WRF-Var Implementation for Data Assimilation Experimentation at MIT

by
John K. Williams
B.S., Yale University (2005)

Submitted to the Department of Earth, Atmospheric, and Planetary Sciences
in partial fulfillment of the requirements for the degree of
Masters in Atmospheric Science
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
September 2008

Author ... .
Department of Earth, Atmospheric, and Planetary Sciences
August 22nd, 2008

Certified by..... .....
Kerry A. Emanuel
Professor
Thesis Supervisor

Certified by .....
Sai Ravela
Research Scientist
Research Supervisor

Certified by .....
Christopher Hill
Principal Research Scientist
Committee Member

Accepted by ....
Maria T. Zuber
E.A. Griswold Professor of Geophysics
Head, Deparment of Earth, Atmospheric and Planetary Sciences

WRF-Var Implementation for Data Assimilation Experimentation at MIT

by

John K. Williams

Submitted to the Department of Earth, Atmospheric, and Planetary Sciences
on August 22nd, 2008, in partial fulfillment of the requirements for the degree of
Masters in Atmospheric Science

Abstract

The goal of this Masters project is to implement the WRF model with 3D variational assimilation (3DVAR) at MIT. A working version of WRF extends the scope of experimentation to mesoscale problems in both real and idealized scenarios. A state-of-the-art model and assimilation package can now be used to conduct science or as a benchmark to compare new methods with.

The second goal of this project is to demonstrate MIT's WRF implementation in an ongoing study of the impact of position errors on contemporary data assimilation (DA) methods [21]. In weather forecasting, accurately predicting the position and shape of small scale features can be as important as predicting their strength. Position errors are unfortunately common in operational forecasts [2, 14, 21, 27] and arise for a number of reasons. It is difficult to factor error into its constituent sources [21].

Traditional data assimilaton methods are amplitude adjustment methods, which do not deal with position errors well [4, 21]. In this project, we configured the WRF-Var system for use at MIT to extend experimentation on data assimilation to mesoscale problems. We experiment on position errors with the WRF-Var system by using a standard WRF test; a tropical cyclone. The results for this identical twin experiment show the common distorted analysis from 3DVAR in dealing with position errors. A *field alignment* solution proposed by Ravela et al. [21] explicitly represents and minimizes position errors. We achieve promising results in testing this algorithm with WRF-Var by aligning WRF fields from the identical twin.

Thesis Supervisor: Kerry A. Emanuel
Title: Professor

Research Supervisor: Sai Ravela
Title: Research Scientist

Committee Member: Christopher Hill
Title: Principal Research Scientist

Acknowledgments

Discussion on the WRF Forum shows that getting all parts of the V2.2 WRF-Var system working on a local machine is a practically impossible task due to the bugs and sparse nature of documentation. My experience was no different, as I would not have been able to achieve this in a reasonable time frame without extensive help from Sai Ravela. I similarly owe him a large debt of gratitude for his tireless help introducing me to the field of state estimation, monitoring my research progress, and editing this thesis.

I would also like to thank Professors Kerry Emanuel and Dennis McLaughlin for their grounding, big picture suggestions, and Chris Hill for his last minute help.

This work was supported by NSF Grants CNS 0540248 and CNS 0540259.

Contents

List of Figures

Chapter 1

Introduction

The goal of this Masters project is to implement the WRF model with 3D variational assimilation (3DVAR) at MIT. Although research at MIT includes other coupled physical-numerical systems, such as the planet-in-a-bottle project by Ravela et al. [22, 23, 24], a working version of WRF extends the scope of experimentation to mesoscale problems in both real and idealized scenarios. A state-of-the-art model and assimilation package[1] can now be used to conduct science or as a benchmark to compare new methods with.

Accordingly, a second goal of this project is to demonstrate MIT's WRF implementation in an ongoing study of the impact of position errors on contemporary data assimilation (DA) methods [21]. We now go on to briefly discuss the position-error problem and the experimental setup.

