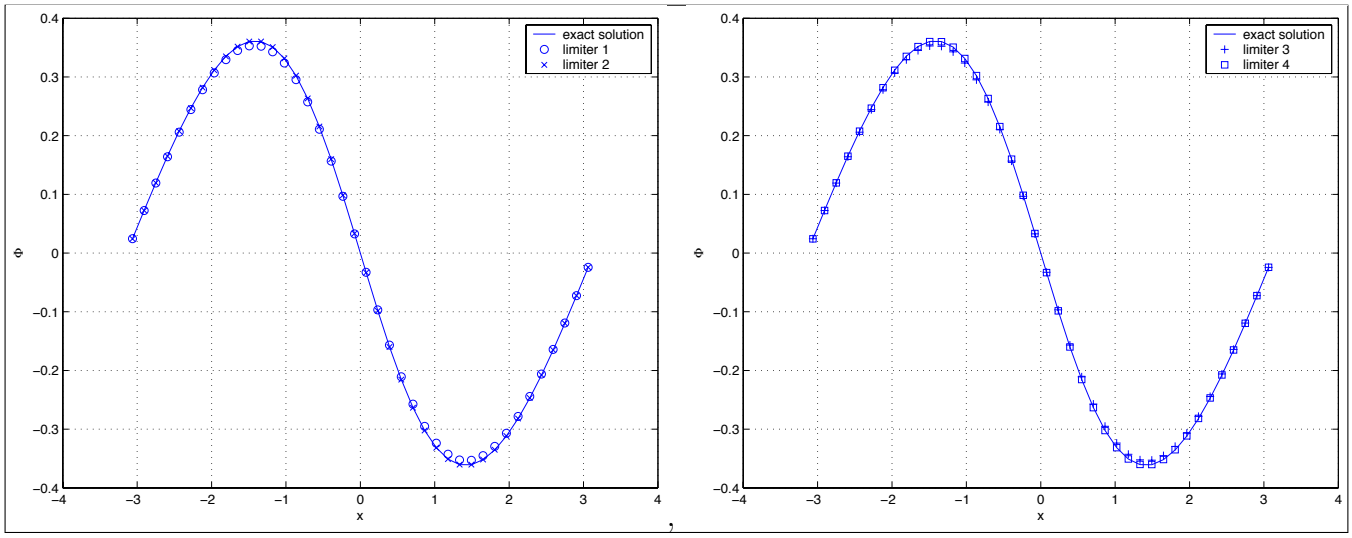

Figure 3.3: Exact and numerical solutions (in forward mode) of the 1-D nonlinear viscous Burgers equation with slope limiters 1, 2, 3 and 4 at $t = 1$.

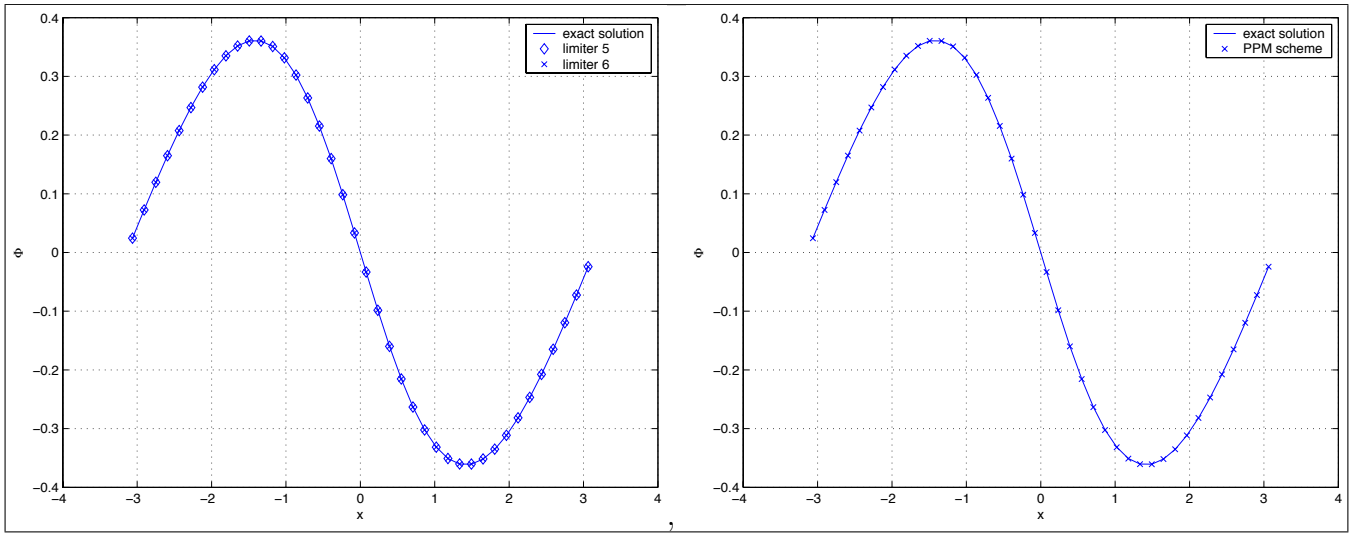

Figure 3.4: Same as in figure (3.3), but with limiters 5, 6 and PPM scheme.

Table 3.1: 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

3.2.2 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,

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

With the following initial condition (on the whole real line) [105],

$$\phi(x,0) = \begin{cases} 0, & x < -1 \\ \frac{1}{2}, & -1 < x < 0 \\ 0, & x > 0. \end{cases} \tag{3.21}$$

The solution develops into a shock and an expansion fan (for details of the solution, see [105]), analytically given by (for $t \leq 4$, i.e., before the expansion fan meets the shock),

$$\phi(x,t) = \begin{cases} 0, & x < -1 \\ \frac{x+1}{t}, & -1 < x < \frac{t}{2} - 1 \\ \frac{1}{2}, & \frac{t}{2} - 1 < x < \frac{t}{4} \\ 0, & x > \frac{t}{4}. \end{cases} \tag{3.22}$$

In figure (3.5) we have plotted the exact along with the numerical solutions obtained using the various numerical schemes (MUSCL schemes, with limiters 1- 6, and the PPM) at $t = 2$. The numerical solutions have been computed using a resolution of $N_x = 80$ grid cells

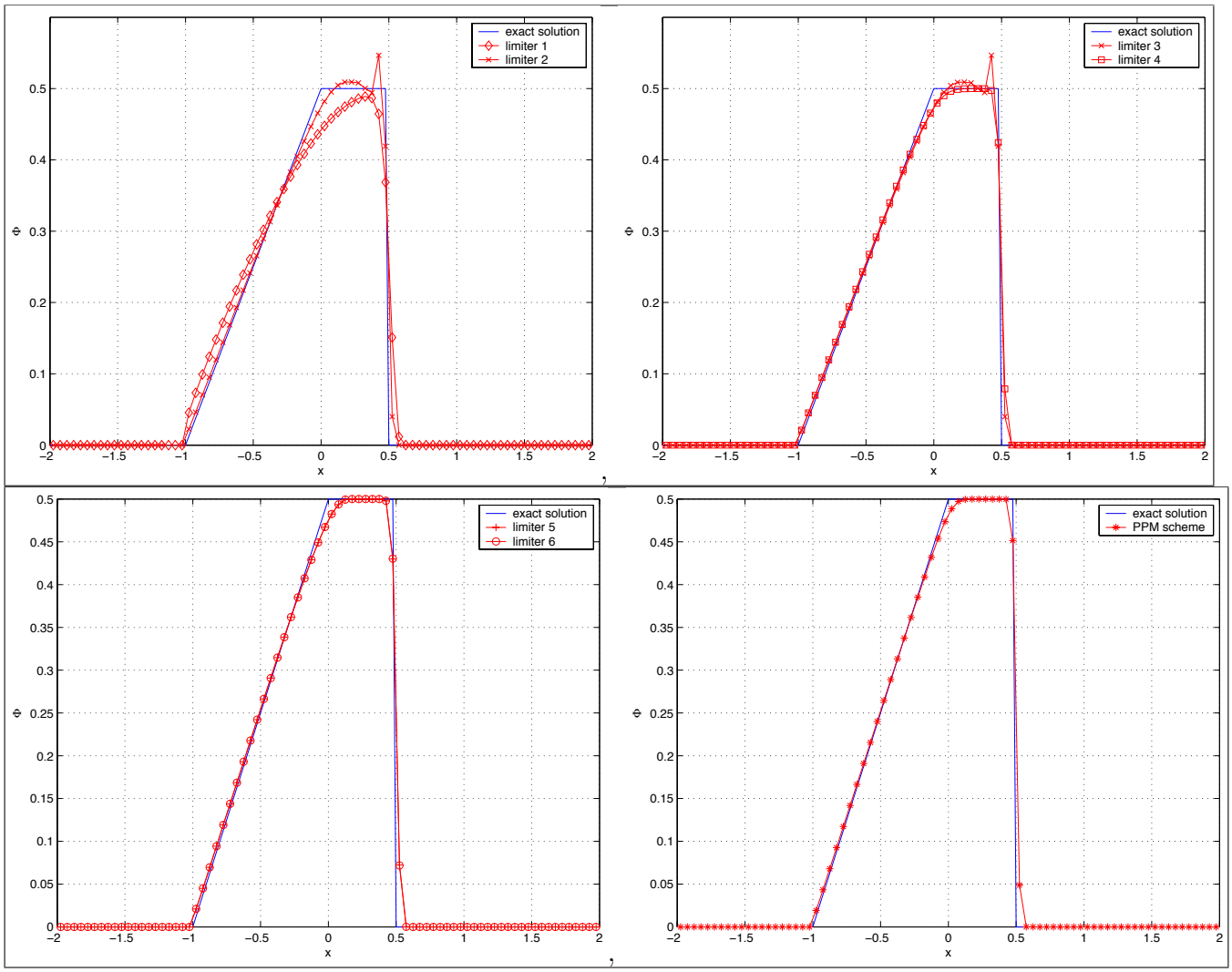

Figure 3.5: Exact and numerical solution (in forward mode) of the 1-D nonlinear inviscid Burgers equation with slope limiters 1, 2, 3, 4, 5, 6 and PPM scheme at $t = 2$

.

($\Delta x = \frac{4}{N_x}$) and a time step (Δt) given by the CFL criteria, $CFL = \frac{U\Delta t}{\Delta x}$, where the CFL number was assigned a value of 0.1 and $U = 0.5$.

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 [65], 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 [64] due to the fact that limiter 5 provides less implicit diffusion that 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 [67] (also see the documentation of the *Community Atmosphere Model 3.0* [106]) hence we will restrict our 2-D study to only these schemes.

3.2.3 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. [101]. 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 [107]. 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 modeling community for a number of reasons. Haurwitz [108] 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 (3.6)- (3.13) provide results obtained by integration of the finite volume SW equations model of Lin and Rood [68] 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 model is the same as in [68], 128 grid cells along the longitude and 64 along the latitude, and a time step of 600 seconds. 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 (3.6) and (3.7). 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.

3.3 Programming the adjoint model

In this section the details of programming adjoint model, which is useful to obtain the gradient of the cost functional with respect to the control parameters efficiently, is described. The following form of the cost functional is considered,

$$\mathcal{J}(\mathbf{x}) = \frac{1}{2}\sum_{k=0}^{n}(\mathbf{x}(t_k) - \mathbf{x}^{obs}(t_k))^T\,\mathbf{W}(t_k)\,(\mathbf{x}(t_k) - \mathbf{x}^{obs}(t_k)), \tag{3.23}$$

where $t \in [t_0, t_n]$ is the (data) assimilation time window comprised of n time steps, $\mathbf{W}(t_k)$ is a diagonal weighting matrix, $\mathbf{x}(t_k)$ is the evolving state vector and $\mathbf{x}^{obs}(t_k)$ 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 the next section. The directional derivative of the above cost functional, in the direction of $\delta\mathbf{x}$ is given by $(\nabla_{\mathbf{x}}\mathcal{J})^T\delta\mathbf{x}$. From equation (3.23),

$$\delta\mathcal{J}(\mathbf{x}) = \sum_{k=0}^{n}(\mathbf{W}(t_k)\,(\mathbf{x}(t_k) - \mathbf{x}^{obs}(t_k)))^T\delta\mathbf{x}(t_k), \tag{3.24}$$

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,

$$(\nabla_{\mathbf{x}}\mathcal{J})^T\delta\mathbf{x} = \sum_{k=0}^{n}(\mathbf{W}(t_k)\,(\mathbf{x}(t_k) - \mathbf{x}^{obs}(t_k)))^T\delta\mathbf{x}(t_k). \tag{3.25}$$

We follow the approach of first discretize and then differentiate (see [10] and [14] for details). Discrete numerical operations in the nonlinear forward model are having unique corresponding operations in the adjoint model. The tangent linear model (TLM), which was first derived and described in the previous chapter, is derived, to start with. The

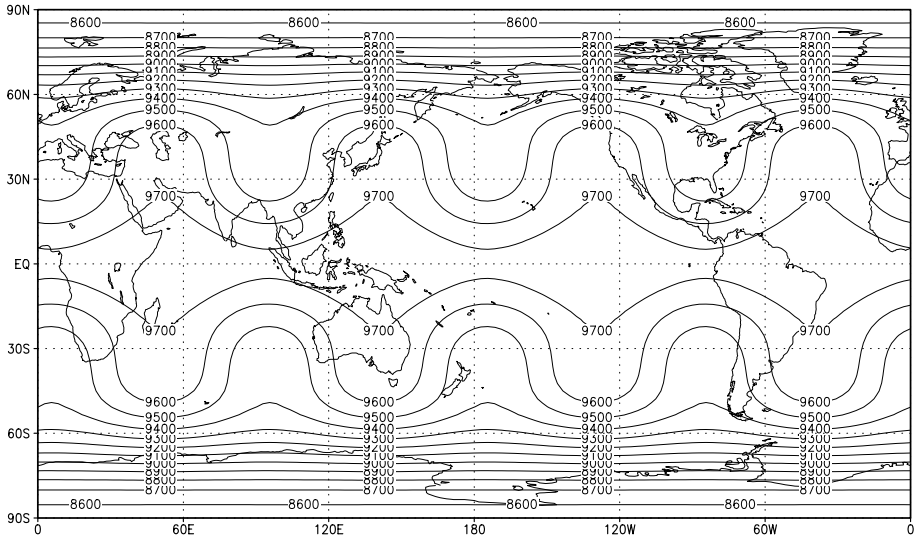

Figure 3.6: Height field isolines at Day-14 using the first order advection scheme (lim.1), for the Rossby-Haurwitz wavenumber 4 case using a finite volume global SW equations model. Contour interval is 100 m.

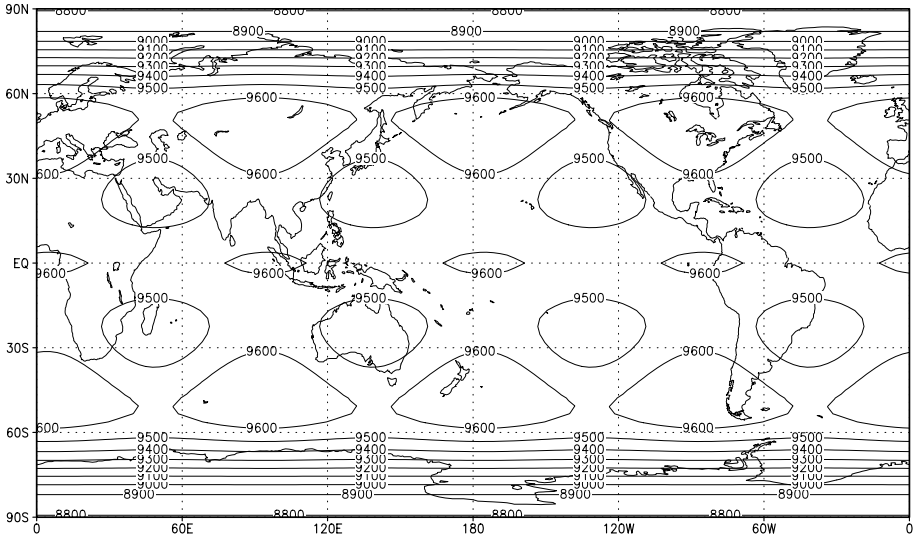

Figure 3.7: Same as above, but at Day-30

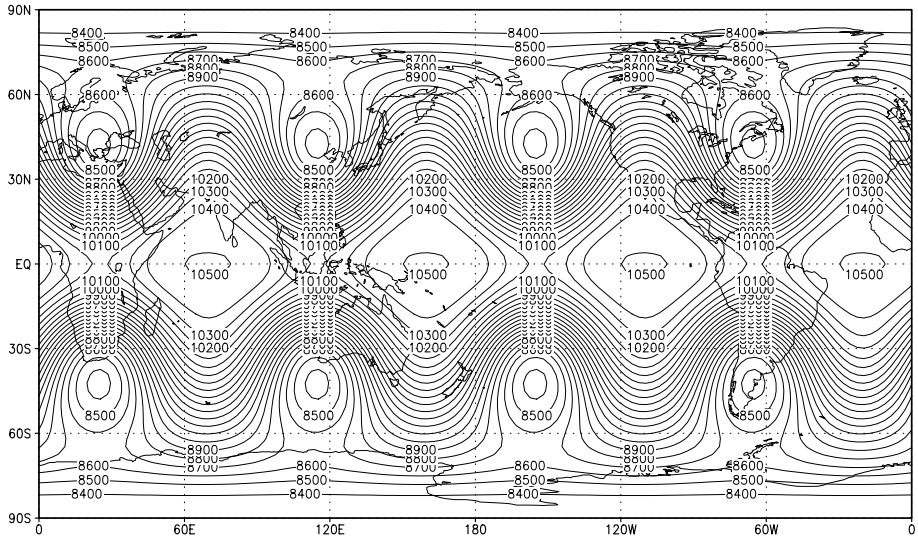

Figure 3.8: Same as in figure (3.6), but using unconstrained van Leer scheme (lim.2).

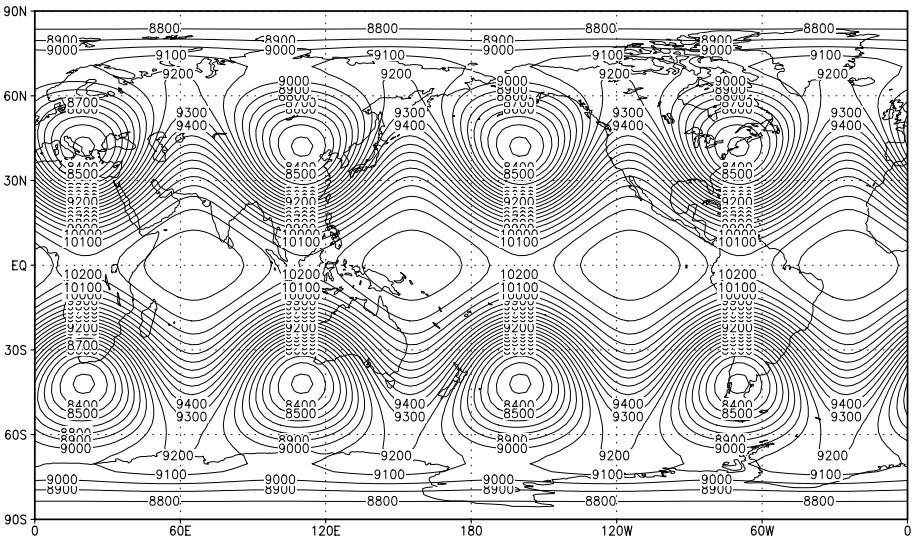

Figure 3.9: Same as above, but at Day-30

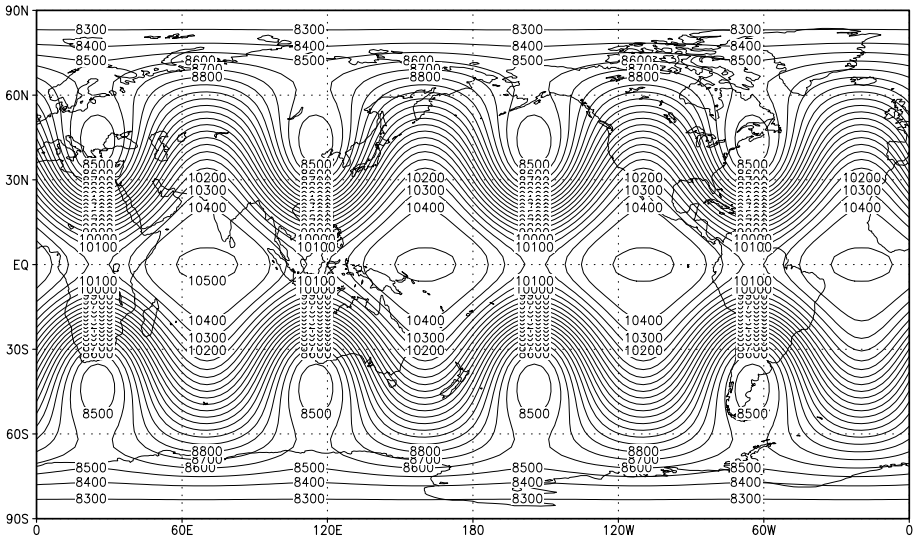

Figure 3.10: Same as in figure (3.6), but using constrained van Leer scheme (lim.5).

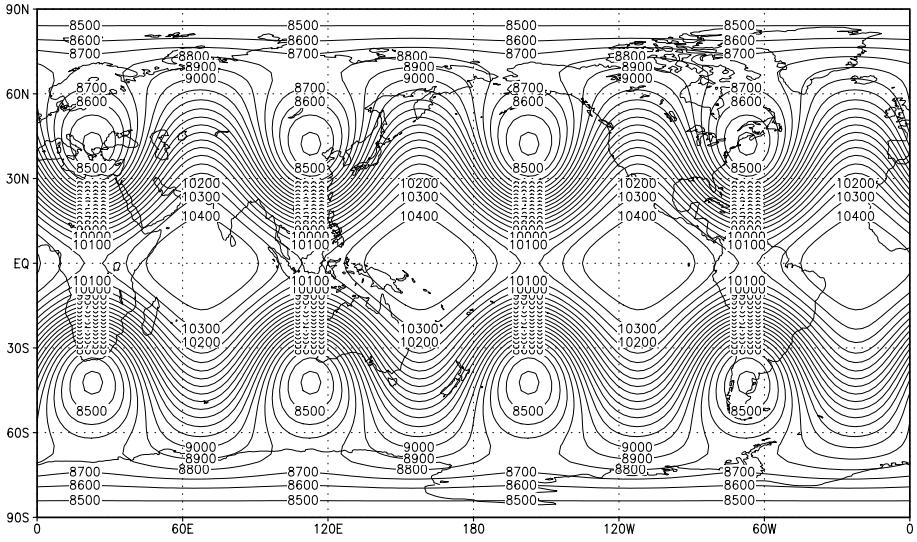

Figure 3.11: Same as above, but at Day-30

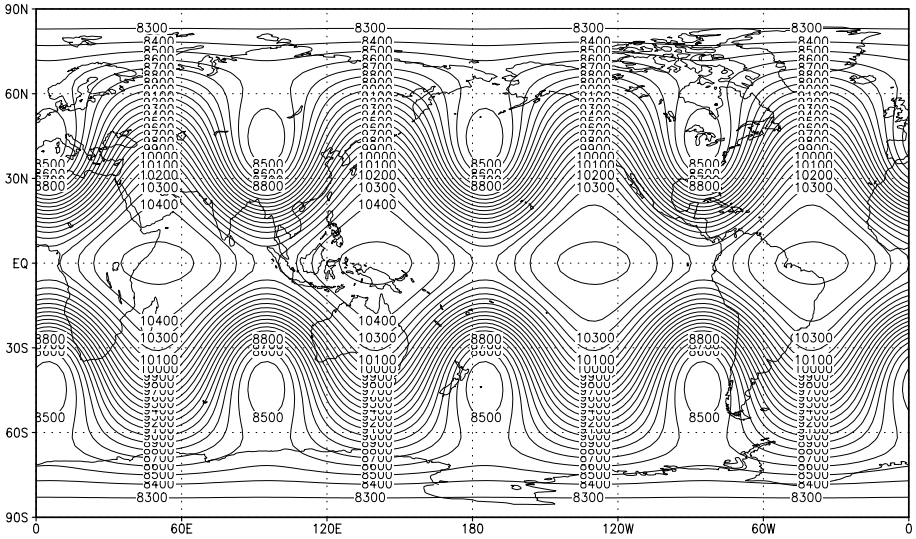

Figure 3.12: Same as in figure (3.6), but using PPM advection scheme.

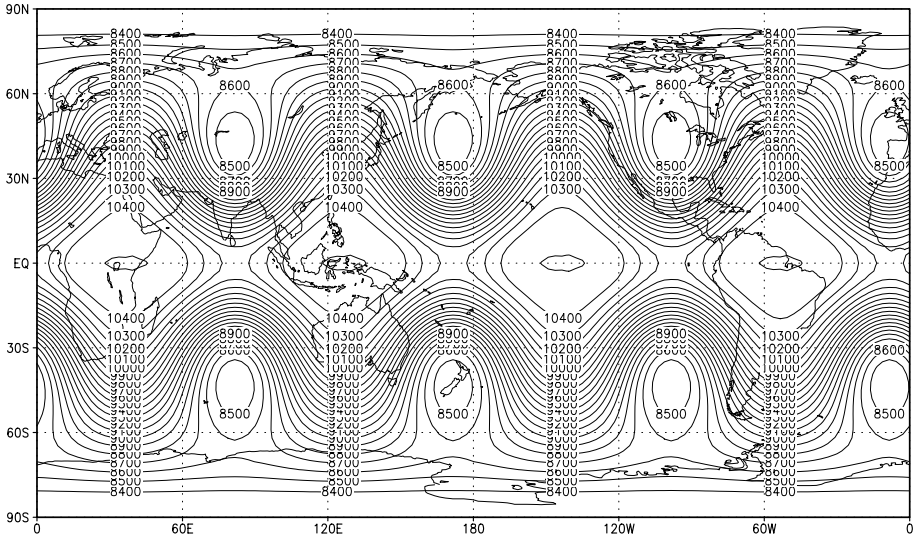

Figure 3.13: Same as above, but at Day-30

TLM code is programmed by linearizing line by line, the nonlinear forward model code. Recall that 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, $L_1^T, L_2^T, \ldots, L_k^T$. 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 [109] and [1] for details), and to verify the same for the transposition property (all our subroutines satisfactorily passed this test). We have used TAMC [43, 110, 111] (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 B, 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, $\nabla_{\mathbf{x}}\mathcal{J}$ 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* $W(t_k)\left(\mathbf{x}(t_k) - \mathbf{x}^{obs}(t_k)\right)$ 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, upto first order,

$$\mathcal{J}(\mathbf{x} + \eta\nabla\mathcal{J}) = \mathcal{J}(\mathbf{x}) + \eta(\nabla\mathcal{J})^T\nabla\mathcal{J} + O(\eta^2), \tag{3.26}$$

where η is a scalar and the gradient, $\nabla\mathcal{J} = \nabla_{\mathbf{x}}\mathcal{J}$, is obtained by using the adjoint model. We can rewrite the above equation as in [10],

$$\Psi(\eta) = \frac{\mathcal{J}(\mathbf{x} + \eta\nabla\mathcal{J}) - \mathcal{J}(\mathbf{x})}{\eta\nabla\mathcal{J}^T\nabla\mathcal{J}} = 1 + O(\eta). \tag{3.27}$$

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 (3.2) and (3.3) 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 [112] for details of the adjoint model for the SW equations model used.

Table 3.2: 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.

	$\Psi(\eta)$						
$-log_{10}(\eta)$	lim.1	lim.2	lim.3	lim.4	lim.5	lim.6	PPM
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

3.4 Minimization

We used an unconstrained limited memory quasi-Newton (L-BFGS) minimization algorithm [113, 114] (availabe for download from www.netlib.org/opt/lbfgs_um.shar) for minimization of the cost functional $\mathcal{J} = \mathcal{J}(\mathbf{x}_k)$, where $\mathbf{x}_k$ is the n component (control) vector at the k^{th} iteration. $\mathbf{g}_k = \mathbf{g}(\mathbf{x}_k) = \nabla \mathcal{J}_k$ is the gradient vector of size n, and $\mathbf{H}_k = \nabla^2 \mathcal{J}_k$ is the $n \times n$ symmetric Hessian matrix of the second partial derivatives of $\mathcal{J}$ with respect to the control vector. The new iterate is given by,

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \, \mathbf{p}_k, \tag{3.28}$$

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

Table 3.3: 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.

	$\Psi(\eta)$			
$-log_{10}(\eta)$	lim.1	lim.2	lim.5	PPM
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

Iterations are terminated when (using the L_2 norm)

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

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

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

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

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 $\mathbf{B}_{k+1}$ (i.e, $\mathbf{H}_{k+1}^{-1}$) is given by,

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \frac{(1 + \mathbf{q}_k^T \mathbf{B}_k \mathbf{q}_k)}{\mathbf{q}_k^T \mathbf{p}_k} \frac{\mathbf{p}_k \mathbf{p}_k^T}{\mathbf{p}_k^T \mathbf{q}_k} - \frac{\mathbf{p}_k \mathbf{q}_k^T \mathbf{B}_k + \mathbf{B}_k \mathbf{q}_k \mathbf{p}_k^T}{\mathbf{q}_k^T \mathbf{p}_k}, \tag{3.29}$$

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 (3.28) 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 [113, 114], [46] and [115, 116] for applications.

3.5 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. [74], 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 (3.23), namely

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

with respect to the initial state $\mathbf{x}(t_0) \equiv \phi(x, 0)$ as the control parameter and we have prescribed $\mathbf{W}(t_k) \equiv I$, 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` of real observations.

In our 1-D twin experiments, we used the initial condition given in (3.19), run the forward model up-to time step t_k to obtain the observations, $\mathbf{x}^{obs}(t_k)$ and in the 2-D case, the Rossby wave has been used as the initial condition. The initial condition is then randomly perturbed,

$$\mathbf{x}^{pert}(t_0) = \mathbf{x}(t_0) + \epsilon \cdot RAND \cdot \mathbf{x}(t_0), \tag{3.30}$$

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

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, $\mathbf{x}$ (from now onwards denoted by $\mathbf{x}^{recovered}$), 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 hours. 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 (3.14) - (3.17) show the variation of the cost functional, $\mathcal{J}$, and gradient norm (in L_2 norm) versus the number of iterations and in table (3.4) we compare the values of $\mathbf{x}^{recovered}$ 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 (3.18)- (3.19) and table (3.5) 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 the previous section, 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 [46]). 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 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 (3.7), (3.9) and (3.11)). Programming these switches in the adjoint model proves to be a very tedious and time consuming task.

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 $\mathbf{x}^{pert}$ and $\mathbf{x}^{recovered}$ to ($T = 2.2$ seconds in 1-D case and $T = 7$ hours in the 2-D case) is provided in tables (3.4) and (3.5) respectively. The forecast errors are reduced for all the schemes, in both 1-D and 2-D, when the $\mathbf{x}^{recovered}$ is used as the optimal initial condition. As evident, in the 1-D case, though the first order scheme (limiter 1) yields the closest $\mathbf{x}^{recovered}$ (to $\mathbf{x}(t_0)$), the forecast obtained by using the $\mathbf{x}^{recovered}$ 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.

Table 3.4: Comparison of the $\phi^{recovered}$ for different advection schemes based on 1-D data assimilation experiments, $\|\phi^{pert}(x,0)-\phi(x,0)\|_2 = 1.3004 \times 10^{-2}$ for all the schemes; forecast time, $T = 2.2$ seconds.

Advection Scheme	$\|\phi^{recovered} - \phi(x,0)\|_2$	$\|\phi^{pert}(x,T) - \phi(x,T)\|_2$	$\|\phi^{recovered}(T) - \phi(x,T)\|_2$
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}

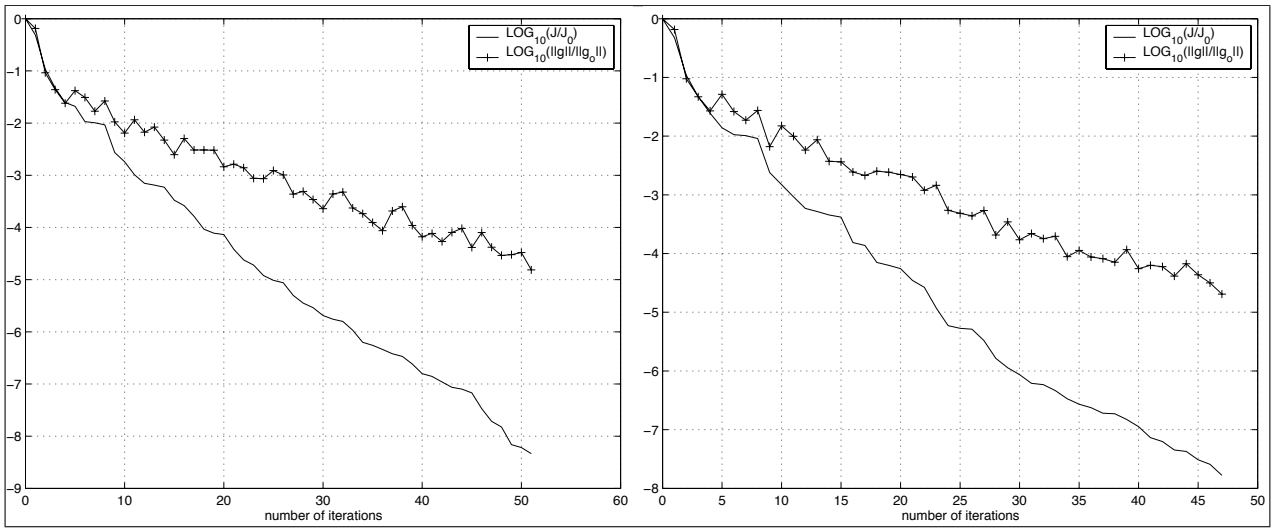

Figure 3.14: Variations of the normalized cost function $\frac{\mathcal{J}}{\mathcal{J}_0}$ and normalized gradient $\frac{\|\mathbf{g}\|}{\|\mathbf{g}_0\|}$ versus the number of minimization iterations using slope limiters 1 and 2 in forward and adjoint models for the 1-D Burgers equation model (in log scale).

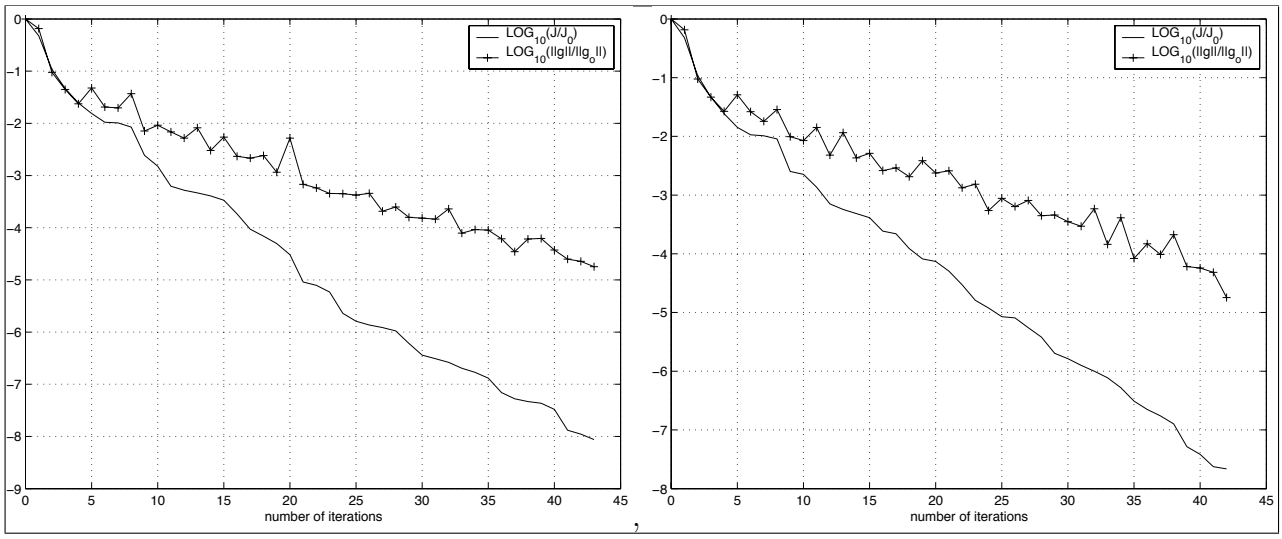

Figure 3.15: Same as in the above figure, but with limiters 3 and 4.

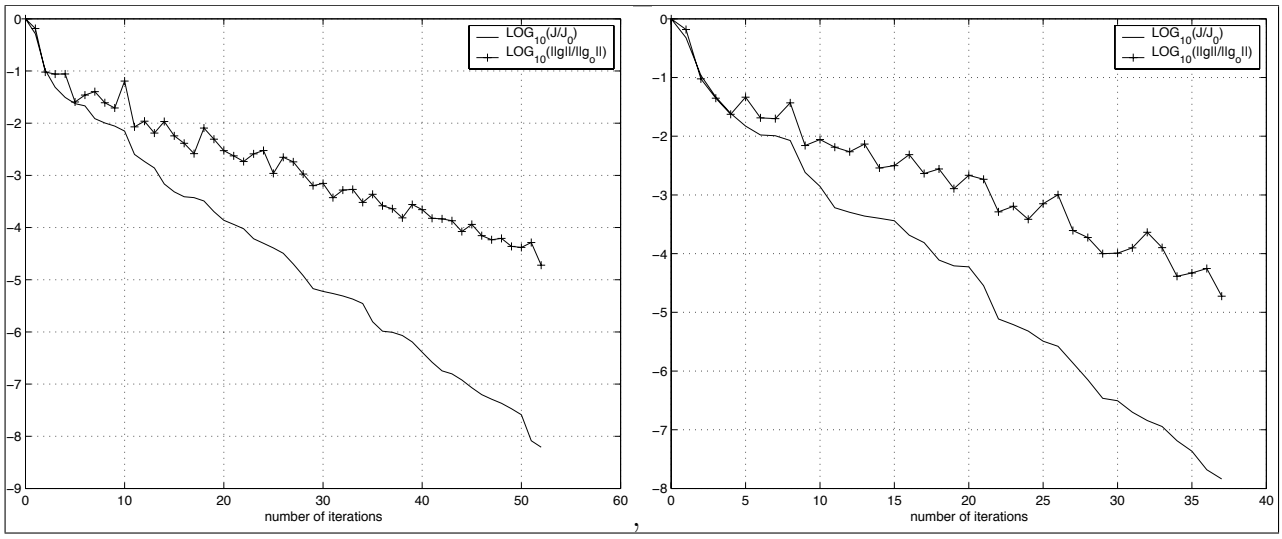

Figure 3.16: Same as in figure (3.14), but with limiters 5 and 6.

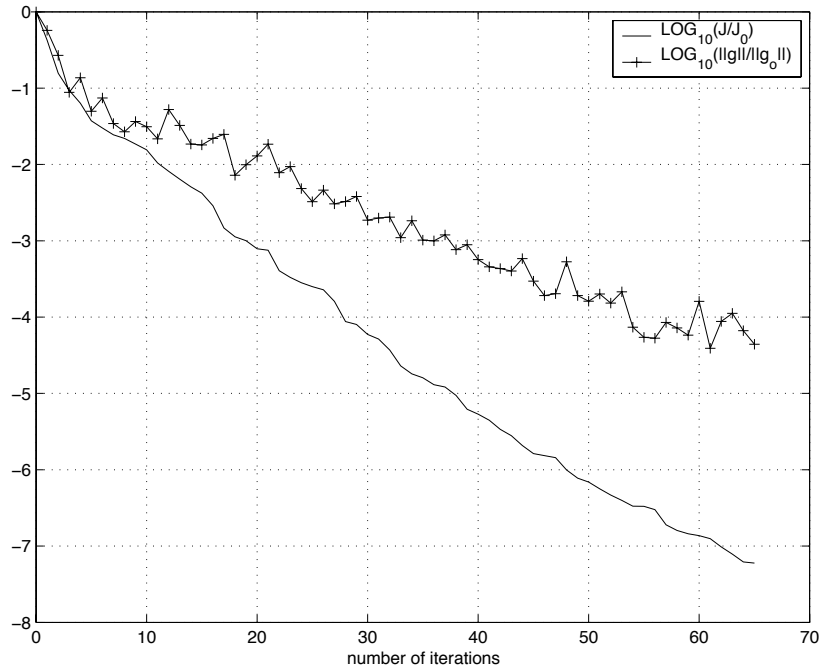

Figure 3.17: Same as in the above figure, but with the PPM scheme.

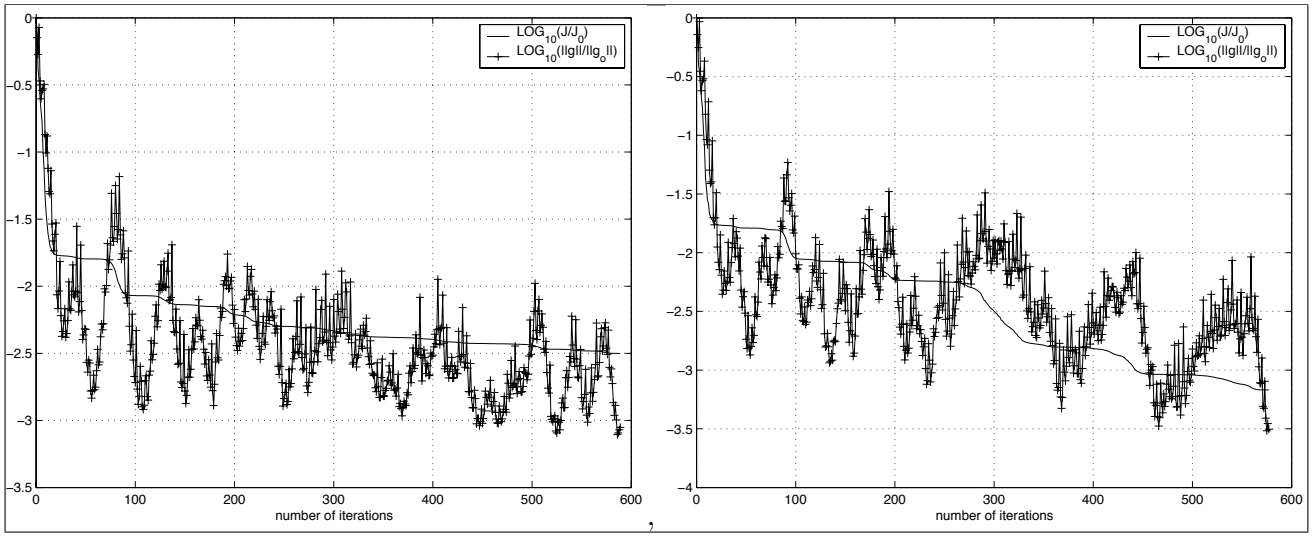

Figure 3.18: Variations of the normalized cost function $\frac{\mathcal{J}}{\mathcal{J}_0}$ and normalized gradient $\frac{\|\mathbf{g}\|}{\|\mathbf{g}_0\|}$ versus the number of minimization iterations using limiters 2 and 5 in forward and adjoint models for the 2-D global spherical SW equations model (in log scale).