1.1 Position Errors

In weather forecasting, accurately predicting the position and shape of small scale features can be as important as predicting their strength. Errors in a forecast for a squall line, for example, can be the difference between a population center that is prepared and one that is off-guard. Thus, such errors can carry a high cost in terms of both life and business.

[1] http://www.wrf-model.org/

Position errors are unfortunately common in operational forecasts [2, 14, 21, 27] and they arise for a number of reasons including timing errors, parameterized and/or approximated physics, coarse resolution, and errors in the background flow [11, 12, 18, 21]. It is difficult to factor position error into its constituent sources [21].

Traditional data assimilaton methods are amplitude adjustment methods, which do not deal with position errors well [4, 21]. Strong, small scale features with differences in position or shape between model and observed data can lead to large amplitude errors. These errors are not accounted for in typical background statistics and, indeed, background statistics which are adequate when no position errors exist are no longer so because of bias, distorted background covariances [21] or both. Thus, model adjustments are sub-optimal in that distorted analyses result while position error remains (as we will show with a simulated tropical cyclone example). Obviously, this can lead to poor forecasts and a corrupt forecast-assimilate cycle.

Operational users have already developed schemes to deal with position errors. For example, *bogussing* is a technique in which a tropical cyclone (TC) is forcibly removed from the model and manually replaced with a standard version in the correct location. Xiao et al. [30] discusses a bogussing scheme for use with WRF-Var that involves introducing a set of *bogus observations* instead of changing the model state directly. Symmetric sea level pressure observations are created from typhoon reports, using an empirical formula, and bogus wind observations are calculated based on an idealized gradient wind relationship. The errors to these synthetic observations are chosen to be small enough so confidence in the bogussed observations overwhelm the existence of the TC in the background. If the TC in the background is very strong and remains after the assimilation cycle then a removal algorithm is used. This is imporant because if any piece of the old cycle remains it can interact in a way to worsen forecasts as. Bogussed observations must also be designed carefully so as not to cause shock to the model. Clearly, this is an ad hoc method and not an optimal solution to the problem.

In a recent paper, Ravela et al. [21] propose a *field alignment* solution which explicitly represents and minimizes position errors. Control variables that represent

displacements are estimated at each node of the state grid and specify the deformation of the grid. A 2-step method of both amplitude and position adjustment can be conveniently used as an approximation to the minimization problem, and thus can be combined with traditional state-estimation methods in a current model data assimilation package. However, tests have not been carried out with a real world model. A goal of this project is to test the plausibility of using the field alignment algorithm with WRF-Var by aligning actual WRF model fields after simulating a position error.

1.2 WRF-Var Implementation & Experiment

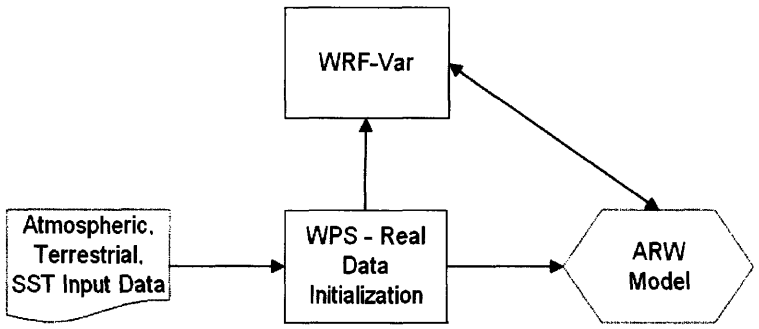

Figure 1-1: WRF system flow chart [28]

As shown in Figure 1-1 the entire WRF system consists of four components: input data, WPS, WRF-ARW Model, and WRF-Var. The WRF Preprocessing System (WPS) processes input data for a simulation. The Advanced Research WRF (ARW) uses the WPS output to initialize the simulation and set the boundary conditions and then runs the forecast. WRF-Var utilizes 3DVAR to assimilate observations to create an analysis from the forecast, and then updates boundary conditions to launch another forecast.