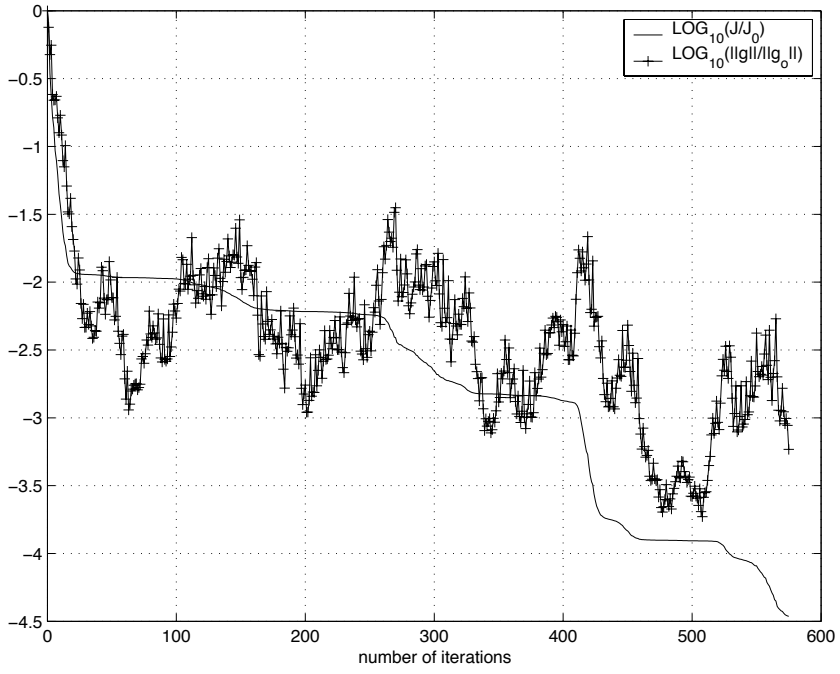

Figure 3.19: Same as in the above figure, but with the PPM scheme.

Table 3.5: Comparison of the $\mathbf{x}^{recovered}$ 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 $h(pert) - h(unpert) = 2.5993$, $u(pert) - u(unpert) = 0.1115$ and $v(pert) - v(unpert) = 8.0081 \times 10^{-2}$ for all the schemes; $T = 7\ hours$.

RMS errors for u - wind

Advection Scheme	$u^{recovered} - u(x,0)$	$u^{pert}(x,T) - u(x,T)$	$u^{recovered}(T) - u(x,T)$
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	$v^{recovered} - v(x,0)$	$v^{pert}(x,T) - v(x,T)$	$v^{recovered}(T) - v(x,T)$
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	$h^{recovered} - h(x,0)$	$h^{pert}(x,T) - h(x,T)$	$h^{recovered}(T) - h(x,T)$
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

CHAPTER 4

ESTIMATION OF MODEL ERROR IN VDA USING HIGH RESOLUTION ADVECTION SCHEMES

So far we have described the goal of variational data assimilation (VDA) as to be finding a model trajectory that best fits (in a least squared sense) the observational data over an assimilation time interval by adjustment of the initial conditions (and perhaps model parameters as well) supplied for forward model integration (Le Dimet and Talagrand 1986 [11]; Navon *et al.* 1992[10]). In the so-called *strong constraint* or *classical* version of VDA, it is assumed that the forecast model perfectly represents evolution of the actual atmosphere. The best fit model trajectory is obtained by adjusting only the initial conditions via minimization of a cost functional, subject to the model equations as strong constraint. However NWP models are imperfect, since they are discretized, dissipative and dispersion errors arise, and, moreover subgrid processes are not included. In addition, most of the physical processes and their interactions in the atmosphere are parametrized, also a complete mathematical modeling of the boundary conditions and forcing terms can never be achieved. Usually all of these modeling drawbacks are collectively addressed by the term, *model error* (ME). Following Dee (1995)[117], we would like to distinguish between forecasting and model errors. ME is one of the causes of forecasting errors, another cause being erroneous specification of initial conditions used to produce the forecast.

Studies indicate that ME can severely impact forecast errors, see Boer (1984)[118]; Dalcher and Kalnay (1987)[119]; Bloom and Shubert (1990)[120]; Zupanski and Zupanski (2002)[121]. For early methods on estimating modeling errors in operational NWP models see Thiébaux and Morone (1990)[122]; Saha (1992)[123]. Thus giving up the assumption that the model is perfect, in the context of strong constraint VDA leads us to *weak constraint* formulation of VDA, which is the main theme of this chapter; since we include time evolution

of the variables, we could say weak constraint 4D-Var (time plus three space dimensions). Instead we prefer to use the general term VDA, because have we used a two dimensional global shallow water model for presenting our results.

In sequential data assimilation using Kalman filtering theory, the inclusion of ME forms an integral part of the filter formulation, various filtering approaches which include ME have been considered by Chepurin *et al.* (2005)[124]; Dee and Todling (2000)[125]; Dee and Da Silva (1999)[126]; Dee et al. (1999)[127]; Dee and Da Silva (1998)[128]; Dee (1995)[117]; Zupanski (2005)[129]. When the number of observations is considerably smaller, the method of *representers* (Bennett 1992 [42]) provides a computationally efficient (in terms of storage/ space requirements) formulation of VDA. The incorporation of ME in such a framework has been shown by Bennett *et al.* (1993, 1996, 1997)[130, 131, 132]; Uboldi and Kamachi (2000)[133].

Model error is formally introduced as a *correction* to the time derivatives of model variables in the weak constraint formulation of VDA. Let the vector $\mathbf{x}(t)$ be used to represent the state of the atmosphere, then its evolution accounting for ME in the NWP model is written as,

$$\frac{d\mathbf{x}(t)}{dt} = \mathcal{M}[\mathbf{x}(t)] + \mathbf{T}[\boldsymbol{\eta}(t)], \tag{4.1}$$

where $\mathcal{M}[.]$ denotes all the mathematical operations involved in the NWP model, $\boldsymbol{\eta}$ represents the ME term and $\mathbf{T}[.]$ is an operator that accounts for the fact that only certain components of the state vector have modeling errors (none-the-less, $\mathbf{T}[.]$ is often set to be equal to the unit matrix). ME usually varies both spatially and temporally, and has both systematic and stochastic components. Comparing the strong and weak constraint VDA, in the formulation of former, it is assumed that η has mean, $E[\boldsymbol{\eta}(t)] = 0, \forall\, t$ and model error covariance matrix, $\mathbf{Q} = E[\boldsymbol{\eta}(t)\, \boldsymbol{\eta}^T(t')] = 0, \forall\, t \,\&\, t'$, where $E[.]$ is the mathematical expectation operator. It should be noted that if the mean and (co)variance of a random vector are prescribed to be equal to zero, then all realizations of that random vector are identically equal to zero, thus, $\boldsymbol{\eta} \equiv 0$. Whereas in the weak constraint version of VDA, the mean and covariance of ME are to be specified. However exact statistical details of ME are difficult to obtain (Daley 1992a,b[134, 135]; Dee and Da Silva 1998[128]; Zhu and Kamachi 2000[136]) a fact which led researchers to suggest a variety of assumptions to approximate ME.

Table 4.1: List of acronyms

Acronym	Definition
ECMWF	European Center for Medium-Range Weather Forecasts
hPa	Hectopascals
RMSE	Root-mean-squared error
UTC	Universal time coordinate
PPM	Piecewise parabolic method
T_{-06}	Data set from ERA-40 reanalysis project valid for 00 UTC 2 February 2001
$T00$	Data set obtained by 6 hour integration of T_{-06}
T_{+06}	12 hour integration of T_{-06}
T_{+12}	18 hour integration
T_{+18}	24 hour integration
T_{+24}	30 hour integration
T_{+30}	36 hour integration

Early efforts to model the systematic component of ME were pioneered by Derber (1989) [137]. He suggested a simplified approach to model $\boldsymbol{\eta}$ to be equal to $\lambda(t)\,\boldsymbol{\phi}$. The temporal part, $\lambda(t)$ is a specified function of time alone, and $\boldsymbol{\phi}$ is a spatially dependent, control variable. Three different forms of λ were considered, namely, parabolic, delta function and constant in time. It was observed that the parabolic variation of λ provided results comparable to a constant in time λ. Using a similar approach (Wergen 1992[138]; Zupanski 1993[139]) it was shown that inclusion of ME allows significant reduction in forecast RMSE (table 4.1 provides a list of acronyms and their definitions).

For dynamically evolving systems such as discrete `NWP` models, ME is expected to depend on the model state and should be evolving in time (Griffith and Nichols 1996, 2000 [140, 141]). Various simple forms of evolution of ME in time were considered by Griffith and Nichols (2000)[141], henceforth referred to as GN00. At any time step, t_k, the evolution of ME is assumed to be given by the following equation,

$$\eta_k = T_k(\mathbf{e}_k) + \mathbf{q}_k, \tag{4.2}$$

where $\mathbf{e}_k$ represents time-varying systematic components of ME, T_k describes the distribution of systematic errors in the `NWP` model equations, and $\mathbf{q}_k$ (stochastic component) is an unbiased, serially correlated, normally distributed random vector, with known covariance.

The evolution of $\mathbf{e}_k$, is in-turn modeled by assuming that it depends on the state vector, $\mathbf{x}_k$,

$$\mathbf{e}_{k+1} = g_k(\mathbf{x}_k, \mathbf{e}_k).$$

GN00 suggested three forms for the evolution of the above systematic component of ME,

1. constant in time: $\mathbf{e}_{k+1} = \mathbf{e}_k$, $T_k \equiv I$. It is inferred that this form is suitable for modeling errors in source terms and boundary conditions.

2. Evolving in time: $\mathbf{e}_{k+1} = \mathcal{F}_k\, \mathbf{e}_k$, $T_k \equiv I$, where $\mathcal{F}_k$ is a linear model, which is appropriate for representing discretization errors.

3. Spectral form: $\mathbf{e}_{k+1} = \mathbf{e}_k$, T_k is a block diagonal matrix, with diagonal entries given by $I, I \cdot \sin(\frac{\kappa}{N\tau}), I \cdot \cos(\frac{\kappa}{N\tau})$, where τ is a constant time scale.

It is to be noted that the control of ME as well as the model initial conditions in weak constraint `VDA` doubles the size of the optimization problem (compared to strong constraint `VDA`), in addition if the stochastic component is included in the ME formulation, then one would have to save every random realization at each model time step, which amounts to tripling the size of the optimization problem. The computational results in GN00 were provided by neglecting $\mathbf{q}_k$, the stochastic component of ME and using the constant and evolving forms of the systematic component, see GN00 for additional details. Similar approaches for modeling the systematic component of ME was considered by Martin *et al.* (2002) [142] and reduction of ME control vector size by projecting it on to the subspace of eigenvectors corresponding to the leading eigenvalues of the adjoint-tangent linear operators was illustrated by Vidard *et al.* (2000)[143].

The above described approach (of GN00) provides the systematic component of ME at any discrete time step, t_k, in other words, the evolution of ME has been considered as a *discrete process*. Vidard *et al.* (2004)[144] (from now onwards referred to as VPLD04) considered a continuous in time form for the evolution of ME. This approach is consistent with the fact that model equations are first written as differential equations and then descretized in space and time. If the initial ME, $\boldsymbol{\eta}(t_0) = \boldsymbol{\eta}_0$, then VPLD04 modeled the evolution of ME as,

$$\frac{d\eta}{dt} = \Phi[\boldsymbol{\eta}(t), \mathbf{x}(t)] + \mathbf{q}(t), \tag{4.3}$$

where $\mathbf{q}(t)$ is the stochastic component of ME. Once again, neglecting the stochastic component reduces the size of the control vector, as in the case considered by VPLD04; they also assumed that $\Phi[\boldsymbol{\eta}(t), \mathbf{x}(t)] = \boldsymbol{\eta}(t)$. This implies that the evolution of ME term is modeled by the following simple exponential growth equation,

$$\frac{d\boldsymbol{\eta}}{dt} = \boldsymbol{\eta}(t).$$

Such a *deterministic* approach to model the evolution of ME significantly simplifies the weak constraint `VDA`, since only the initial ME ($\boldsymbol{\eta}_0$) is to be obtained via solution of the optimization problem (see VPLD04 for additional details).

Daley (1992a)[134] suggested that model error is correlated in time and used a Markov process to model its evolution in a simple Kalman filtering (KF) framework. The Markovian assumption is based on the observation that as the numerical model is integrated in time, errors show a trend of serial correlation in both time and space.The most important property of a Markov process is that the state at any time in future is dependent only on its present value, but not on its value in the past. Considering the model error as a Markov process, at any two successive time steps, t_k and t_{k+1},

$$\boldsymbol{\eta}_{k+1} = \mu \mathbf{G}_k[\boldsymbol{\eta}_k] + (1-\mu)\mathbf{q}_k, \tag{4.4}$$

where μ is a scalar, such that $0 \leqslant \mu < 1$, $\mathbf{q}_k$ is the random component of ME and $\mathbf{G}_k[.]$ is a linear operator (see Daley, 1992a[134] for discussion on the implementation of two different forms of this operator, again in a KF setting). Using $\mathbf{G}_k \equiv I$, in the above form of ME, D. Zupanski (1997)[145] and Zupanski *et al.* (2005)[146] (henceforth referred to as ZZ05) provided results obtained using the NCEP' s regional weather prediction system in weak constraint `VDA` framework. They assumed the initial value of the model error, $\boldsymbol{\eta}_0$ to be equal to zero and a coarse time scale for the evolution of $\mathbf{q}_k$, such that at-most four of such vectors are present in a 12 hour time window of data assimilation (to limit the size of control vector in minimization). In addition, all of those $\mathbf{q}_k$' s were obtained via minimization of the weak constraint `VDA` cost functional, thereby implicitly assuming that each $\mathbf{q}_k$ is a *m*ean deterministic forcing term. The model error covariance matrix was derived in a novel way, please see ZZ05 and references therein for details.

Often discontinuities (on the scale of the model grid resolution) in solutions to `NWP` models

arise due to sharp fronts formed in low-pressure systems, hydraulic jumps, etc. High resolution advection schemes provide means to capture such discontinuous solutions accurately. A consistent formulation to improve the accuracy of the numerical high resolution advection scheme, ranging from first order to second and third order accuracy in space was provided in chapter 3 (see Akella and Navon (2005) [147], hereafter referred to as AN05). In the same chapter was provided a comparison of the impact of using different slope limited monotone upstream centered schemes for conservation laws (MUSCL) on data assimilation (in strong constraint form) in one space dimension using a viscous Burgers equation model and in two dimensions using a shallow water equations model. Another consistent method of decreasing the discretization errors (principally truncation errors) is to refine the model resolution, such an approach in VDA was suggested by Le Dimet and Shutyaev (2005)[148], henceforth referred to as LDS05. However this approach is limited by the resolution of the observational system, indeed one of the conclusions of LDS05 was that the improvement in predictability is most sensitive to the observational errors. In the present chapter we extend the studies in AN05, by conducting various VDA experiments in a more practical setting than that presented in AN05. In particular the cost functional includes $\mathcal{J}_b$, the background cost functional, the formulation of which seriously impacts the performance of data assimilation system. The square root formulation using linear balance operator and diffusion operator has been used so that the inverse of the background error covariance matrix is not required to be specified (details are described in appendix C).

Following is the outline of the present chapter. In the next section we focus on the issue of uncertainty in the specification of initial conditions only (recall that it contributes to forecast errors) with no ME term in VDA. We first show that changing the advection scheme used in discretization of the non-linear terms in the governing equations (which can be considered as altering the numerical model) leads to a decrease in forecasting error. Next we provide results obtained using various schemes by conducting VDA in strong constraint form, and once again an improvement in predictability is achieved by improving the numerical model used in VDA. Further on we focus on the issue of accounting for model error in VDA, via weak constraint formulation. We provide a detailed formulation of feasible forms of modeling the ME. Using three different forms of modeling the evolution of ME, an analysis of the obtained results is discussed.

4.1 Impact of using different numerical advection schemes: Model forecasts

In this chapter we have used the same global two-dimensional shallow water equations (SWE) model for numerical experiments and analysis, that was used in the previous chapter, namely the explicit flux-form semi-Lagrangian, finite volume shallow water equations model of Lin and Rood (1997)[68], henceforth referred to as LR97. This model serves as the dynamical core in the community atmosphere model (CAM), version 3.0 [106], and its operational version implemented at NCAR and NASA is known as finite volume-general circulation model (FV-GCM). We will follow the suggestion in LR97, and always use unconstrained van Leer scheme to advect winds on the C-grid (this strategy provides solutions whose accuracy is comparable to those obtained by using more CPU demanding advection schemes, for e.g., constrained van Leer and PPM schemes), except for the first order advection scheme, in which case, we will use first order scheme on both C- and D-grids. Therefore on the D-grid, we will be using the unconstrained, constrained van Leer, and the PPM schemes. Using the finite volume method, within each cell of the discrete grid, if we consider a piecewise linear approximation to the solution, whose slope is *limited* in a certain way depending on the values of the solution at the neighboring grid cells, one can consistently derive a family of van Leer schemes. Alternatively, if we assume a piecewise parabolic approximation to the solution within each cell, then we obtain the PPM scheme. For further details on formulation of these schemes see Lin *et al.* (1994)[64] and AN05. From now onwards we will use the following convention to refer to our **test cases**,

1. *first order advection scheme*: first order advection on both C- and D-grids,
2. *unconstrained van Leer scheme*: unconstrained van Leer on both C- and D-grids,
3. *constrained van Leer scheme*: constrained van Leer on D-grid and unconstrained van Leer scheme on C-grid,
4. *PPM scheme*: PPM scheme on D-grid and unconstrained van Leer scheme on C-grid.

Further details of the model can be found in LR97 and references therein. Also a comparison of these schemes for the test cases proposed in Williamson *et al.* (1992)[101] are provided in LR97.

Unless specified otherwise, here we consider a regular latitude-longitude discretization on the sphere, using a $2.5^o \times 2.5^o$ grid resolution, and a time step of $\Delta t = 450$ seconds. Reanalyzed data at 500 hPa pressure level obtained from the ERA-40, ECMWF 40-year reanalysis (ECMWF (2002)[149]) system, valid for 00 UTC 2 February 2001 (henceforth denoted by T_{-06}) was used to specify the geopotential height field (winds fields were obtained using geostrophic assumption) as initial conditions for forward model integration. Using the above specified advection schemes, we integrated the model for 36 hours, saving forecasts at every 6 hour interval. In the forecast and adjoint models, to introduce systematic errors ω (angular velocity of the earth) was set to 0.95 times the value used for generating observations, which was specified to be equal to $7.292 \times 10^{-5} s^{-1}$. In order to obtain the observations, a twin experiment framework is considered. To simulate real-life *noisy* observations, a 1% random perturbation in the initial conditions prescribed at T_{-06} was added and the PPM advection scheme was used on both C- and D- grids (though the PPM scheme is expensive to implement, it provides a very accurate forecast, see LR97, AN05). We integrated the model for 36 hours, once again saving the states after every 6 hours, as *observations*, see table (4.1) for nomenclature of the different time intervals. RMSE between model forecasts and observations are provided for first order, unconstrained van Leer, constrained van Leer and PPM schemes (test cases: 1- 4) in Fig. 4.1. The RMSE indicate a trend of decreasing errors, the first order scheme being the most erroneous whereas the PPM scheme being characterized by the least error when compared with the observational data (since the first order scheme performs poorly when compared to the other schemes, we will discontinue its usage in our further studies), and the unconstrained van Leer scheme exhibits larger errors than the constrained van Leer scheme. These results are consistent with previous results (see for instance AN05; LR97; Lin *et al.* (1994)[64]), the larger implicit diffusive property of the van Leer schemes (the constrained van Leer scheme is better than the unconstrained van Leer scheme due to the monotonicity constraint applied in the former) when compared to the PPM scheme has been argued to be the reason for the above trend in errors. In the following section, we further analyze these schemes, particularly in the context of strong constraint `VDA` (thus dealing with the issue of erroneous specification of initial conditions only).

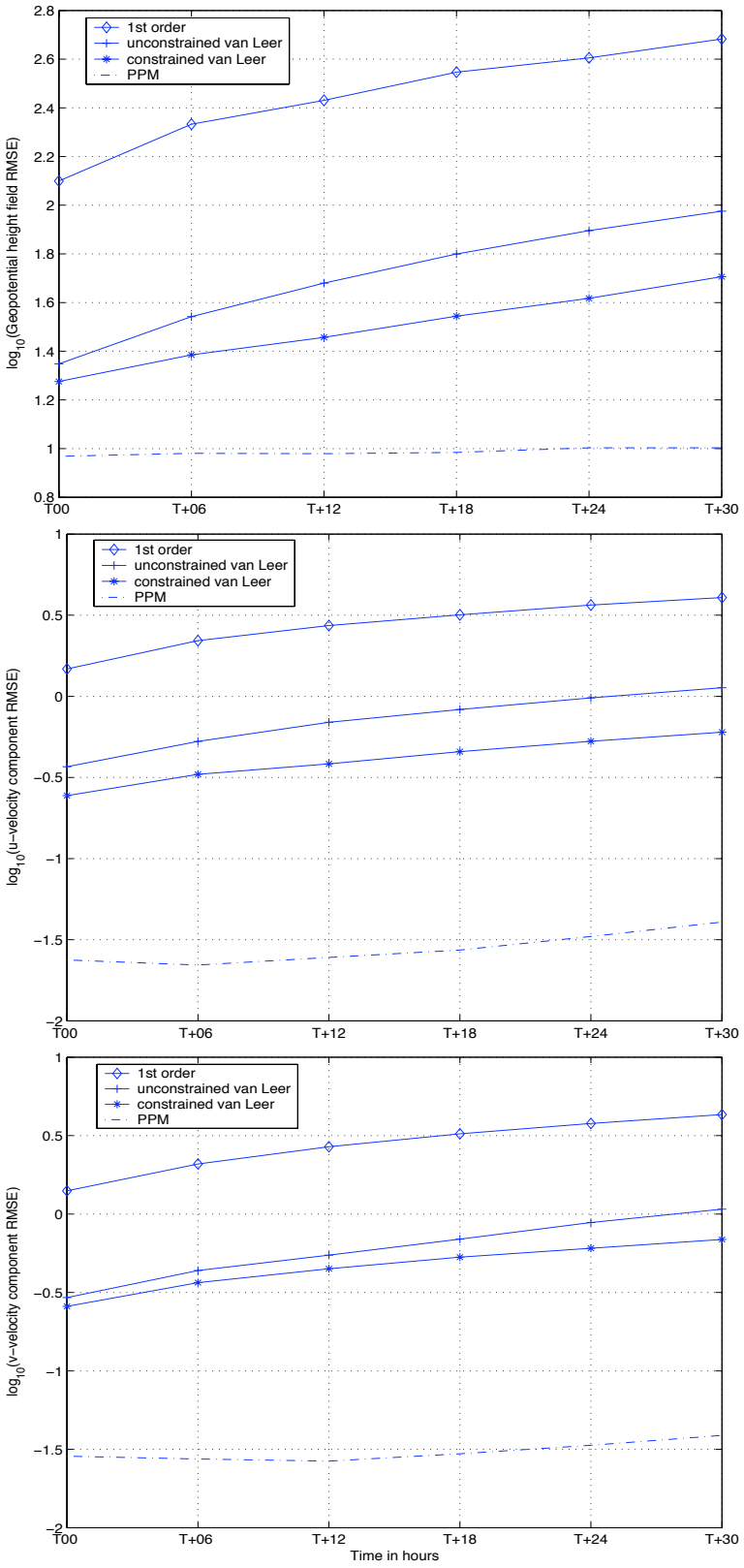

Figure 4.1: RMSE in the geopotential height and wind fields for different advection schemes.

4.2 VDA experiments in strong constraint formalism

Data assimilation schemes determine the analysed atmospheric state as an optimal combination of a-priori background information and observational information. Let $\mathbf{x}^t$ be the *true* state of the atmosphere, $\mathbf{x}^b$ be the background field and $\mathbf{y}^o$ denote the observations. Usually a short-range forecast provides $\mathbf{x}^b$. Then the error in the background field is equal to $\mathbf{e}^b = \mathbf{x}^b - \mathbf{x}^t$, and the error in the observations field is given by $\mathbf{e}^o = \mathbf{y}^o - \mathbf{H}(\mathbf{x}^t)$. $\mathbf{H}$ is an observation operator that maps model variables to observations, if all the model variables are observed and observations occur at every model grid point, then $\mathbf{H} \equiv I$. Denoting the mathematical expectation operator by $E[.]$, the background error and observation error covariances are given by $\mathbf{B} = E[(\mathbf{e}^b - E[\mathbf{e}^b])(\mathbf{e}^b - E[\mathbf{e}^b])^T]$ and $\mathbf{R} = E[(\mathbf{e}^o - E[\mathbf{e}^o])(\mathbf{e}^o - E[\mathbf{e}^o])^T]$, respectively. Error covariances measure the uncertainty involved with both of these data sources, hence they determine the quality of data assimilation. Due to lack of knowledge of the true state of the atmosphere, $\mathbf{x}^t$, we can only guess what $\mathbf{B}$ and $\mathbf{R}$ should be. Thus they are approximations of the *true* error covariances.

In strong constraint version of VDA, neglecting the ME, minimization of the following nonlinear quadratic cost functional, $\mathcal{J}_o$, accomplishes the goal of fitting model states $(\mathbf{x}(t_i))$ and observations $(\mathbf{y}^o(t_i))$ in an assimilation time interval, $[t_0, t_n]$. Often the so-called background cost, $\mathcal{J}_b$ is added to $\mathcal{J}_o$ to regularize the following cost functional. Its minimization with respect to the initial state, $\mathbf{x}(t_0)$ as a control variable (Kalnay 2003[1]),

$$\begin{aligned} \mathcal{J}[\mathbf{x}(t_0)] = \mathcal{J}(\mathbf{x}_0) &= \underbrace{\frac{1}{2}[\mathbf{x}(t_0) - \mathbf{x}^b]^T \mathbf{B}^{-1} [\mathbf{x}(t_0) - \mathbf{x}^b]}_{\mathcal{J}_b} \\ &\quad + \underbrace{\frac{1}{2}\sum_{i=0}^{n} [H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)]^T \mathbf{R}^{-1} [H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)]}_{\mathcal{J}_o}, \end{aligned} \tag{4.5}$$

subject to the following model equations as strong constraint,

$$\begin{aligned} \mathbf{x}(t_0) &= \mathbf{x}_0, \\ \frac{d\mathbf{x}(t)}{dt} &= \mathcal{M}[\mathbf{x}(t)], \end{aligned} \tag{4.6}$$

is achieved by using iterative minimization algorithms, such as quasi-Newton or truncated-Newton methods. These algorithms require availability of gradient of the cost functional with respect to the control variables, which is in-turn efficiently obtained by backward integration of the adjoint model (Lorenc 1986[12]; Navon *et al.* 1992[10]; Zou *et al.* 1993[116]). Note that in the above model equations, we did not account for ME, i.e, $\boldsymbol{\eta}(t) \equiv 0, \forall t$.

One of the principal causes of observational errors is instrumentation error, which is sequentially correlated in space and time. Accurate specification of the observation error statistics is very important in the implementation of data assimilation techniques. However in this study we deal with model errors and the observational error and background error covariances have been assumed to be invariant in time, for further details regarding the impact of observational errors on data assimilation, please see Daley (1992a, 1993)[134, 150]. Further we assume that the observations are not biased and that the background state and observations are mutually uncorrelated.

The formulation of the $\mathcal{J}_b$ term is crucial to the performance of the data assimilation system. Considering a single observation, at a single grid point, the analysis increment is proportional to a column of $\mathbf{B}$. Hence background error covariance spreads out the information in the analysis from the observations and provides statistically consistent increments at the neighboring grid points and levels of the model. It also ensures that observations of one model variable produce dynamically consistent increments in the other model variables. Using background knowledge makes the `VDA` problem well-posed even when there are only a few observations, also it fills any data voids with *good quality* information (Navon *et al.* 2005[151]). In addition the background state, $\mathbf{x}^b$ provides an initial guess for minimization of $\mathcal{J}$. Ideally the optimal design of background error covariance should take into account the average variances, autocorrelations and balance properties of the background errors, so that the covariances of short range forecast errors in data assimilation are adequately represented (Derber and Bouttier 1999[152]). Dynamic or flow dependent formulation of $\mathbf{B}$ could improve analyses and subsequent forecasts (Riishøjgaard 1998[153]), particularly if the observations are nonuniformly distributed. However most of the studies and in particular, operational implementations use a *static* background error covariance; since the focus of this paper is model error, we do not deal with these issues anymore and follow the approaches of Weaver and Courtier (2001)[154]; Derber and Bouttier (1999)[152] to construct $\mathbf{B}$ as a multivariate and cross correlated operator, see appendix C for further

details (see also Gaspari and Cohn 1999[155] for an alternative formulation).

We used the same observations as used in the previous section to compare different model forecasts to conduct DA using different advection schemes for a time interval of 24 hours in a twin experiment framework. These observations were obtained by introducing random perturbations in the initial condition, which can be looked upon as introducing uncertainty in initial conditions and using a slightly different version of the model (namely, using PPM advection scheme on both C- and D-grids). The fact that observations were generated by usage of a different model than that used for DA, introduces a systematic model bias. We conducted three DA experiments, in each case, we used either unconstrained van Leer (test case 2), constrained van Leer (test case 3) or PPM advection scheme (test case 4) in both forward and adjoint modes (see AN05 for details on derivation of the adjoint model for high resolution advection schemes).

The background state, $\mathbf{x}^b$, which was the first guess for DA was obtained by a 6 hour forward integration of the reanalyzed data, at time, T_{-06}; see table (4.1) and Fig. 4.2 for naming and illustration of the different data sets used in DA experiment. Five observational data sets (at times, $T_{00}, T_{+06}, T_{+12}, T_{+18}, T_{+24}$, such that every 6 hours we have an observation) within a 24 hour interval are assimilated using the observations to model space operator, $\mathbf{H} = I$, i.e, the observations occur at every grid point of the model resolution. The observation error covariance matrix has been taken to be a block diagonal matrix, $R = [10^4 I, 100I, 100I]$, such that observational errors at every grid point are only autocorrelated and stationary in time. We used an unconstrained limited memory quasi-Newton (L-BFGS) minimization algorithm (Liu and Nocedal 1989[113]; Nash and Nocedal 1991[114]) for minimization of the cost functional given in Eq. (4.5). The following termination criteria was used to conduct DA experiments,

$$\|(\nabla \mathcal{J})_k\| \leq EPS \cdot MAX(1, \|\mathbf{x}_k(t_0)\|), \tag{4.7}$$

where $\|\cdot\|$ is the L_2 norm, $(\nabla \mathcal{J})_k$ is the gradient and $\mathbf{x}_k(t_0)$ is the optimal initial state vector at the $k^{th.}$ minimization iteration, and EPS was set to 5×10^{-5}.

During the minimization process, due to the regularization property of the minimization algorithm, the differences on larger scales are fit in the first few iterations, yielding the largest decrease in the cost functional, thereafter minimization proceeds to fit the smaller discrepancies, or small decreases in the value of the functional. In general the observations

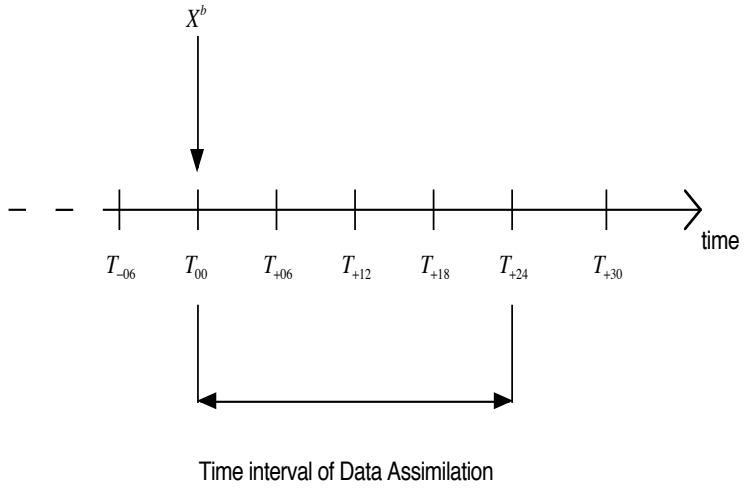

Figure 4.2: Illustration of data assimilation time window

occurring in the middle of the DA time window (in this case, at times, T_{+12} and T_{+18}) are best fit. We used the same minimization termination criteria given by Eq. (4.7) for all three DA experiments, and it took 114 function and gradient evaluations for the unconstrained van Leer, 78 for the constrained van Leer and the least, 63 for the PPM. A comparison of the RMSE in geopotential height field before and after DA is provided in Fig. 4.3 for the different advection schemes. Comparison of the differences in geopotential height field between model forecast using PPM advection scheme and observations at T_{+30}, forecast verification time, before and after data assimilation is provided in fig. 4.4 (similar results were obtained for the wind fields, as well as, using the unconstrained and constrained van Leer advection schemes). In all three cases we achieved more than 50% reduction in the RMSE by DA. Also the optimized initial condition is able to provide a better forecast at T_{+30} (30 hour forward integration) in all the cases. Clearly the PPM scheme is much superior when compared to the constrained and unconstrained van Leer schemes using lesser number of function and gradient computations in achieving the same reduction in forecast RMSE. We have demonstrated that it is possible to decrease the component of forecasting error associated with the mis-specification of initial conditions only by consistently improving the numerical advection scheme used for discretizing the nonlinear advection terms in the model, holding everything else fixed (resolution of the model and observational system as well). In the next

section we will study the impact of introducing various forms of model error in VDA via weak constraint formulation.

4.3 Weak Constraint VDA formulation and results from various experiments

Now we turn our attention to one other cause of forecasting error, namely the model error. In the strong constraint VDA, the model equations are assumed to be *perfect*, therefore modeling errors (whose causes have been described earlier in the introduction) are not taken into account. The weak constraint VDA provides a framework for incorporation of ME in the model equations, via explicit introduction of an extra term, $\boldsymbol{\eta}(t)$, (as in Eq. 4.1),

$$\frac{d\mathbf{x}(t)}{dt} = \mathcal{M}[\mathbf{x}(t)] + \mathbf{T}[\boldsymbol{\eta}(t)].$$

The operator $\mathbf{T}$ maps the space of the ME to the space of the model state, $\mathbf{x}$. If one has a-priori knowledge that the numerical model has some drawbacks, for e.g., modeling of the atmosphere in certain regions of the globe, (say, one of the poles) then the operator, $\mathbf{T}$ should be specified in such a way that only those model grid points (at the poles) have modeling errors, and the rest of the model states do not have any ME. In the literature (for instance, see GN00 and VPLD04) it has often been assumed that the model state at every grid point has an associated error, which implies that $\mathbf{T}$ is identically equal to the unit matrix, I, and, the dimension of $\boldsymbol{\eta}$ is equal to that of the model state, $\mathbf{x}$; in the present article we assume $\mathbf{T} = I$.

Past research work by Dee and Da Silva (1998)[128] indicated that ME has contributions that are both systematic (or, deterministic) and random in nature. Following Derber (1989)[137] and VPLD04 in the spirit of *variational continuous assimilation*, we will model the evolution of ME as a continuous process, using the following initial value problem (IVP), which is a continuous-in-time differential equation, (Eq. 4.3),

$$\frac{d\boldsymbol{\eta}}{dt} = \Phi[\boldsymbol{\eta}(t), \mathbf{x}(t)] + \mathbf{q}(t).$$

In this article, we are concerned only with the systematic part of ME, hence, the stochastic component, $\mathbf{q}(t)$, is neglected; the sequential filtering approaches based on KF, such as the ensemble KF provide an implicit framework for the inclusion of the stochastic terms.

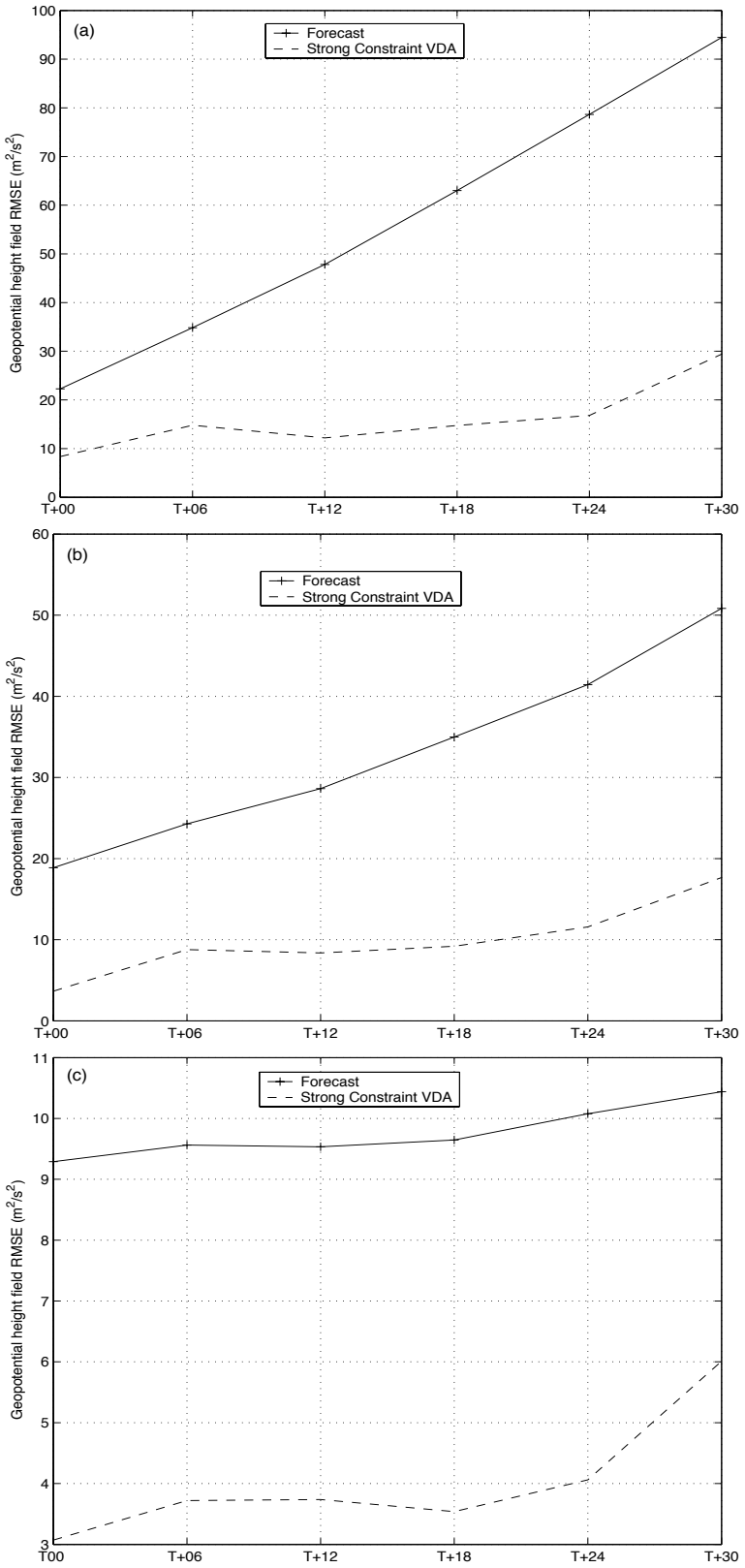

Figure 4.3: RMSE in the geopotential height fields for different advection schemes before and after DA in strong constraint form (a) Unconstrained Van Leer (b), Constrained van Leer (c), PPM schemes

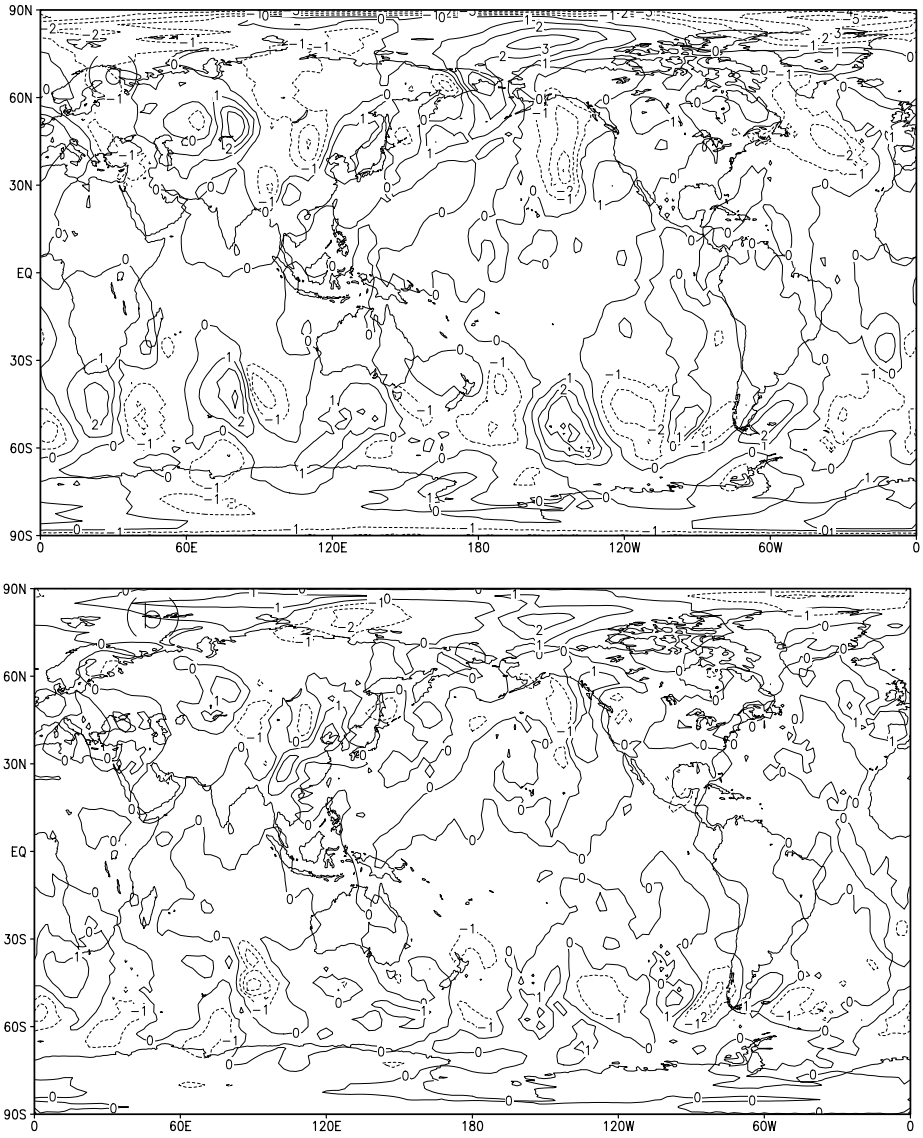

Figure 4.4: Isolines of differences in geopotential height field between model forecast using PPM scheme for advection and observations at T_{+30}, forecast verification time (a), Using $\mathbf{x}^b$ (b), using initial condition obtained after strong constraint data assimilation

Therefore the above differential equation simplifies to,

$$\frac{d\boldsymbol{\eta}}{dt} = \Phi[\eta(t), \mathbf{x}(t)]. \tag{4.8}$$

For closure of the above IVP, we need to specify the initial value of $\boldsymbol{\eta}(t_0) = \boldsymbol{\eta}_0$, and the nature of the mapping, $\Phi[.]$; both of them being very important parameters. Uncertainty in the initial value of ME ($\boldsymbol{\eta}_0$) is similar to uncertainty involved with the determination of initial conditions for the model equations, solution to which was accomplished by minimization of the cost functional given in Eq. (4.5). A similar approach is adopted for obtaining $\boldsymbol{\eta}_0$, via minimization of a cost functional, which is similar to that in Eq. (4.5), and includes the following extra term, $\mathcal{J}_{\boldsymbol{\eta}}$, First we describe the methodology used to calculate the initial value of ME and then address the issue of different approaches for modeling the evolution of ME, using different forms of $\Phi[.]$. To obtain $\boldsymbol{\eta}_0$ the following weak constraint `VDA` cost functional ($\mathcal{J}$) is minimized (note that it is similar to the cost functional in Eq. (4.5), but includes an extra term, $\mathcal{J}_{\boldsymbol{\eta}}$),

$$\begin{aligned}
\mathcal{J}[\mathbf{x}(t_0), \boldsymbol{\eta}(t_0)] &= \mathcal{J}(\mathbf{x}_0, \boldsymbol{\eta}_0) \\
&= \underbrace{\frac{1}{2}[\mathbf{x}_0 - \mathbf{x}^b]^T \mathbf{B}^{-1} [\mathbf{x}_0 - \mathbf{x}^b]}_{\mathcal{J}_b} \\
&+ \underbrace{\frac{1}{2}\sum_{i=0}^{n}[H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)]^T \mathbf{R}^{-1} [H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)]}_{\mathcal{J}_o}, \\
&+ \underbrace{\frac{1}{2}[\eta_0 - \eta^b]^T \mathbf{Q}^{-1} [\eta_0 - \eta^b]}_{\mathcal{J}_{\boldsymbol{\eta}}},
\end{aligned} \tag{4.9}$$

where $\mathbf{Q}$ is the model error covariance matrix (explained in appendix D). Just as the background state, $\mathbf{x}^b$ was used as an initial guess for $\mathbf{x}_0$, to minimize the strong constraint `VDA` cost functional, we use $\boldsymbol{\eta}^b$ as an initial guess for $\boldsymbol{\eta}_0$ to minimize the above $\mathcal{J}_{\boldsymbol{\eta}}$ in the weak constraint `VDA`. Hence the above cost functional, $\mathcal{J}(\mathbf{x}_0, \boldsymbol{\eta}_0)$ is minimized subject to the following equations as constraints,

$$\left.\begin{aligned}
&\mathbf{x}(t_0) = \mathbf{x}_0; \boldsymbol{\eta}(t_0) = \boldsymbol{\eta}_0, \\
&\frac{d\mathbf{x}(t)}{dt} = \mathcal{M}[\mathbf{x}(t)] + \boldsymbol{\eta}(t); \frac{d\boldsymbol{\eta}}{dt} = \Phi[\boldsymbol{\eta}(t), \mathbf{x}(t)].
\end{aligned}\right\} \tag{4.10}$$

Introducing the following augmented Lagrangian functional, the above constrained minimization problem becomes an unconstrained problem,

$$\begin{aligned}\mathcal{L}(\mathbf{x},\boldsymbol{\eta},\mathbf{x}^*,\boldsymbol{\eta}^*) &= \mathcal{J}(\mathbf{x}_0,\boldsymbol{\eta}_0) \\ &+ \int_{t_0}^{t_n} \langle\, \mathbf{x}^*,\, \{\frac{d\mathbf{x}(t)}{dt} - \mathcal{M}[\mathbf{x}(t)] - \boldsymbol{\eta}(t)\}\rangle\, dt \\ &+ \int_{t_0}^{t_n} \langle\, \boldsymbol{\eta}^*,\, \{\frac{d\boldsymbol{\eta}}{dt} - \Phi[\boldsymbol{\eta}(t),\mathbf{x}(t)]\}\rangle\, dt, \end{aligned} \tag{4.11}$$

where $\mathbf{x}^*, \boldsymbol{\eta}^*$ are the Lagrange multiplier vectors corresponding to $\mathbf{x}, \boldsymbol{\eta}$ respectively, in other words, $\mathbf{x}^*$ is the adjoint state corresponding to $\mathbf{x}$ and $\boldsymbol{\eta}^*$ is the adjoint state of $\boldsymbol{\eta}$, and $\langle\cdot,\cdot\rangle$ denotes Euclidean inner product.