For experimentation, a standard WRF test case is used; a 2 day simulation of Hurricane Katrina. As the short range forecast for Katrina performed well in reality, forecast position errors are simulated through a zonal wind perturbation. The real simulation is also run and functions as both truth and a source of synthetic

observations. Using the mechanics of this set up, real-time forecasting and DA experimentation for future TCs is possible, using real observations from the MADIS network[2].

Implementing the WRF-VAR system on a standard test-case should've been merely an exercise in dowloading, compiling and running. Unfortunately, our experience suggested otherwise, entirely. As documented locally[3] and on the website used for WRF user discussion at the time[4], our effort required a number of forays into the code and its invocation. Indeed, this effort seems to be of substantial benefit to other researchers too. They are thus documented.

1.3 Organization

The rest of this thesis is arranged as follows. Chapter two describes the WRF forecast model and configuration issues. Chapter three provides an overview of data assimilation methods and describes how 3DVAR is implemented by WRF-Var. Chapter four describes setup for the identical twin experiment involving position errors and then compares results of WRF-Var assimilation in the experiment to field alignment of WRF fields.

[2]http://madis.noaa.gov/madis_wrf-var.html
[3]http://ecovision.mit.edu/~ecovision/forum/
[4]http://tornado.meso.com/wrf_forum/

Chapter 2

WRF Model

As shown in Figure 1-1, the Weather Research and Forecasting model (WRF) is a mesoscale forecast model developed by several collaborating institutions [26]. It is supported as a community model for research but is also run operationally at many private and public institutions[1]. The model is claimed to be fully portable and comes with an initialization routine, making it suitable for both real and idealized experiments.

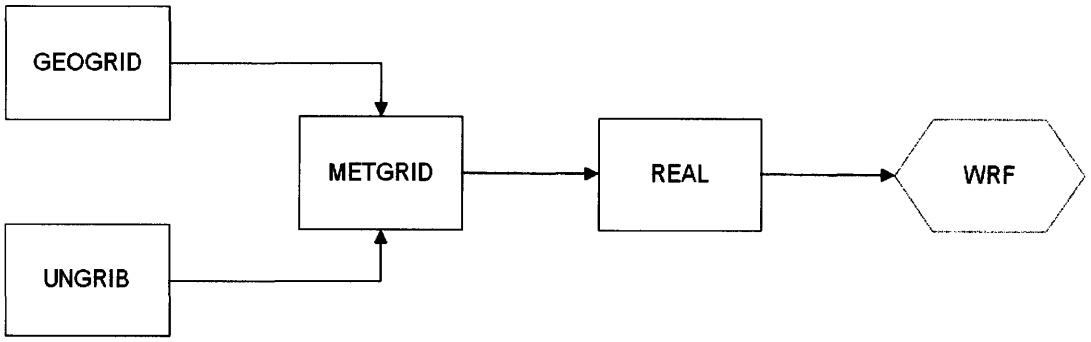

Figure 2-1: WPS and ARW WRF flow chart [28]

The steps for running a real data case in WRF are shown in Figure 2-1. In this chapter we briefly discuss the steps necessary to run a simulation and the integration of equations/physics included in the ARW solver. More detailed information

[1]http://wrf-model.org/plots/wrfrealtime.php

on running the model can be found in the ARW User's Guide [28] while technical information on ARW is in Skamarock et al. [26].

2.1 Domain & Input Data

The WRF Preprocessor System (WPS) is made up of the routines GEOGRID, UNGRIB, and METGRID shown in Figure 2-1. The functions are to set up the model domain and interpolate input data horizontally onto it.

The first step is the routine GEOGRID, which defines the projection and coordinates of the domain. It then interpolates 2D static land data[2] and outputs a single *netcdf* [25] file. Time dependent atmospheric and sea surface data are extracted from *grib* [1] formatted files by the utility UNGRIB. For the Katrina simulation, AVN 0.5^o analysis data is used for the atmospheric variables and NCEP 0.5^o RTG (Real-Time Global) analysis is used for sea surface temperatures. The routine METGRID interpolates all of the data horizontally onto the domain. The output from METGRID are a succession of *netcdf* files used for initialization and boundary conditions of the simulation.