Using calculus of variations, the extrema of $\mathcal{L}$ are the solutions of the Euler-Lagrange equations (the extrema of $\mathcal{L}$ are the same as the extrema of $\mathcal{J}(\mathbf{x}_0,\boldsymbol{\eta}_0)$). Using the first order optimality criteria, at the extrema of the Lagrangian, $\mathcal{L}$, the following equations are satisfied,

$$\frac{\partial\mathcal{L}}{\partial\mathbf{x}} = 0, \frac{\partial\mathcal{L}}{\partial\boldsymbol{\eta}} = 0, \tag{4.12a}$$

$$\frac{\partial\mathcal{L}}{\partial\mathbf{x}^*} = 0, \frac{\partial\mathcal{L}}{\partial\boldsymbol{\eta}^*} = 0. \tag{4.12b}$$

Equations (4.12b) yield the equations describing the evolution of model state and ME,

$$\frac{d\mathbf{x}(t)}{dt} = \mathcal{M}[\mathbf{x}(t)] - \boldsymbol{\eta}(t), \frac{d\boldsymbol{\eta}}{dt} = \Phi[\boldsymbol{\eta}(t),\mathbf{x}(t)].$$

respectively. Equations (4.12a) yield the following adjoint equations which describe the evolution of the adjoint variables $\mathbf{x}^*, \eta^*$,

$$\mathbf{x}^*(t_n) = 0,\ \boldsymbol{\eta}^*(t_n) = 0, \tag{4.13a}$$

$$-\frac{d\mathbf{x}^*(t)}{dt} = [\frac{\partial\mathcal{M}}{\partial\mathbf{x}}]^T\mathbf{x}^* + [\frac{\partial\Phi}{\partial\mathbf{x}}]^T\boldsymbol{\eta}^* + \delta(t-t_i)\sum_{i=0}^{n}[\frac{\partial H}{\partial\mathbf{x}}]^T\,\mathbf{R}^{-1}\,[H(\mathbf{x}(t_i)) - \mathbf{y}^o(t_i)], \tag{4.13b}$$

$$-\frac{d\boldsymbol{\eta}^*(t)}{dt} = [\frac{\partial\Phi}{\partial\boldsymbol{\eta}}]^T\boldsymbol{\eta}^* + \mathbf{x}^*. \tag{4.13c}$$

Note that the evolution of $\mathbf{x}^*$ and $\boldsymbol{\eta}^*$ is coupled via the $\Phi[.]$ operator. Also the gradient of the cost functional, $\mathcal{J}(\mathbf{x}_0,\boldsymbol{\eta}_0)$, with respect to the model state, $\mathbf{x}_0$ and ME state, $\boldsymbol{\eta}_0$ is given

by,

$$\nabla_{\mathbf{x}_0}\mathcal{J} = \nabla_{\mathbf{x}_0}\mathcal{J}_b + \nabla_{\mathbf{x}_0}\mathcal{J}_o = \mathbf{B}^{-1}[\mathbf{x}_0 - \mathbf{x}^b] + \mathbf{x}^*(t_0), \tag{4.14a}$$

$$\nabla_{\boldsymbol{\eta}_0}\mathcal{J} = \nabla_{\boldsymbol{\eta}_0}\mathcal{J}_{\boldsymbol{\eta}} + \nabla_{\boldsymbol{\eta}_0}\mathcal{J}_o = \mathbf{Q}^{-1}[\boldsymbol{\eta}_0 - \boldsymbol{\eta}^b] + \boldsymbol{\eta}^*(t_0), \tag{4.14b}$$

as usual, backward integration of the adjoint models (4.13b) and (4.13c) from time $t_n \rightarrow t_0$, provides us the values of initial adjoint states: $\mathbf{x}^*(t_0)$ and $\boldsymbol{\eta}^*(t_0)$. Therefore the gradient in weak constraint `VDA`is given by $(\nabla_{\mathbf{x}_0}\mathcal{J}, \nabla_{\boldsymbol{\eta}_0}\mathcal{J})^T$. Comparing this to the gradient in strong constraint `VDA`, which was only $\nabla_{\mathbf{x}_0}\mathcal{J}$, the size of the optimization problem is doubled. Recall that $\mathbf{T}$ was set equal to I, hence the size of the initial ME control vector, $\boldsymbol{\eta}^*(t_0)$ is equal to size of the initial model state control vector, $\mathbf{x}_0$.

In strong constraint version of `VDA` we used a square-root formulation for the background error covariance matrix, $\mathbf{B}$, and transformed the space in which minimization was performed, such that there was no need for calculating $\mathbf{B}^{-1}$. In appendix D we provide a similar treatment which involves both $\mathbf{B}$ and the ME covariance matrix, $\mathbf{Q}$, thereby circumventing the need to specify $\mathbf{B}^{-1}$ and $\mathbf{Q}^{-1}$.

Now we address possible approaches to model the evolution of ME, using different forms of the mapping, $\Phi[\boldsymbol{\eta}(t), \mathbf{x}(t)]$, which maps the space of state variables, $\mathbf{x}$ and the space of ME, $\boldsymbol{\eta}$, on-to $\boldsymbol{\eta}$ only. As noted above, this mapping couples the evolution of the adjoint variables corresponding to the model states and ME variables, and it also increases the complexity involved in the backward integration of the adjoint models, Eq. (4.13). To the best of our knowledge, the issue of model errors in solutions of inverse problems using high resolution advection schemes has not been addressed as yet, hence to begin with, in this article we assume that $\frac{\partial \Phi}{\partial \mathbf{x}} = 0$, i.e, $\Phi[.]$ maps ME on-to itself. This assumption significantly simplifies the adjoint model equations, since the evolution of $\mathbf{x}^*$ is unchanged, and we can concentrate only the evolution of ME and its corresponding adjoint state.

The strong constraint `VDA` can significantly reduce the component of forecast errors due to inaccurate specification of model initial conditions. Therefore through weak constraint `VDA`, we aspire to further reduce the forecasting errors by reduction of errors such as those arising from discretization. We have used a range of schemes which have different dissipative and dispersive errors, the unconstrained van Leer being most dissipative among all three of the advection schemes and the PPM scheme, which is well known to be least dissipative

and dispersive. Hence we expect that efficient modeling of the evolution of ME should provide further improvement of results obtained using the unconstrained van Leer scheme, for example. Note that other causes of model error such as those in limited area models due to mis-specification of boundary conditions can be tackled by following the approach used in ZZ05. Based on our experience about the evolution of forecast errors, we can say that they exhibit a trend of anywhere between linear to exponential growth. Hence we desire to model $\boldsymbol{\eta}(t)$ to be an increasing function of time. Since the ME evolution is given by the following equation,

$$\boldsymbol{\eta}(t_0) = \boldsymbol{\eta}_0, \frac{d\boldsymbol{\eta}}{dt} = \Phi[\boldsymbol{\eta}(t)],$$

the rate of increase of $\boldsymbol{\eta}(t)$ in time is given by the particular form of $\Phi[\boldsymbol{\eta}]$. If $\Phi[\boldsymbol{\eta}] < 0, \forall \boldsymbol{\eta}$ then the ME decreases in time, if $\Phi[\boldsymbol{\eta}] = 0, \forall \boldsymbol{\eta}$ then ME is constant in time, and if $\Phi[\boldsymbol{\eta}] > 0, \forall \boldsymbol{\eta}$ then ME increases in time. We considered all these possibilities and investigated the following three forms of Φ,

1. Decreasing ME, $\Phi[\boldsymbol{\eta}] = -\beta\boldsymbol{\eta}$,
2. Constant ME, $\Phi[\boldsymbol{\eta}] = 0$,
3. Exponentially increasing ME, $\Phi[\boldsymbol{\eta}] = \gamma\boldsymbol{\eta}$,

where β and γ are constants, and in our numerical results we specified $\beta = 0.2/\Delta t$, and $\gamma = 0.01/\Delta t$. We used the same termination criteria as in Eq.(4.7),

$$\|(\nabla \mathcal{J})_k\| \leq EPS \cdot MAX[1, (\|\mathbf{x}_k(t_0)\| + \|\boldsymbol{\eta}_k(t_0)\|)],$$

where the gradient now includes the model error information, $\nabla \mathcal{J} = (\nabla_{\mathbf{x}_0} \mathcal{J}, \nabla_{\boldsymbol{\eta}_0} \mathcal{J})^T$, and $\boldsymbol{\eta}_k(t_0)$ is the optimal initial state vector at the $k^{th.}$ minimization iteration. Note that since the size of the gradient vector has doubled, the size of the Hessian of the cost functional is increased by four times. Due to the monotonicity criteria, increasing the size of the Hessian increases its condition number, which implies that a larger number of minimization iterations would have to be performed to achieve the same termination criteria (the same value of EPS was used in both strong and weak `VDA` for comparison sake). In fig. 4.5 we provide a comparison of the RMSE in geopotential height field after data assimilation using above forms of the ME for the unconstrained, constrained van Leer and PPM schemes (though

not shown, the RMSE in wind fields was significantly reduced using different forms of ME). Clearly inclusion of ME provides further reduction in forecast errors. As we said earlier, the model errors usually increase in time, therefore as expected the results obtained using the decreasing form of ME are inferior to those obtained with the other two forms. This conclusion is further substantiated by the fact that since the PPM scheme is least dissipative (compared to the van Leer schemes) the decreasing form of ME yields the same results as given by the strong constraint `VDA`. Though the constant form of ME is very simple, the results obtained using the van Leer schemes indicate that it yields results comparable to the increasing form. Noteworthy is the fact that in the case of PPM scheme, the increasing form is the best. To indicate that the inclusion of ME positively impacts DA via efficient preconditioning and weak constraint `VDA`, we provide a plot of $\mathcal{J}_o$ versus the number of cost functional evaluations using the constant ME form and the unconstrained van Leer scheme in fig. 4.6. Note the markedly improved fit of model states and observations using even the simplest form of constant ME. Figures 4.7(a)- 4.9(a) show the differences in the geopotential height field between model forecast and observation at forecast verification time, T_{+30} for various forms of model error using the PPM advection scheme (though not shown, the van Leer schemes yielded similar results). Note the improvement in forecast using the ME, when compared to the strong constraint `VDA` (i.e., with no ME), fig. 4.4. In figures 4.7(b)-4.9(b) we show isolines of different initial model error state corresponding to the geopotential height field for the different forms of ME and PPM advection scheme. The decreasing form of ME is very dispersed when compared to the localized nature of the other two forms. As expected the obtained initial ME state for the constant and increasing forms with the PPM scheme is even more localized than that obtained with the van Leer schemes, indicating the lack of dissipative effect with the PPM scheme.

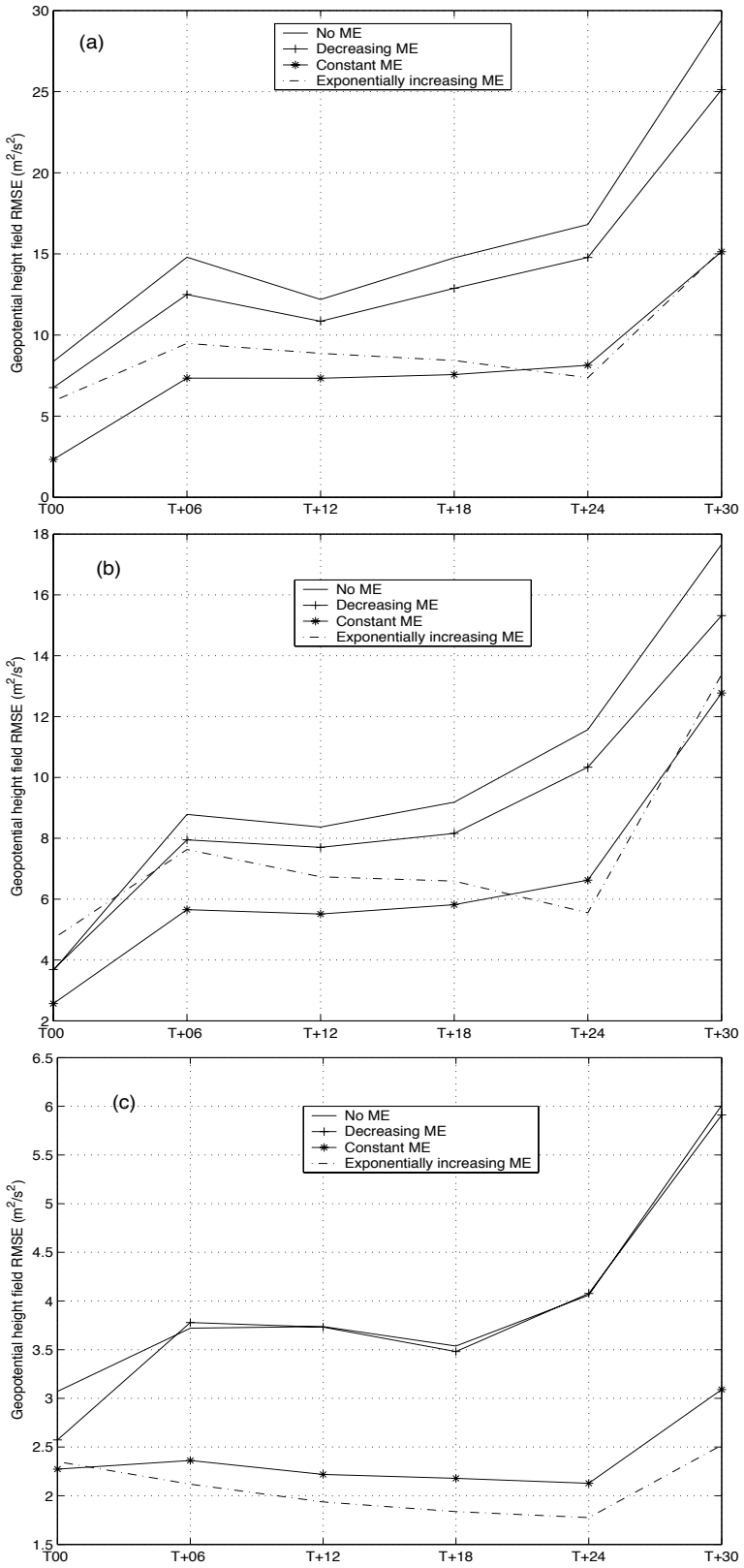

Figure 4.5: RMSE in the geopotential height fields for different advection schemes with different forms of model error (a) Unconstrained Van Leer (b), Constrained van Leer (c), PPM schemes

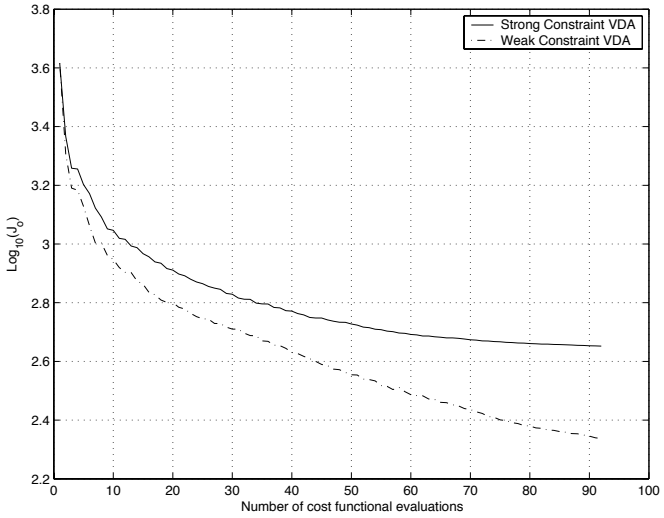

Figure 4.6: Observational component variation using constant ME, with unconstrained van Leer, weak and strong constraint `VDA`

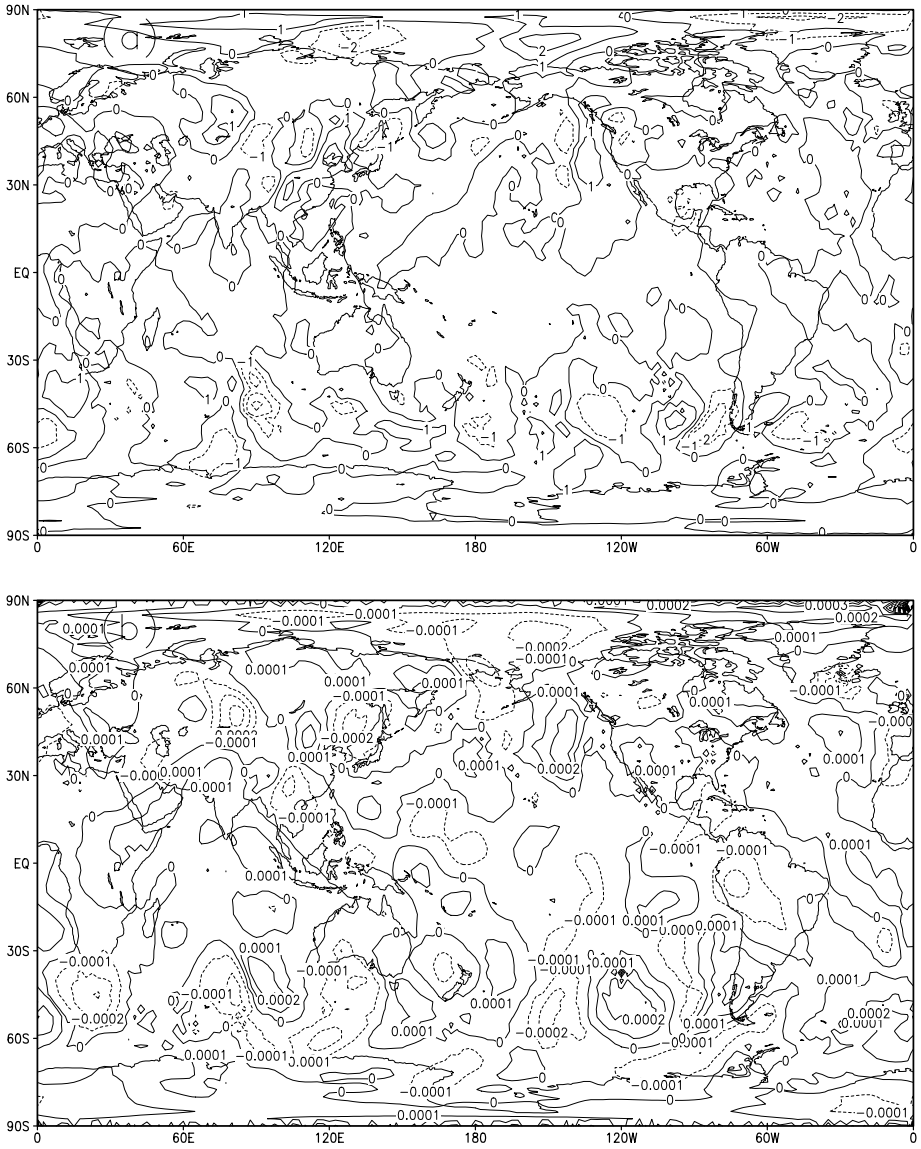

Figure 4.7: (a) Isolines of differences in geopotential height field between model forecast using optimized initial conditions obtained after weak constraint `VDA`(decreasing in time form of ME) and the PPM advection scheme, (b) initial model error state corresponding to the geopotential height field