2.2 Initialization & Boundary Conditions

The program REAL creates initial and boundary condition files for WRF. REAL interpolates atmospheric inputs vertically onto the hydrostatic pressure coordinate η. REAL also partitions some of the variables into a standard dry reference state and perturbation state [28]. The latter is defined by the difference between the reference state and the full state including moisture. The prognostic variables are thus in exact hydrostatic balance for the model equations. The initialization file outputted by REAL is a *netcdf* file with a single time at the start of the simulation.

Additionally REAL creates a lateral boundary condition file at the desired temporal spacing which defines the four sides of the rectangular grid. The fields have

[2]available from the MMM website (www.mmm.ucar.edu)

values valid at the boundary time and a tendency term to get to the next boundary time period [26]. For the Katrina test case, REAL was also used to create a bottom boundary condition file containing SST data. The top boundary condition is set as $\omega = 0$ (where ω is the pressure vertical velocity), and is gravity wave absorbing.

2.3 Integration of Governing Equations

The advanced research wrf (ARW) solver used for this project integrates the compressible, nonhydrostatic Euler equations in flux form [26]. The equations are formulated using a terrain-following hydrostatic-pressure vertical coordinate defined as

$$\eta = (p_h - p_{ht})/\mu \quad \text{where} \quad \mu = p_{hs} - p_{ht} \tag{2.1}$$

where p_h is the hydrostatic component of pressure, and p_{hs} and p_{ht} are surface and top boundary values. Model variables are converted to flux form as follows

$$\mathbf{V} = \mu\mathbf{v} = (U, V, W), \quad \Omega = \mu\dot{\eta}, \quad \Theta = \mu\theta \tag{2.2}$$

where $\mathbf{v} = (u, v, w)$ are covariant velocities in the horizontal and vertical directions while $\omega = \dot{\eta}$ is the contravariant vertical velocity, and θ is the potential temperature. The prognostic flux-form dry Euler equations are

$$\begin{aligned}
\partial_t U + (\nabla \cdot \mathbf{V}u) - \partial_x(p\phi_\eta) + \partial_\eta(p\phi_x) &= F_U && (2.3)\\
\partial_t V + (\nabla \cdot \mathbf{V}v) - \partial_y(p\phi_\eta) + \partial_\eta(p\phi_y) &= F_V && (2.4)\\
\partial_t W + (\nabla \cdot \mathbf{V}w) - g(\partial_\eta p - \mu) &= F_W && (2.5)\\
\partial_t \Theta + (\nabla \cdot \mathbf{V}\theta) &= F_\Theta && (2.6)\\
\partial_t \mu + (\nabla \cdot \mathbf{V}) &= 0 && (2.7)\\
\partial_t \phi + \mu^{-1}[(\mathbf{V} \cdot \nabla\phi) - gW] &= 0 && (2.8)
\end{aligned}$$

where the subscripts x, y, and η denote differentiation, g is the gravitational constant, $\phi = gz$ is the geopotential, $\alpha = 1/\rho$ is the inverse density, and the right-hand-side terms F_U, F_V, F_W, and F_Θ represent forcing terms from model physics, turbulent mixing, spherical projections, and the earth's rotation. The diagnostic relation for the inverse density and the equation of state respectively

$$\partial_\eta \phi = -\alpha\mu \tag{2.9}$$

$$p = p_0(R_d\theta/p_0\alpha)^\gamma \tag{2.10}$$

are used to close the system, where $\gamma = c_p/c_v = 1.4$ is the ratio of heat capacities for dry air, R_d is the gas constant for dry air, and p_0 is a reference pressure.