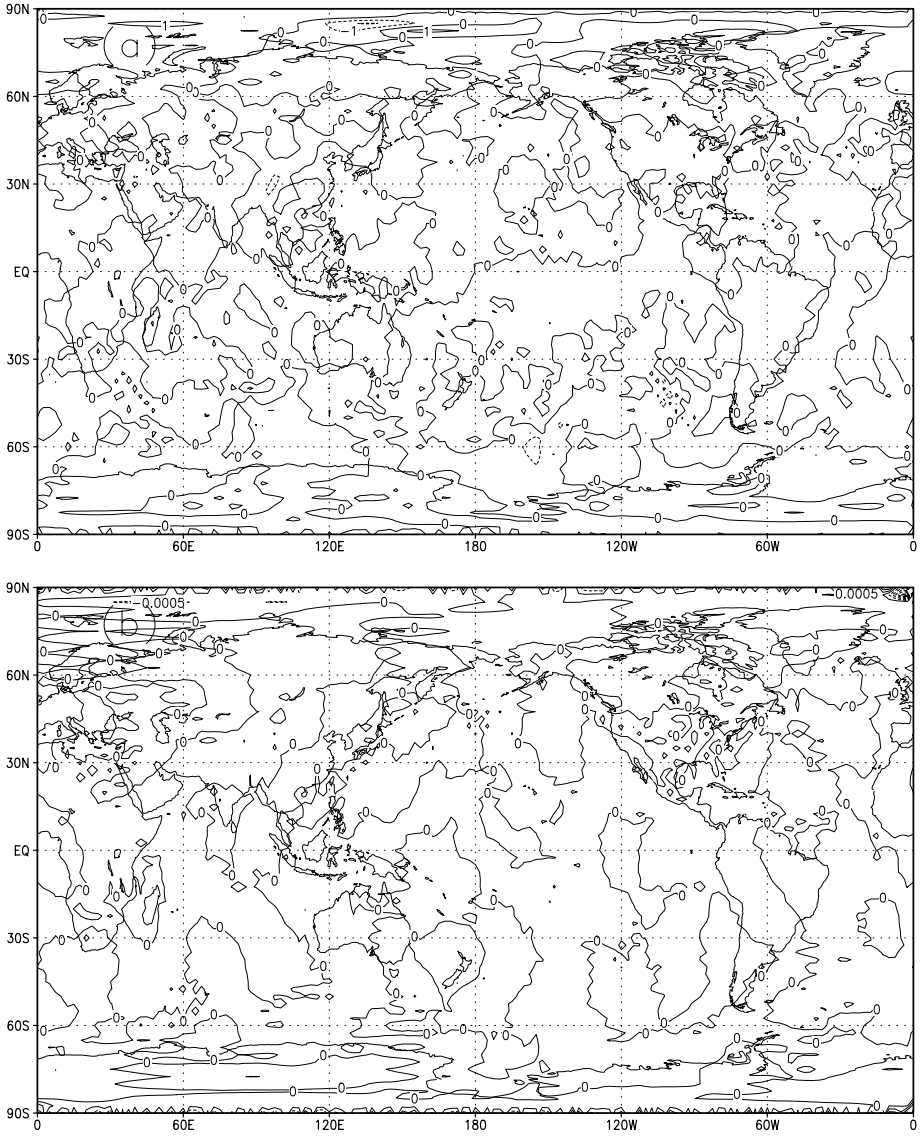

Figure 4.8: (a) Isolines of differences in geopotential height field between model forecast using optimized initial conditions obtained after weak constraint `VDA`(constant ME) and the PPM advection scheme, (b) initial model error state corresponding to the geopotential height field

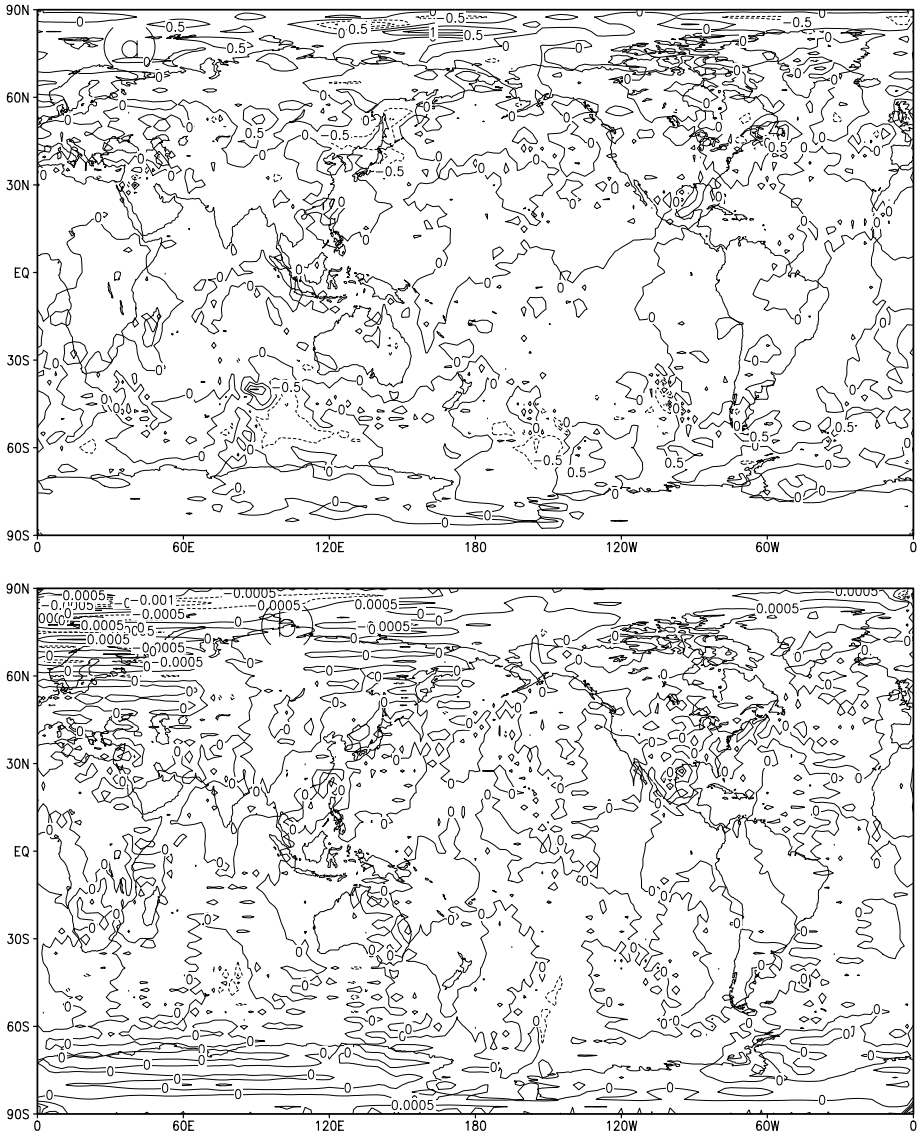

Figure 4.9: (a) Isolines of differences in geopotential height field between model forecast using optimized initial conditions obtained after weak constraint `VDA`(increasing in time form of ME) and the PPM advection scheme, (b) initial model error state corresponding to the geopotential height field

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

Variational data assimilation methods provide us the necessary tools to optimally estimate the state of the atmosphere and its evolution in space and time using all of the available information. We considered the formulation of a few VDA methods, such as 3D-var, PSAS and 4D-Var, which is the state of the art VDA method that can be used to assimilate observations realistically, as they occur in space and time. In this dissertation we have presented the formulation of various high resolution TVD, FV (which use MUSCL slope limiters and PPM) schemes. In chapter 3 we showed that a consistent change of just the advection scheme yielded significant change in the DA experimental results; these results are novel in the sense that so far the impact on VDA comparing the performance such advection schemes, in particular the PPM was not studied. Two specific nonlinear model problems, namely the viscous Burgers equation in one space dimension and the global SW equations model in two space dimensions have been used for conducting numerical experiments, as a first step towards addressing the same issues which will arise in the practical implementation of higher dimensional (3-D) complex NWP models for DA (such as using the FV-GCM for 4D-Var DA). In both cases that were studied, smooth solutions have been considered. 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.

Motivated by the results obtained in chapter 3, we studied one of the approaches to improve the results obtained from (classical or strong constraint) 4D-Var VDA, once again using various high resolution advection schemes (such as unconstrained, constrained van Leer and the PPM). In chapter 4 we started out by investigating different forecasting errors in

a twin experiment framework. Two principal causes for forecast errors being the erroneous specification of model initial conditions and modeling errors. Various strong constraint VDA (which does not include model error) experimental results obtained using different advection schemes indicate that a mere change in the advection scheme alone provides more than 50% reduction in forecast RMSE. Next we studied in depth the nature of modeling errors and suggested a decreasing, constant and increasing in time forms of ME. Implementation of these forms in a weak constraint VDA framework yielded a further reduction in forecast errors. If it is a-priori known (perhaps through forecasts or numerical analysis using 1-D cases) that dissipative schemes were used for weak VDA, then even the simplest forms of ME, such as a constant in time form provides significantly better results. For highly accurate advection schemes such as the PPM scheme, the increasing form of ME turns out to be the best (when tested in the framework of a twin experiment). As such, based on the results obtained from our preliminary investigation of impact of various forms of ME in the context of weak VDA, the increasing form of model error is a good candidate for further research on this topic. To sum up, three different forms of ME using high resolution advection schemes in the presence of non-linear advection terms were studied in both strong and weak constraint VDA framework.

A discussion of related topics of future research is provided in what follows. For the sake of simplicity in the implementation of weak VDA, we selected forms of ME which were independent of the model state, i.e, $\Phi[\boldsymbol{\eta}, \mathbf{x}] = \Phi[\boldsymbol{\eta}]$. However at the expense of extra computational work and developmental challenges, one could consider the TLM to be Φ, such a case was studied in KF framework by Dee and Da Silva (1998) [128]. Recall that we used the first order optimality criteria to derive the adjoint equations which provided us the means to obtain the gradient of the cost functional with respect to the control variables. But if the TLM is used for modeling the evolution of ME, i.e, $\Phi = \frac{\partial \mathcal{M}}{\partial \mathbf{x}}$ then one also needs to specify the action of $[\frac{\partial \Phi}{\partial \mathbf{x}}]^T$ and $[\frac{\partial \Phi}{\partial \boldsymbol{\eta}}]^T$ on $\boldsymbol{\eta}^*$ (the adjoint state corresponding to model error variables) which amounts to using second order information, see Le Dimet et al. (2002) [39]. The complexity of such a second order adjoint model certainly depends on the complexity of the NWP model as well as the equation used for the evolution of ME. The second order optimality criteria provide the necessary and sufficient conditions for extrema of the cost functional, whereas first order criteria are only necessary but not sufficient. Also availability of the Hessian (of the cost functional) information, via Hessian/vector product obtained from

the second order adjoint model, speeds up the minimization process, since implementation of the Newton methods is now possible.

Another topic of further research addressing improvement of the models used for modeling the evolution of ME can be derived by using the additive property in both weak and strong constraint versions of `VDA` (see Li and Navon 2001[41]). We can separate the observations into a few subsets and perform `VDA` for each subset (see Jarvinnen *et al.* 1996[156]). Same can be done with the model error, provided errors are uncorrelated in time, allowing model error to adjust within a smaller time window. Hence, this could be beneficial for better estimation of model errors. This is referred to as the property of consistent optimality by Li and Navon (2001).

If the the stochastic component of ME, $\mathbf{q}(t)$ is not negligible, then it is to be included in the modeling of ME, thereby the evolution of ME becomes stochastic as well. Earlier we modeled the evolution of ME as,

$$\frac{d\boldsymbol{\eta}}{dt} = \Phi[\boldsymbol{\eta}(t), \mathbf{x}(t)] + \mathbf{q}(t).$$

The above ordinary differential equation should be rewritten as a stochastic differential equation (SDE), such that $\eta(t) = \eta_t$ is an Itô process given by,

$$d\boldsymbol{\eta}_t = \Phi[\boldsymbol{\eta}_t, \mathbf{x}_t]dt + d\mathbf{q}_t, \tag{5.1}$$

and is a solution of the following integral equation,

$$\boldsymbol{\eta}_t = \boldsymbol{\eta}_0 + \int_{t_0}^{t_n} \Phi[\boldsymbol{\eta}_s, \mathbf{x}_s]ds + \int_{t_0}^{t_n} dq_s, \tag{5.2}$$

where we assumed that $\mathbf{x}(t)$ is deterministic, $\boldsymbol{\eta}_0$ is the initial value of the model error term, and $[t_0, t_n]$ is the time window of data assimilation. The simplest possible model for the stochastic term $\mathbf{q}_t$, is the Brownian motion, B_t, which has continuous paths. Whereas the white Gaussian noise process W_t (integral of white noise process results in Brownian motion) does not have continuous paths and has been used in various research work thus-far for data assimilation, variational and sequential. The problem of optimally controlling the stochastic model error would require formulation of optimal control within the SDE framework which will be addressed in future research.

Our current research focus is on further investigation of validity of the above findings for a higher dimensional system with real observations. Due to the fact that higher dimensional

systems significantly stretch the limits of the available computational resources, various methods for model reduction, such as those using properly orthogonal decomposition have been suggested [157]. The application of such model reduction techniques to `VDA` is presently being conducted by the author (in collaboration with Drs. I. M. Navon and D. N. Daescu).

APPENDIX A

A SIMPLE FLOWCHART ILLUSTRATING THE DATA ASSIMILATION CYCLE IMPLEMENTED AT WEATHER FORECASTING CENTERS

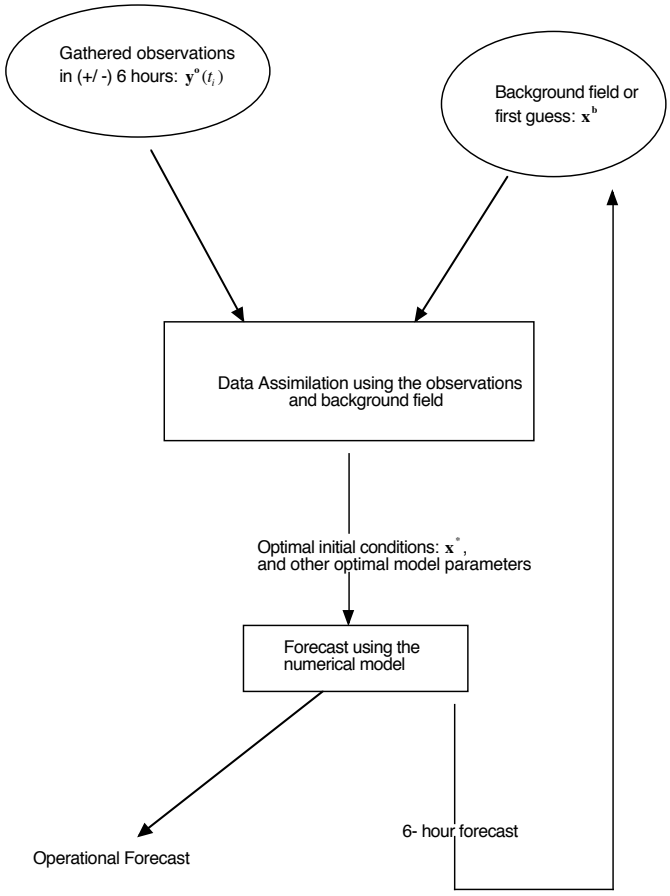

Figure A.1: An illustration of the variational data assimilation cycle, which is periodically carried out at all the major weather forecasting centers to generate operational forecasts.