The full equations solved by the ARW are put in perturbation form, and include moisture and map factors for different projections [26]. To include moisture the vertical coordinate is written as

$$\eta = (p_{dh} - p_{dht})/\mu_d \tag{2.11}$$

where μ_d is the mass of dry air in the column and p_{dh} and p_{dht} are the hydrostatic pressure of the dry atmosphere and hydrostatic pressure at the top of the dry atmosphere. The coupled variables are

$$\mathbf{V} = \mu_d\mathbf{v} = (U, V, W), \quad \Omega = \mu_d\dot{\eta}, \quad \Theta = \mu_d\theta. \tag{2.12}$$

To include map factors, the momentum variables are redefined

$$U = \mu_d u/m, \quad V = \mu_d v/m, \quad W = \mu_d w/m, \quad \Omega = \mu_d\dot{\eta}/m \tag{2.13}$$

where m is the ratio of the distance in computational space to the corresponding distance on the earth's surface. Lastly, variables are defined with respect to a hydrostatic reference state that is only a function of z. Pressure p, velocity potential ϕ,

inverse density α, and mass of the column μ_d are defined

$$x = \bar{x}(z) + x' \tag{2.14}$$

where x is each of the mentioned variables respectively. The momentum equations are now written

$$\partial_t U + m[\partial_x(Uu) + \partial_y(Vu)] + \partial_\eta(\Omega u) + (\mu_d \alpha \partial_x p' + \mu_d \alpha' \partial_x \bar{p}) \tag{2.15}$$

$$+(\alpha/\alpha_d)(\mu_d \partial_x \phi' + \partial_\eta p' \partial_x \phi - \mu_d' \partial_x \phi) = F_U \tag{2.16}$$

$$\partial_t V + m[\partial_x(Uv) + \partial_y(Vv)] + \partial_\eta(\Omega v) + (\mu_d \alpha \partial_y p' + \mu_d \alpha' \partial_y \bar{p}) \tag{2.17}$$

$$+(\alpha/\alpha_d)(\mu_d \partial_y \phi' + \partial_\eta p' \partial_y \phi - \mu_d' \partial_y \phi) = F_V \tag{2.18}$$

$$\partial_t W + m[\partial_x(Uw) + \partial_y(Vw)] + \partial_\eta(\Omega w) \tag{2.19}$$

$$-m^{-1} g(\alpha/\alpha_d)[\partial_\eta p' - \bar{\mu}_d(q_v + q_c + q_r)] + m^{-1}\mu_d' g = F_W \tag{2.20}$$

where q_v, q_c, and q_r are mixing ratios for vapor, clouds, and rain, the mass conservation and geopotential equation respectively are

$$\partial_t \mu_d' + m^2[\partial_x U + \partial_y V] + m\partial_\eta \Omega = 0 \tag{2.21}$$

$$\partial_t \phi' + \mu_d^{-1}[m^2(U\phi_x + V\phi_y) + m\Omega\phi_\eta - gW] = 0 \tag{2.22}$$

where q_m is the total mixing ratio and $Q_m = \mu_d q_m$, the conservation equations for the potential temperature and scalars are

$$\partial_t \Theta + m^2[\partial_x(U\theta) + \partial_y(V\theta)] + m\partial_\eta \Omega\theta) = F_\Theta \tag{2.23}$$

$$\partial_t Q_m + m^2[\partial_x(Uq_m) + \partial_y(Vq_m)] + m\partial_\eta(\Omega q_m) = F_{Q_m} \tag{2.24}$$

the hydrostatic relation is

$$\partial_\eta \phi' = -\bar{\mu}_d \alpha_d' - \alpha_d \mu_d' \tag{2.25}$$

and the equation of state is

$$p = p_0(R_d \theta_m / p_0 \alpha_d)^\gamma \tag{2.26}$$

where $\phi_m \approx \phi(1 + 1.61q_v)$.

The equations 2.16 through 2.26 are solved for using a time-split integration scheme, where the meteorologically significant low-frequency modes are integrated using third-order Runge-Kutta (RK3) while acoustic modes are integrated over smaller time steps for numerical stability [26]. Acoustic integration is cast as a correction to the RK3 integration and acoustic modes are filtered by using divergence damping. Diabatic forcing is integrated within the acoustic steps as well. For the Katrina simulation, 5^{th} order accurate spatial discretizations (using an Arakawa-C grid) of the flux divergence are used for momentum, scalars, and geopotential.