APPENDIX B

A SEGMENT OF TANGENT LINEAR AND ADJOINT MODEL CODES, WHEN HIGH ORDER ADVECTION SCHEMES ARE INVOLVED IN NONLINEAR MODEL

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 following,

```
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 [158, 159] for discussion on the trade-off between storing in memory and recomputation.

APPENDIX C

DESCRIPTION OF THE BACKGROUND ERROR COVARIANCE OPERATOR

Following Eq. (4.5) the strong constraint `VDA` cost functional is given by,

$$\mathcal{J} = \mathcal{J}_b + \mathcal{J}_o.$$

A static-in-time $\mathbf{B}$ is constructed in the grid point space as an operator, which is based on the formulation provided by Weaver and Courtier (2001)[154]; Derber and Bouttier (1999)[152].

Let $\delta\mathbf{x} = \mathbf{x}(t_0) - \mathbf{x}^b$, and define a transformation, $\mathbf{v} = \mathbf{B}^{-1/2}\,\delta\mathbf{x}$, which implies that $\delta\mathbf{x} = \mathbf{B}^{1/2}\mathbf{v}$. Where the $\mathbf{B}^{1/2}$ is taken to be any square-root matrix, such that $\mathbf{B} = \mathbf{B}^{1/2}\,\mathbf{B}^{T/2}$; $\mathbf{B}^{T/2}$ denotes the transpose of $\mathbf{B}^{1/2}$. Therefore the background cost functional can be rewritten as,

$$\mathcal{J}_b = \frac{1}{2}\delta\mathbf{x}^T\,\mathbf{B}^{-1}\,\delta\mathbf{x} = \frac{1}{2}\delta\mathbf{x}^T\,(\mathbf{B}^{1/2}\,\mathbf{B}^{T/2})^{-1}\,\delta\mathbf{x} = \frac{1}{2}\mathbf{v}^T\,\mathbf{v}.$$

Hence the contribution to the gradient of the cost functional, $\mathcal{J}$ from the background cost functional is equal to $\nabla_{\mathbf{v}}\,\mathcal{J}_b = \mathbf{v}$, and to the Hessian of the cost functional, $\nabla^2{}_{\mathbf{v}}\,\mathcal{J}_b = I$. At the beginning of the minimization, $\mathbf{v} = \delta\mathbf{x} = 0$, such that the initial guess for $\mathbf{x}(t_0)$ is $\mathbf{x}^b$. This transformation of variables preconditions the minimization problem for faster convergence of the minimization algorithm. An ideal preconditioning is obtained if the Hessian matrix is an identity matrix. A good approximation to this is to ensure that the Hessian of $\mathcal{J}_b$ is equal to I, which is indeed the case here, since the minimization is performed in the $\mathbf{v}$ space. To summarize,

$$\mathcal{J} = \mathcal{J}_b + \mathcal{J}_o = \frac{1}{2}[\mathbf{x}(t_0) - \mathbf{x}^b]^T\mathbf{B}^{-1}[\mathbf{x}(t_0) - \mathbf{x}^b] + \mathcal{J}_o = \frac{1}{2}\mathbf{v}^T\,\mathbf{v} + \mathcal{J}_o,$$

where $\delta\mathbf{x} = \mathbf{x}(t_0) - \mathbf{x}^b$, and $\mathbf{v} = \mathbf{B}^{-1/2}\,\delta\mathbf{x}$, which implies $\delta\mathbf{x} = \mathbf{B}^{1/2}\mathbf{v}$. Therefore gradient of

the cost functional with respect to $\mathbf{v}$ is given by,

$$\nabla_{\mathbf{v}} \mathcal{J} = \mathbf{v} + \nabla_{\mathbf{v}} \mathcal{J}_o = \mathbf{v} + \mathbf{B}^{T/2} \nabla_{\mathbf{x}_0} \mathcal{J}_o.$$

Thus every minimization iteration requires application of $\mathbf{B}^{1/2}$ to obtain the analysis increment $\delta\mathbf{x}$ from $\mathbf{v}$ and $\mathbf{B}^{T/2}$ to get the gradient $\nabla_{\mathbf{v}} \mathcal{J}_o$ from $\nabla_{\mathbf{x}_0} \mathcal{J}_o$ (which is computed by a single integration of the adjoint model backward in time). As evident, we do not require inverse of $\mathbf{B}$ in the above formulation.

The model variables $(\mathbf{h}, \mathbf{u}, \mathbf{v})$ are partitioned into balanced and unbalanced components. The so-called balancing operator, $\mathbf{K}_b$ acts on the unbalanced components of the model variables and in-turn, $\mathbf{K}_b = \mathbf{K'}_b + I$. Following Vidard *et al.* (2004)[144], $\mathbf{K'}_b$ is formulated using the linear balance equations, based on geostrophic balance (written in spherical coordinates) and hydrostatic hypothesis.

Geostrophic balance:

$$\begin{aligned} u &= -\frac{1}{\rho f}\left[\frac{1}{a}\frac{\partial p}{\partial \theta}\right], \\ v &= \frac{1}{\rho f}\left[\frac{1}{a\cos\theta}\frac{\partial p}{\partial \lambda}\right]. \end{aligned}$$

Hydrostatic hypothesis: $p = \rho\, g\, h$.

Which implies,

$$\begin{aligned} u &= -\frac{g}{f}\left[\frac{1}{a}\frac{\partial h}{\partial \theta}\right], \\ v &= \frac{g}{f}\left[\frac{1}{a\cos\theta}\frac{\partial h}{\partial \lambda}\right]. \end{aligned}$$

Therefore

$$\mathbf{K}_b = \mathbf{K'}_b + I = \begin{pmatrix} I & 0 & 0 \\ -\frac{g}{a\,f}\frac{\partial}{\partial\theta} & I & 0 \\ \frac{g}{a\,f\,\cos\theta}\frac{\partial}{\partial\lambda} & 0 & I \end{pmatrix}$$

which is a lower triangular matrix, since our control vector is of the form $(\mathbf{h},\ \mathbf{u},\ \mathbf{v})^T$.

Remark: At the North and South poles, one sided differences have been used for computing the above derivative with respect to the latitude and at the equator, where $\theta = \pi/2$, we have used the average values of the derivative (with respect to the longitude) from the two neighboring latitude circles, above and below the equator.

Using the balance operator, we can write $\mathbf{B} = \mathbf{K}_b \, \mathbf{B}_u \, \mathbf{K}_b^T$, where $\mathbf{B}_u$ is a block diagonal error covariance matrix for the unbalanced component of the variables (see [154]), which implies that the cross-covariances between the unbalanced variables is taken to be negligible. Thus $\mathbf{B}_u = \mathbf{\Sigma_b} \, \mathbf{C} \, \mathbf{\Sigma_b}$, where $\mathbf{\Sigma_b}$ is a block-diagonal matrix of the background-error variances in the grid point space, such that the diagonal entries represent error variances at every grid point (in this work, we prescribed $\Sigma_b = [2000\, I, 100\, I, 100\, I]$).

$\mathbf{C}$ is a symmetric matrix of background-error correlations for the unbalanced component of the variables. Assuming that $\mathbf{C}$ is block-diagonal, which is a valid assumption, since $\mathbf{B}_u$ has already been assumed to be block-diagonal, we obtain the square-root factorization $\mathbf{C} = \mathbf{C}^{1/2} \, \mathbf{C}^{T/2}$.

Thus the square-root factorization of the background error covariance can be written as,

$$\begin{aligned} \mathbf{B} &= \mathbf{K}_b \, \mathbf{B}_u \, \mathbf{K}_b^T = \mathbf{K}_b \, (\mathbf{\Sigma_b} \, \mathbf{C} \, \mathbf{\Sigma_b}) \, \mathbf{K_b^T} = \mathbf{K_b} \, (\mathbf{\Sigma_b} \, \mathbf{C^{1/2}} \, \mathbf{C^{T/2}} \mathbf{\Sigma_b}) \, \mathbf{K_b^T} \\ &= (\mathbf{K}_b \, \mathbf{\Sigma_b} \, \mathbf{C^{1/2}}) \, (\mathbf{C^{T/2}} \mathbf{\Sigma_b} \, \mathbf{K_b^T}) \\ &= \mathbf{B}^{1/2} \, \mathbf{B}^{T/2}. \end{aligned} \tag{C.1}$$

Notice that the above formulation ensures that $\mathbf{B}$ is symmetric and positive definite, both of these properties are usually required to be satisfied by any preconditioning matrix. The analysis increment is given by $\delta\mathbf{x} = \mathbf{B}^{1/2}\mathbf{v} = \mathbf{K}_b \, \mathbf{\Sigma_b} \, \mathbf{C^{1/2}}\mathbf{v}$. Since $\mathbf{C}$ is block-diagonal, the operation $\mathbf{C}^{1/2} \, \mathbf{v}$ can be split into individual operators $\mathbf{C}_\alpha^{1/2} \, \mathbf{v}_\alpha$, that act independently on different components of the variable $\mathbf{v}$, such as $\mathbf{v}_\alpha$. For each variable, the univariate operator can be factorized into $\mathbf{C}_\alpha = \mathbf{C}_\alpha^{1/2} \, \mathbf{C}_\alpha^{T/2}$. The procedure suggested by Weaver and Courtier (2001) [154] has been implemented to model the univariate correlation operator, has been implemented to model the univariate correlation operator, $\mathbf{C}_\alpha$ as an isotropic diffusion operator, assuming Gaussianity with a decorrelation length equal to $500\, km$.

We considered height field which was comprised of a single Dirac delta pulse located at equator and longitude 180^o, and prescribed no wind field, the action of $\mathbf{B}$ on such a field is shown in Fig. C.1 (a). We see the effect of the correlation operator on the Dirac pulse and also on the wind field obtained under geostrophic balance assumption (Fig. C.1 (b)), which is parallel to the isobars of the pressure. Since there is a *high pressure* at the center, the direction of the wind is clockwise in the Northern hemisphere and anti-clockwise in the Southern hemisphere; at the equator due to the balancing of the pressure gradient and Coriolis forces, the wind blows straight.

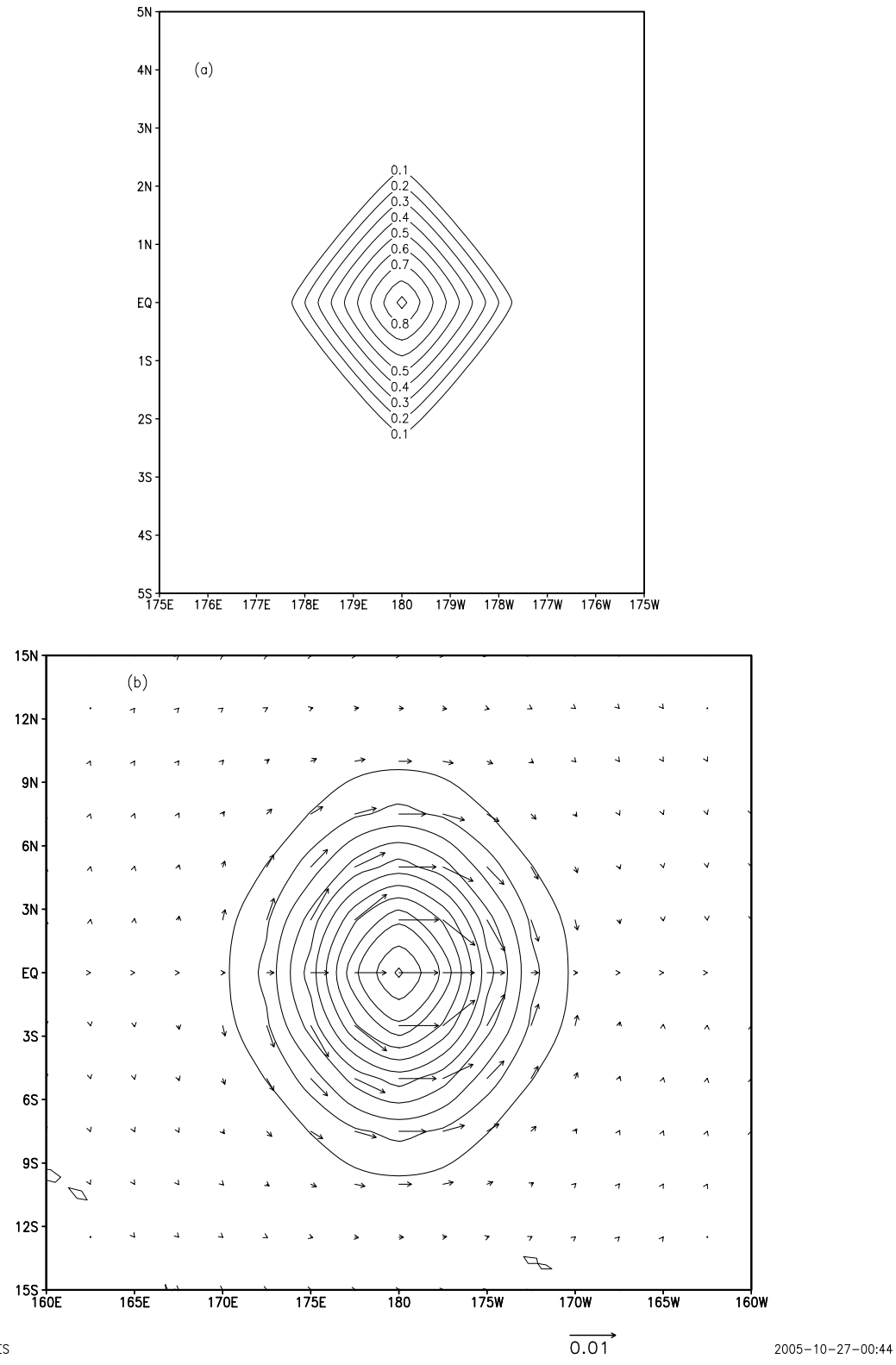

Figure C.1: Result obtained by operating with **B** on a single Dirac delta pulse in the height field (a), isolines of the height field (b), geostrophic wind plotted along with the isolines of the height field.

APPENDIX D

SQUARE-ROOT TRANSFORMATION USED FOR B AND Q

We have extended the procedure described in the previous Appendix A to include model error control vector, thereby we achieve all the preconditioning properties for the minimization process we discussed earlier (this procedure has also been described by Vidard *et. al.* 2004 [144]). This approach circumvents the need to specify inverses of the background and model error covariance matrices via a transformation which is similar to the one described earlier for the strong constraint `VDA` which included only $\mathcal{J}_b$ and $\mathcal{J}_o$ in the cost functional. Let

$$\mathbf{z} = \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\eta} \end{bmatrix},$$

such that

$$\delta\mathbf{z} = \begin{bmatrix} \mathbf{x}_0 - \mathbf{x}^b \\ \boldsymbol{\eta}_0 - \boldsymbol{\eta}^b \end{bmatrix}.$$

Recall that the sum of background and model error cost functionals ($\mathcal{J}_b$ and $\mathcal{J}_{\boldsymbol{\eta}}$, respectively) is given by,

$$\begin{aligned} \mathcal{J}_b + \mathcal{J}_{\boldsymbol{\eta}} &= \frac{1}{2}[\mathbf{x}_0 - \mathbf{x}^b]^T \mathbf{B}^{-1} [\mathbf{x}_0 - \mathbf{x}^b] + [\boldsymbol{\eta}_0 - \boldsymbol{\eta}^b]^T \mathbf{Q}^{-1} [\boldsymbol{\eta}_0 - \boldsymbol{\eta}^b] \\ &= \frac{1}{2}\delta\mathbf{z}^T \begin{bmatrix} \mathbf{B}^{-1} & 0 \\ 0 & \mathbf{Q}^{-1} \end{bmatrix} \delta\mathbf{z} \end{aligned} \tag{D.1}$$

As in appendix A, let $\mathbf{B}^{1/2}$ and $\mathbf{Q}^{1/2}$ be any square-root matrices such that $\mathbf{B} = \mathbf{B}^{1/2}\mathbf{B}^{T/2}, \mathbf{Q} = \mathbf{Q}^{1/2}\mathbf{Q}^{T/2}$, and let

$$\begin{aligned} \mathbf{w} &= \begin{bmatrix} \mathbf{B}^{-1/2} & 0 \\ 0 & \mathbf{Q}^{-1/2} \end{bmatrix} \delta\mathbf{z} \\ \Rightarrow \delta\mathbf{z} &= \begin{bmatrix} \mathbf{B}^{1/2} & 0 \\ 0 & \mathbf{Q}^{1/2} \end{bmatrix} \mathbf{w}. \end{aligned} \tag{D.2}$$

The above transformation from $(\mathbf{x}, \boldsymbol{\eta}) \rightarrow \mathbf{w}$ is similar to the previously described transformation: $\mathbf{x} \rightarrow \mathbf{v}$, which involved $\mathbf{B}^{1/2}$ only. Using the above equations (D.1) and (D.2),

$$\mathcal{J}_b + \mathcal{J}_{\boldsymbol{\eta}} = \frac{1}{2}\mathbf{w}^T\mathbf{w},$$

hence the entire cost functional,

$$\begin{aligned} \mathcal{J} &= \mathcal{J}_b + \mathcal{J}_{\boldsymbol{\eta}} + \mathcal{J}_o \\ &= \frac{1}{2}\mathbf{w}^T\mathbf{w} + \mathcal{J}_o, \end{aligned} \tag{D.3}$$

and the gradient of the cost functional with respect to the transformed variable, $\mathbf{w}$ is given by

$$\begin{aligned} \nabla_{\mathbf{w}}\mathcal{J} &= \mathbf{w} + \nabla_{\mathbf{w}}\mathcal{J}_o \\ &= \mathbf{w} + \begin{bmatrix} \mathbf{B}^{T/2} & 0 \\ 0 & \mathbf{Q}^{T/2} \end{bmatrix} \begin{pmatrix} \nabla_{\mathbf{x}_0}\mathcal{J}_o \\ \nabla_{\boldsymbol{\eta}_0}\mathcal{J}_o \end{pmatrix}. \end{aligned} \tag{D.4}$$

We have modeled $\mathbf{Q}$ as a block diagonal matrix,

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_{hh} & 0 & 0 \\ 0 & \mathbf{Q}_{uu} & 0 \\ 0 & 0 & \mathbf{Q}_{vv} \end{bmatrix},$$

such that each of the blocks is an univariate Gaussian correlation operator and has a square-root decomposition given by,

$$\mathbf{Q}_{\alpha\alpha} = \boldsymbol{\Sigma}_{\mathbf{q}}\mathbf{C}_\alpha\boldsymbol{\Sigma}_{\mathbf{q}} = \boldsymbol{\Sigma}_{\mathbf{q}}\mathbf{C}_\alpha{}^{\mathbf{1/2}}\mathbf{C}_\alpha{}^{\mathbf{T/2}}\boldsymbol{\Sigma}_{\mathbf{q}} = \left(\boldsymbol{\Sigma}_{\mathbf{q}}\mathbf{C}_\alpha{}^{\mathbf{1/2}}\right)\left(\boldsymbol{\Sigma}_{\mathbf{q}}\mathbf{C}_\alpha{}^{\mathbf{1/2}}\right)^{\mathbf{T}} = \mathbf{Q}_{\alpha\alpha}^{\mathbf{1/2}}\mathbf{Q}_{\alpha\alpha}^{\mathbf{T/2}},$$

where $\alpha = \mathbf{h}, \mathbf{u}, \mathbf{v}$; $\boldsymbol{\Sigma}_{\mathbf{q}}$ is a diagonal matrix of variances (we prescribed $\boldsymbol{\Sigma}_{\mathbf{q}} = \mathbf{10}^{-\mathbf{2}}\boldsymbol{\Sigma}_{\mathbf{b}}$) and $\mathbf{C}_\alpha$ is an isotropic diffusion operator, construction of which was described in the previous appendix. The most simplistic model error covariance matrix is a diagonal matrix, which

implies that the analyzed model error increment at any specific grid point does have any influence on the increments at the neighboring grid points. Alternatively a diffusion operator provides such an increment in a localized region (given by the length scale of diffusion) inexpensively. We have not used the balance operator, $\mathbf{K}_b$ in the above construction of $\mathbf{Q}$. The goestrophic balance and hydrostatic hypothesis which were used in the construction of $\mathbf{K}_b$ are not required for $\mathbf{Q}$ since we do not have the same information about model error covariances as we have for background error covariances; therefore usage of $\mathbf{K}_b$ in specification of $\mathbf{Q}$ will only involve extra computational work.

At the beginning of minimization, the initial guess for $\mathbf{x}_0 = \mathbf{x}^b$ and $\boldsymbol{\eta}_0 = \boldsymbol{\eta}^b$, therefore $\mathbf{w} = \delta\mathbf{z} = 0$. Every minimization iteration (carried out in $\mathbf{w}$ space) requires application of $\mathbf{B}^{1/2}, \mathbf{Q}^{1/2}$ to obtain the analysis increment $\delta\mathbf{z}$ from $\mathbf{w}$ (Eq. (D.2)) and the adjoint operators, $\mathbf{B}^{T/2}, \mathbf{Q}^{T/2}$ to get the gradient $\nabla_{\mathbf{w}}\mathcal{J}_o$ from $(\nabla_{\mathbf{x}_0}\mathcal{J}_o, \nabla_{\boldsymbol{\eta}_0}\mathcal{J}_o)^T$ (Eq. (D.4)).

REFERENCES

[1] E. Kalnay. *Atmospheric Modeling, Data Assimilation and Predictability*. Cambridge University Press, 2002. 1, 2.2.3, 2.2.5, 2.2.5, 3.3, 4.2

[2] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME- Journal of Basic Engineering*, 82:35–45, 1960. 1

[3] R. E. Kalman and R. S. Bucy. New results in linear filtering and prediction theory. *Transactions of the ASME- Journal of Basic Engineering*, 83:95–107, 1961. 1

[4] M. Ghil, S. Cohn, J. Tavantzis, K. Bube, and E. Isaacson. *Dynamic Meteorology: Data Assimilation Methods*, chapter Applications of estimation theory to numerical weather prediction, pages 139–224. Springer-Verlag: New-York, 1981. 1

[5] R. Todling and M. Ghil. Comparisons between the kalman filter and optimal interpolation for two-dimensional atmospheres. In *Proc. Seventh Brazilian Meteorology Congress*, pages 660–664, Säo Paulo, Brazil, 1992. Brazilian Meteorology Society. 1

[6] R. Todling. Estimation theory and foundations of atmospheric data assimilation. Technical Report DAO Office Note 1999-01, Goddard Space Flight Center, Greenbelt, Maryland-20771, 1999. 1

[7] M. Ghil. Advances in sequential estimation for atmospheric and oceanic flows. *Journal of the Meteorological Society of Japan*, 75:289–304, 1997. 1

[8] C. Wunsch. *The Ocean circulation inverse problems*. Cambridge University Press, 1996. 1

[9] L. Yu and J. J. O'Brien. Variational data assimilation for determining the seasonal net surface heat flux using a tropical pacific ocean model. *Journal of Physical Oceanography*, 25:2319–2343, 1995. 1

[10] I. M. Navon, X. Zou, J. Derber, and J. Sela. Variational data assimilation with an adiabatic version of the NMC spectral model. *Monthly Weather Review*, 120:1433–1446, 1992. 1, 3.3, 3.3, 4, 4.2

[11] F.-X. LeDimet and O. Talagrand. Variational algorithms for analysis and assimilation of meteorological observations- theoretical aspects. *Tellus Series A-Dynamic Meteorology and Oceanography*, 38:97–110, 1986. 1, 4

[12] A. Lorenc. Analysis methods for numerical weather prediction. *Q. J. R. Meteorological Society*, 112:1177–1194, 1986. 1, 2.2.3, 4.2

[13] B. Mohammadi and O. Pironneau. *Applied Shape Optimization for Fluids*. Oxford University Press, 2001. 1

[14] M. D. Gunzburger. *Perspectives in flow control and optimization*. SIAM: Philadelphia, 2003. 1, 3.3

[15] D. G. Cacuci. Sensitivity theory for nonlinear systems. I: Nonlinear functional analysis approach. *Journal of Mathematical Physics*, 22:2794–2802, 1981. 1

[16] D. G. Cacuci. Sensitivity theory for nonlinear systems. II: Extensions to additional classes of responses. *Journal of Mathematical Physics*, 22:2803–2812, 1981. 1

[17] D. G. Cacuci. *Sensitivity and uncertainty analysis*. Chapman & Hall/CRC, 2003. 1

[18] D. G. Cacuci, I-B. Mihaela, and I. M. Navon. *Sensitivity and uncertainty analysis, Volume II: Applications to Large-Scale Systems*. Chapman & Hall/CRC, 2005. 1

[19] World Meteorological Organization. Numerical methods used in atmospheric models. GARP Publication Series, Number 17, Volume II, 1979. 2.1.2

[20] P. Bergthorsson and B. R. Döös. Numerical weather map analysis. *Tellus Series A-Dynamic Meteorology and Oceanography*, 7:329–340, 1955. 2.2

[21] G. P. Cressman. An operational objective analysis system. *Monthly Weather Review*, 87:367–374, 1959. 2.2

[22] N. Gustafsson. A review of methods for objective analysis. In L. Bengtsson, M. Ghil, and E. Kallen, editors, *Dynamic Meteorology, Data Assimilation Methods*, page 325. Springer-Verlag: New-York, 1981. 2.2

[23] X. Zou, I. M. Navon, and F.-X. Le Dimet. An optimal nuding data assimilation scheme using parameter estimation. *Q. J. R. Meteorological Society*, 118:1163–1186, 1992. 2.2

[24] P. A. Vidard, F.-X. Le Dimet, and A. Piacentini. Determination of optimal nuding coefficients. *Tellus Series A-Dynamic Meteorology and Oceanography*, 55:1–15, 2003. 2.2

[25] L. Bengtssoson, M. Ghil, and E.Källén. *Dynamic Meteorology: Data Assimilation Methods*. Springer-Verlag: New-York, 1981. 2.2.1

[26] L. S. Gandin. *Objective Analysis of Meteorological Fields*. Israel Program for Scientific Translations, 1963. 2.2.1

[27] R. Daley. *Atmospheric Data Analysis*. Cambridge University Press, Cambridge, 1991. 2.2.1

[28] A. W. F. Edwards. *Likelihood*. Cambridge University Press, 1984. 2.2.2

[29] F. Vandenberghe and Y.-H. Kuo. Introduction to the MM5 3DVAR data assimilation system: Theoretical basis. Technical report, NCAR, 1999. 2.2.2

[30] P. Courtier, E. Andersson, and W. Heckley et al. The ECMWF implementation of three-dimensional variational assimilation (3D-Var). I: Formulation. *Q. J. R. Meteorological Society*, 124:1783–1807, 1998. 2.2.2

[31] E. Andersson, J. Haseler, and P. Undén et al. The ECMWF implementation of three-dimensional variational assimilation (3D-Var). III: Experimental results. *Q. J. R. Meteorological Society*, 124:1831–1860, 1998. 2.2.2

[32] G. Golub and C. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, third edition, 1996. 2.2.3

[33] P. Courtier. Dual formulation of four-dimensional variational assimilation. *Q. J. R. Meteorological Society*, 123:2249–2261, 1997. 2.2.3, 2.2.4

[34] P. Courtier. Variational methods. *Journal of the Meteorological Society of Japan*, 75:211–218, March 1997. 2.2.3, 2.2.5

[35] A. Da Silva, J. Pfaendtner, J. Guo, M. Sienkiewicz, and S. Cohn. Assessing the effects of data selection with DAO's physical-space statistics analysis system. In *Proceedings of the second international symposium on the assimilation of observations in meteorology and oceanography*, pages 13–17, Tokyo, Japan, 1995. World Meteorological Organization and Japan Meteorological Agency. 2.2.4

[36] A. da Silva and J. Guo. Documentation of the physical-space statistical analysis system (PSAS) Part I: The conjugate gradient solver version PSAS-1.00. Technical Report DAO Office Note 96-02, Data Assimilation Office, Goddard Space Flight Center, 1996. 2.2.4

[37] S. E. Cohn, A. da Silva, J. Guo, M. Sienkiewicz, and D. Lamich. Assessing the effects of data selection with DAO physical-space statistical analysis system. *Monthly Weather Review*, 126:2913–2926, 1998. 2.2.4

[38] F. Rabier, H. Jarvinen, E. Klinker, J. F. Mahfouf, and A. Simmons. The ECMWF operational implementation of four-dimensional variational assimilation. I: Experimental results with simplified physics. *Q. J. R. Meteorological Society*, 126:1148–1170. 2.2.5