The time steps for low and high frequency modes were chosen based on stability constraints. The maximum time step for advection in RK3 is given by

$$\Delta t_{max} < \frac{Cr_{theory}}{\sqrt{3}} \cdot \frac{\Delta x}{u_{max}} \tag{2.27}$$

where Cr is the Courant number, Δx is the grid size, and u_{max} is the expected maximum velocity [26]. For the Katrina simulation the Δt_{max} value was $246s$. The WRF manual recommendation is to stay 25% below the maximum figure, so the time step of $180s$ is used. The ratio of the advection time step to the acoustic time step is set as the default value of $\frac{1}{4}$. This time scale length ensures both stability and ease of experimentation, as a shorter time scale leads to longer wall-clock running time.

2.4 Physics

Model physics are categorized in a modular way into microphysics, cumulus parameterization, planetary boundary layer (PBL), land-surface model, and radiation. There are multiple physics options available for each module to be chosen by the user depending on the type of simulation. Details on each option (including those that follow in this section) are found in Skamarock et al. [26].

For microphysics in the Katrina simulation, the WRF Single-Moment 3-class simple ice scheme is used. For cumulus parameterization the Kain-Fritsch scheme is used.

To calculate surface heat/moisture fluxes for the land-surface model and surface stress for the PBL scheme, the Eta surface layer scheme based on similarity theory is used. The land-surface model uses a 5-layer thermal diffusion based on the MM5 5-layer soil temperature model. The PBL module uses Yonsei University PBL. The radiation scheme for longwave is the RRTM (Rapid Radiative Transfer Model) taken from the MM5 model and the shortwave scheme is similarly taken from MM5. Interactions between the modules occur via model state variables and surface fluxes.

2.5 Configuration

The WRF model is configured for *ecovision.mit.edu*, the computer used for this research, using the linux, single processor option. A requirement is that *netcdf* libraries are compiled by the same compiler used by WRF, and that they are linked to WRF by setting the environmental variable "NETCDF" to the location of the libraries.

A segmentation fault runtime error in WPS v2.2.1 is remedied by inserting the following statement at line 325 of */WPS/geogrid/src/interp_module.f90*:

```
call mprintf(.true.,STDOUT,'maskval. %f',f1=maskval)
```

At first glance, this seems like a bizarre modification and one that would have no bearing on the functioning of the code because it is merely a print statement. It turns out to be, however, a critical patch to the program that prevented it from segfaulting. The mechanics of memory management in fortran, the key to make this statment meaningful, are unfortunately beyond the scope of this document. Discussion of this fix is found on the WRF Forum [3].

[3] http://tornado.meso.com/wrf_forum/

Chapter 3

Data Assimilation

The numerical weather prediction (NWP) problem requires an estimate of the future state of the atmosphere from a set of initial conditions and time varying boundary conditions. This is predominantly accomplished using approximated governing equations making up the model.

Prof. Lorenz's pioneering work [17] has shown that errors merely due to finite numerical representation will result in a nonlinear and chaotic error growth between model forecasts and truth. If the uncertainty in the numerical representation of a model's state were perfectly known, then one could invoke the Fokker-Plank equation to forecast both states and their uncertainties, up to the predictability limit suggested by some suitable objective measure on the distribution.

However, not only is an analytic solution to the Fokker-Planck impossible, but even an appropriate representation of uncertainty, which is an even more fundamental issue, is far from understood. To complicate matters further, models are not perfect (and some believe they never can be). There is no chance that Fokker-Plank is a viable means for effective prediction, although it is a reasonable mechanism for understanding predictability limits.