[39] F.-X. LeDimet, I. M. Navon, and D. N. Daescu. Second-order information in data assimilation. *Monthly Weather Review*, 130:629–648, 2002. 2.2.5, 5

[40] Q. Xu. Comments on "Tangent linear and adjoint of "on-off" processes and their feasibility for use in 4-dimensional variavtional data assimilation. *Tellus Series A-Dynamic Meteorology and Oceanography*, 50:653–656, 1998. 2.2.5

[41] Z. Li and I. M. Navon. Optimality of variational data assimilation and its relationship with the Kalman filter and smoother. *Q. J. R. Meteorological Society*, 127:661–683, 2001. 2.2.5, 5

[42] A. F. Bennett. *Inverse Methods in Physical Oceanography*. Cambridge University Press, 1992. 2.2.5, 4

[43] http://www.autodiff.com/tamc. Tangent and adjoint model compiler (TAMC). 2.2.5, 3.3

[44] http://www unix.mcs.anl.gov/autodiff/. Automatic differentiation of fortran (ADIFOR). 2.2.5

[45] http://www sop.inria.fr/safir/SAM/Odyssee/odyssee.html. Odyssée. 2.2.5

[46] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer series in Operations Research. Springer, 1999. 2.3, 3.4, 3.5

[47] S. G. Nash and A. Sofer. *Linear and Nonlinear Programming*. McGraw-Hill series in Industrial Engineering and Management Science. McGraw-Hill, 1996. 2.3

[48] I. M. Navon and D. M. Legler. Conjugate-gradient methods for large-scale minimization in meteorology. *Monthly Weather Review*, 115:1479–1502, 1987. 2.3.2

[49] R. Fletcher and C. M. Reeves. Function minimization by conjugate gradients. *Computer Journal*, 7:149–154, 1964. 2.3.2

[50] E. Polak. *Optimization: Algorithms and Consistent Approximations*, volume 124 of *Applied Mathematical Sciences*. Springer, 1997. 2.3.2

[51] J. W. Thomas. *Numerical Partial Differential Equations: Finite Difference Methods*. Springer, 1998. 3, 3.1.1

[52] J. W. Thomas. *Numerical Partial Differential Equations : Conservation Laws and Elliptic Equations*. Springer, 1999. 3

[53] R. J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, 2002. 3, 3.1, 3.1.1

[54] O. C. Zienkiewicz and R. L. Taylor. *Finite Element Method: The Basis*, volume 1. Butterworth-Heinemann, 5th. edition, 2000. 3

[55] O. C. Zienkiewicz and R. L. Taylor. *Finite Element Method: Solid Mechanics*, volume 2. John Wiley & Sons, 5th. edition, 2000. 3

[56] O. C. Zienkiewicz and R. L. Taylor. *Finite Element Method: Fluid Mechanics*, volume 3. Butterworth-Heinemann, 5th. edition, 2000. 3

[57] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang. *Spectral Methods in Fluid Dynamics*. Springer, 3rd. edition, 1991. 3

[58] G. Karniadakis and S. J. Sherwin. *Spectral/hp Element Methods For Computational Fluid Dynamics (Numerical Mathematics and Scientific Computation)*. Oxford University Press, 2nd. edition, 2005. 3

[59] R. J. LeVeque. *Numerical Methods for Conservation Laws*. Birkhäuser, 1990. 3, 3.1

[60] C. B. Laney. *Computational Gasdynamics*. Cambridge University Press, 1998. 3, 3.1.1, 3.1.2

[61] D. R. Durran. *Numerical Methods for Wave Equations in Geophysical Fluid Dynamics*. Springer-Verlag: New-York, 1999. 3

[62] S. Zhang, X. Zou, J. Ahlquist, and I. M. Navon. Use of differentiable and nondifferentiable optimization algorithms for variational data assimilation with discontinuous cost functions. *Monthly Weather Review*, 128:4031–4044, 2000. 3

[63] R. B. Rood. Numerical advection algorithms and their role in atmospheric transport and chemistry models. *Reviews of Geophysics*, 25:71–100, 1987. 3

[64] S.-J. Lin, W. C. Chao, Y. C. Sud, and G. K. Walker. A class of the van Leer transport schemes and its applications to the moisture transport in a general circulation model. *Monthly Weather Review*, 122:1575–1593, 1994. 3, 3.1.1, 3.2.1, 3.2.2, 4.1, 4.1

[65] S.-J. Lin and R. B. Rood. Multidimensional flux-form semi-lagrangian transport schemes. *Monthly Weather Review*, 124:2046–2070, 1996. 3, 3.1.5, 3.2.2

[66] B. van Leer. Towards the ultimate conservative difference scheme. IV: A new approach to numerical convection. *Journal of Computational Physics*, 23:276–299, 1977. 3, 3.1, 4, 5

[67] S.-J. Lin. A "vertically lagrangian" finite-volume dynamical core for global models. *Monthly Weather Review*, 132:2293–2307, 2004. 3, 3.1.5, 3.2.2

[68] S.-J. Lin and R. B. Rood. An explicit flux-form semi- lagrangian shallow-water model on the sphere. *Q. J. R. Meteorological Society*, 123:2477–2498, 1997. 3, 3.1.5, 3.2.3, 4.1

[69] P. K. Smolarkiewicz. A fully multidimensional positive definite advection algorithm with small implicit diffusion. *Journal of Computational Physics*, 54:325–362, 1984. 3

[70] P. K. Smolarkiewicz and L. G. Margolin. MPDATA: A finite-difference solver for geophysical flows. *Journal of Computational Physics*, 140:459–480, 1998. 3

[71] B. P. Leonard. A stable and accurate convective modeling procedure based on quadratic upstream interpolation. *Computer Methods in Applied Mechanics and Engineering*, 19:59–98, 1979. 3

[72] B. P. Leonard. The ULTIMATE conservative difference scheme applied to unsteady one-dimensional advection. *Computer Methods in Applied Mechanics and Engineering*, 88:17–74, 1991. 3

[73] B. P. Leonard. Order of accuracy of QUICK and related convection-diffusion schemes. *Applied Mathematical Modeling*, 19:640–653, 1995. 3

[74] T. Vukićević, M. Steyskal, and M. Hecht. Properties of advection algorithms in the context of variational data assimilation. *Monthly Weather Review*, 129:1221–1231, 2001. 3, 3.5

[75] http://hycom.rsmas.miami.edu. Hycom consortium for data-assimilative ocean modeling. 3

[76] Z. Sokol. Comparison of several numerical schemes applied to advection equations. *Q. J. R. Meteorological Society*, 125:213–224, 1999. 3

[77] M. Iskandarani, J. C. Levin, B.-J. Choi, and D. B. Haidvogel. Comparison of advection schemes for high-order h-p finite element and finite volume methods. *Ocean Modelling*, 10:233–252, 2005. 3

[78] J. Thuburn and T. W. N. Haine. Adjoint of nonoscillatory advection schemes. *Journal of Computational Physics*, 171:616–631, 2001. 3

[79] J. M. Burgers. A mathematical model illustrating the theory of turbulence. *Advances in Applied Mechanics*, 1:171–199, 1948. 3.1

[80] G. Whitham. *Linear and Nonlinear Waves*. Wiley-Interscience, 1974. 3.1

[81] D. S. Zhang, G. W. Wei, D. J. Kouri, and D. K. Hoffman. Burgers' equations with high Reynolds number. *Physics of Fluids*, 9:1853–1855, 1997. 3.1

[82] C. A. J. Fletcher. *Computational techniques for fluid dynamics*. Springer-Verlag: Berlin, 1988. 3.1

[83] B. van Leer. *Towards the ultimate conservative difference scheme. I: The quest for monotonicity*, chapter Springer Lecture Notes in Physics, pages 163–168. Springer, 1973. 3.1

[84] B. van Leer. Towards the ultimate conservative difference scheme. II: Monotonicity and conservation combined in a second order scheme. *Journal of Computational Physics*, 140:361–370, 1974. 3.1

[85] B. van Leer. Towards the ultimate conservative difference scheme. III: Upstream-centered finite difference schemes for ideal compressible flow. *Journal of Computational Physics*, 23:263–275, 1977. 3.1

[86] S. Osher. Convergence of generalized MUSCL schemes. *SIAM Journal of Numerical Analysis*, 22:947–961, 1985. 3.1

[87] A. Harten. High resolution schemes for hyperbolic conservations laws. *Journal of Computational Physics*, 49:357–393, 1983. 3.1.1

[88] A. Harten. On a class of high resolution total variation stable finite difference schemes. *SIAM Journal of Numerical Analysis*, 21:1–23, 1984. 3.1.1

[89] B. van Leer. Towards the ultimate conservative difference scheme. V: A second order sequel to godunov' s method. *Journal of Computational Physics*, 32:101–136, 1979. 4, 5

[90] A. Harten and S. Osher. Uniformly high-order accurate nonoscillatory schemes. I. *SIAM Journal of Numerical Analysis*, 24:279–309, 1987. 3.1.2

[91] A. Harten, B. Engquist, S. Osher, and S. R. Chakravarthy. Uniformly high-order accurate essentially non-oscillatory schemes, III. *Journal of Computational Physics*, 71:231–303, 1987. 3.1.2

[92] A. Harten. ENO schemes with subcell resolution. *Journal of Computational Physics*, 83:148–184, 1987. 3.1.2

[93] A. Harten, P. D. Lax, and B. van Leer. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM Review*, 25:35–61, 1983. 3.1.2

[94] P. Colella and P. Woodward. The piecewise-parabolic method (PPM) for gas-dynamical simulations. *Journal of Computational Physics*, 54:174–201, 1984. 3.1.2, 3.1.3

[95] P. R. Woodward and P. Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of Computational Physics*, 54:115–173, 1984. 3.1.2

[96] R. L. Carpenter, K. K. Droegemeier, P. W. Woodward, and C. E. Hanem. Application of the piecewise parabolic method (PPM) to meteorological modeling. *Monthly Weather Review*, 118:586–612, 1990. 3.1.2

[97] C-W. Shu. Total-variation-diminishing time discretizations. *SIAM Journal of Scientific Statical Computation*, 9:1073–1084, 1988. 3.1.4

[98] C-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock capturing schemes. *Journal of Computational Physics*, 77:439–471, 1988. 3.1.4

[99] S. Gottlieb and C-W. Shu. Total variation diminishing Runge-Kutta schemes. *Mathematics of Computation*, 67:73–85, 1998. 3.1.4

[100] S.-J. Lin. A finite-volume integration method for computing pressure gradient force in general vertical coordinates. *Q. J. R. Meteorological Society*, 123:1749–1762, 1997. 3.1.5

[101] D. L. Williamson, J. B. Drake, J. J. Hack, R. Jakob, and P. N. Swarztrauber. A standard test set for numerical approximations to the shallow water equations in spherical geometry. *Journal of Computational Physics*, 102:211–224, 1992. 3.1.5, 3.2.3, 4.1

[102] G. J. Haltiner and R. T. Williams. *Numerical Prediction and Dynamic Meteorology*. Wiley: New York, 2nd. edition, 1980. 3.1.5

[103] M. J. Suarez and L. L. Takacs. Documentation of the ARIES/ GEOS dynamical core. NASA Technical Memorandum 104606, NASA, 1996. 3.1.5

[104] E. R. Benton and G. W. Platzman. A table of solutions of the one-dimensional Burgers equation. *Quarterly of Applied Mathematics*, 30:195–212, 1972. 3.2.1

[105] http://www.extra.rdg.ac.uk/Maths/Research/Publications/Msc_dissertations/rakhib_ahmed.pdf. Numerical schemes applied to the Burgers and Buckley-Leverett equations, R. Ahmed. 3.2.2

[106] http://www.ccsm.ucar.edu/models/atm cam. National center for atmospheric research (NCAR), The community atmosphere model (CAM). 3.2.2, 4.1

[107] N. A. Philips. Numerical integration of the primitive equations on the hemisphere. *Monthly Weather Review*, 87:333–345, 1959. 3.2.3

[108] B. Haurwitz. The motion of atmospheric disturbances on the spherical earth. *Journal of Marine Research*, 3:254–267, 1940. 3.2.3

[109] W. Yang and I. M. Navon. 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 Report Series on Global Modeling and Data Assimilation 104606, NASA, 1996. 3.3

[110] R. Giering and T. Kaminski. Recipes for adjoint code construction. *ACM Transactions on Mathematical Software*, 24:437–474, 1998. 3.3

[111] A. Griewank. *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*. SIAM, Philadelphia, 2000. 3.3

[112] http://www.csit.fsu.edu/∼navon/publ.html. Documentation of the TLM and adjoint models of the Lin-Rood spherical shallow water finite volume model. 3.3

[113] D. C. Liu and J. Nocedal. On the limited memory BFGS method for large scale minimization. *Mathematical Programming*, 45:503–528, 1989. 3.4, 3.4, 4.2

[114] S. G. Nash and J. Nocedal. A numerical study of the limited memory BFGS method and the truncated-Newton method for large scale optimization. *SIAM Journal on Optimization*, 1:358–372, 1991. 3.4, 3.4, 4.2

[115] D. Daescu and I. M. Navon. An analysis of a hybrid optimization method for variational data assimilation. *International Journal of Computational Fluid Dynamics*, 17:231–233, 2003. 3.4

[116] X. Zou, I. M. Navon, M. Berger, M. K. Phua, T. Schlick, and F.-X. LeDimet. Numerical experience with limited-memory, quasi-newton methods for large-scale unconstrained nonlinear minimization. *SIAM Journal on Optimization*, 3:582–608, 1993. 3.4, 4.2

[117] D. Dee. On-line estimation of error covariance parameters for atmospheric data assimilation. *Monthly Weather Review*, 123:1128–1145, 1995. 4

[118] G. J. Boer. A spectral analysis of predictability and error in an operational forecast system. *Monthly Weather Review*, 112:1183–1197, 1984. 4

[119] A. Dalcher and E. Kalnay. Error growth and predictability in operational ECMWF forecasts. *Tellus Series A-Dynamic Meteorology and Oceanography*, 39:474–491, 1987. 4

[120] S. C. Bloom and S. Shubert. The influence of Monte-Carlo estimates of model error growth on the GLA OI assimilation system. In *International symposium on assimilation of observations in meteorology and oceanography*, Geneva, Switzerland, 1990. WMO. 4

[121] D. Zupanski and M. Zupanski. Fine-resolution 4D-VAR data assimilation for the great plains tornado outbreak of 3 May 1999. *Weather and Forecasting*, 121:506–525, 2002. 4

[122] H. J. Thiébaux and L. L. Morone. Short-term systematic errors in global forecasts: their estimation and removal. *Tellus Series A-Dynamic Meteorology and Oceanography*, 42:209–229, 1990. 4

[123] S. Saha. Response of the NMC MRF model to systematic error correction within the integration. *Monthly Weather Review*, 120:345–360, 1992. 4

[124] G. Chepurin, J. Carton, and D. Dee. Forecast model bias correction in ocean data assimilation. *Monthly Weather Review*, 133:1328–1342, 2005. 4

[125] D. Dee and R. Todling. Data assimilation in the presence of forecast bias: the GEOS moisture analysis. *Monthly Weather Review*, 127:3268–3282, 2000. 4

[126] D. Dee and A. da Silva. Maximum-likelihood estimation of forecast and observational error covariance parameters. Part 1: Methodology. *Monthly Weather Review*, 127:1822–1834, 1999. 4

[127] D. Dee, G. Gaspari, C. Redder, L. Rukhovets, and A. da Silva. Maximum-likelihood estimation of forecast and observational error covariance parameters. Part 2: Applications. *Monthly Weather Review*, 127:1835–1849, 1999. 4

[128] D. Dee and A. da Silva. Data assimilation in the presence of forecast bias. *Q. J. R. Meteorological Society*, 124:269–295, 1998. 4, 4, 4.3, 5

[129] M. Zupanski. Maximum-likelihood ensemble filter: theoretical aspects. *Monthly Weather Review*, 133:1710–1726, 2005. 4

[130] A. Bennett, L. Leslie, C. Hagelberg, and P. Powers. Tropical cyclone prediction using a barotropic model initialized by a generalized inverse method. *Monthly Weather Review*, 121:1714–1729, 1993. 4

[131] A. Bennett, B. Chua, and L. Leslie. Generalized inversion of a global numerical weather prediction model. *Meteorology and Atmospheric Physics*, 60:165–178, 1996. 4

[132] A. Bennett, B. Chua, and L. Leslie. Generalized inversion of a global numerical weather prediction model (II). *Meteorology and Atmospheric Physics*, 62:129–140, 1997. 4

[133] F. Uboldi and M. Kamachi. Time-space weak-constraint data assimilation for nonlinear models. *Tellus Series A-Dynamic Meteorology and Oceanography*, 52:412–421, 2000. 4

[134] R. Daley. The effect of serially correlated observation and model error on atmospheric data assimilation. *Monthly Weather Review*, 120:164–177, 1992. 4, 4, 4, 4.2

[135] R. Daley. Estimating model-error covariances for application to atmospheric data assimilation. *Monthly Weather Review*, 120:1735–1746, 1992. 4

[136] J. Zhu and M. Kamachi. An adaptive variational method for data assimilation with imperfect models. *Tellus Series A-Dynamic Meteorology and Oceanography*, 52:265–279, 2000. 4

[137] J. C. Derber. A variational continuous assimilation technique. *Monthly Weather Review*, 117:2437–2446, 1989. 4, 4.3

[138] W. Wergen. The effect of model errors in variational assimilation. *Tellus Series A-Dynamic Meteorology and Oceanography*, 44:297–313, 1992. 4

[139] M. Zupanski. Regional four-dimensional variational data assimilation in a quasi-operational forecasting environment. *Monthly Weather Review*, 121:2396–2408, 1993. 4

[140] A. K. Griffith and N. K. Nichols. *Computational Differentiation: Techniques, Applications, and Tools*, chapter Accounting for model error in data assimilation using adjoint methods, pages 195–205. SIAM: Philadelphia, 1996. 4

[141] A. K. Griffith and N. K. Nichols. Adjoint methods in data assimilation for estimating model error. *Flow, Turbulence and Combustion*, 65:469–488, 2000. 4

[142] J. Martin, M. Bell, and N. Nichols. Estimation of systematic error in an equatorial ocean model using data assimilation. *International Journal of Numerical Methods in Fluids*, 40:435–444, 2002. 4

[143] P. A. Vidard, E. Blayo, F.-X. Le Dimet, and A. Piacentini. 4D variational data analysis with imperfect model. *Flow, Turbulence and Combustion*, 65:489–504, 2000. 4

[144] P. A. Vidard, A. Piacentini, and F.-X. Le Dimet. Variational data analysis with control of the forecast bias. *Tellus Series A-Dynamic Meteorology and Oceanography*, 56:177–188, 2004. 4, C, D

[145] D. Zupanski. A general weak constraint applicable to operational 4DVAR data assimilation systems. *Monthly Weather Review*, 125:2274–2292, 1997. 4

[146] M. Zupanski, D. Zupanski, T. Vukicevic, K. Eis, and T. Haar. CIRA/CSU Four-Dimensional variational data assimilation system. *Monthly Weather Review*, 123:829–843, 2005. 4

[147] S. Akella and I. M. Navon. A comparative study of the performance of high resolution advection schemes in the context of data assimilation. *International Journal of Numerical Methods in Fluids*, In Press, 2006. 4

[148] F.-X. LeDimet and V. Shutyaev. On deterministic error analysis in variational data assimilation. *Nonlinear Processes in Geophysics*, 12:481–490, 2005. 4

[149] http://www.ecmwf.int/research/era/. ECMWF Re-Analysis Project, ERA-40. 4.1

[150] R. Daley. Estimating observation error statistics for atmospheric data assimilation. *Annales Geophysicae-Atmospheres Hydrospheres and Space Sciences*, 11:634–647, 1993. 4.2

[151] I. M. Navon, D. N. Daescu, and Z. Liu. The impact of background error on incomplete observations for 4d-var data assimilation with the FSU GSM. In V. S. Sunderam, editor, *Computational Science- ICCS 2005, Part 2, Lecture Notes in Computer Science 3515*, pages 837–844. Springer-Verlag, Berlin Verlag Berlin, 2005. 4.2

[152] J. C. Derber and F. Bouttier. A reformulation of the background error covariance in the ECMWF global data assimilation system. *Tellus Series A-Dynamic Meteorology and Oceanography*, 51:195–221, 1999. 4.2, C

[153] L. P. Riishøjgaard. A direct way of specifying flow-dependent background error correlations for meteorological analysis systems. *Tellus Series A-Dynamic Meteorology and Oceanography*, 50:42–57, 1998. 4.2

[154] A. Weaver and P. Courtier. Correlation modeling on the sphere using a generalized diffusion equation. *Q. J. R. Meteorological Society*, 127:1815–1846, 2001. 4.2, C, C

[155] G. Gaspari and S. Cohn. Construction of correlation functions in two and three dimensions. *Q. J. R. Meteorological Society*, 125:723–757, 1999. 4.2

[156] H. Jarvinnen, J. N. Thépaut, and P. Courtier. Quasi-continuous variational data assimilation. *Q. J. R. Meteorological Society*, 122:515–534, 1996. 5

[157] P. Holmes, J. L. Lumley, and G. Berkooz. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge Monographs on Mechanics. Cambridge University Press, 1996. 5

[158] J.M. Restrepo, G. K. Leaf, and A. Griewank. Circumventing storage limitations in variational data assimilation studies. *SIAM Journal on Scientific Computing*, 19:1586–1605, 1998. B

[159] A. Griewank and A. Walther. An implementation of checkpointing for the reverse or adjoint mode of computational differentiation. *ACM Transactions on Mathematical Software*, 26:19–45, 2000. B

BIOGRAPHICAL SKETCH

Santharam Akella

Santha Akella was born on August 5, 1977, in a village (called Tuni, latitude 17.21 N, longitude 82.55 E), in south India. He pursued a Bachelor of Technology degree in Ocean Engineering and Naval Architecture from the Indian Institute of Technology, in Kharagpur (India) during 1996-2000. In the fall of 2000, he started to work towards a Ph.D. in Applied and Computational Mathematics under the supervision of Ionel M. Navon in the area of optimal control of distributed parameter systems and variational data assimilation methods.

Santha' s research interests include minimization methods, computational fluid dynamics, high performance scientific computing, etc. His non-research interests include playing tennis, road-biking, running, soccer, imitating sketching and painting styles of Rembrandt and van Gogh.