Our best bet to improve predictions, many will argue, is to couple numerical models with observations of the real system in an attempt to arrest a forecast distribution from diverging. This can be accomplished by inferring a variety of properties of the coupled physical-numerical system. Just what is inferred (and how) is the subject of

study called *data assimilation* in the meteorological community and, as such, it is a proper subset of estimation, control, optimization and stochastic systems theory.

Alas, the data assimilation problem is also hard. We now have a high-dimensional, imperfect numerical representation of nonlinear governing equations. Not only are initial and boundary conditions for the model unknown, but real atmospheric data are available from relatively sparse and noisy data and not necessarily in the form of model variables. *State estimation* is in fact a fundamental challenge for NWP and much algorithmic and applied work is devoted to advancing it.

3.1 State Estimation Methods

State estimation is the process by which we constrain model states using observations, whilst balancing uncertainties in the knowledge of model states and observations. Current methods for state estimation can be broken into two general approaches, filtering and smoothing.

Smoothing is the natural framework when we wish to estimate all states, typically at discrete times within a time window, using all available observations. A *batch smoother* is conceptually the simplest of smoothers, which estimates model states at all times by creating a meta-state that concatenates all of them. However, such an approach is computationally expensive for all but the most trivial problems. Fortunately, for dynamical systems of interest to us, smoothing over specific intervals can be done sequentially.

The Rauch-Tung-Striebel algorithm [20] is an optimal sequential smoother for a linear system and approximate methods have been developed to deal with nonlinear systems. These include the variational forms such as the representer method [3] and 4DVAR [16]. The latter is also known as the adjoint method [29] and derived from solutions to two-point boundary value problems [5]. Another class of methods uses Monte-Carlo representations of uncertainty to statistically compute approximate adjoints, and this forms the basis of the ensemble Kalman smoother [8, 9].

Smoothing by any these methods implies that model states at all previous times in

the interval relative to a fixed observation time are updated. In an NWP application, however, we typically only care for an estimate (analysis) at a fixed point in time. For a fixed-point, smoothing may be used too, by leveraging observations in the future relative to it. This is decidedly better than only using observations from the past and present. For example, ECMWF (the European Centre for Medium-Range Weather Forecasts) uses observations within a time window to estimate the state at the beginning of the window and then launches a forecast in to the future from it. This form of smoothing, where a smaller finite window progresses within a larger or infinite time window is called fixed-lag smoothing. For a more complete discussion, see Ravela and McLaughlin [21].

Filtering refers to the estimation of state at the current time using observations from the entire past up to the present time. When viewed naively, the filtering problem might appear to have a disastrously increasing burden as time progresses. Fortunately, again, filtering dynamical systems of interest to us is recursive. A model forecast to the present is updated with current observations and a new forecast is launched from the analysis. The process repeats and in this way a model recursively benefits from an entire observation sequence. As we shall see shortly, a Bayesian formulation expresses recursive filtering in a direct and elegant manner.

At any given time of analysis however, an estimation problem must be solved to update model states and their uncertainties. The measure used to solve such problems is overwhelmingly a metric. Although metrics may come in different flavors, they share an underlying form that traces its history to least-squares methods for minimum variance estimation.

For example, Optimal Interpolation (OI) [15] utilizes a least squares approach for minimizing the errors between observations and a forecast. OI assumes an isotropic forecast error covariance matrix (also known as the background error covariance) for use at all assimilation times. 3DVAR [15] produces a best estimate analysis by minimizing a cost function by a variational approach, finding the solution through an iterative method[1]. The background errors in this case can be estimated in a variety

[1]This approach has the advantage of incorporating nonlinear relationships between observations

of ways, but tyically a weakly flow-dependent, but still static technique called the NMC method [19] is often used. This solution has been adopted as a standard for NWP in the US.

The Kalman filter [10, 15] is the optimal recursive filter for linear systems with additive Gaussian uncertainties. For a linear observation model under a Gaussian assumption, there is a methodological consistency between the Kalman filter and 3DVAR and OI. The former however produces an estimate of both the first and second moment for a Gaussian-parameterized forecast distribution. Doing so is optimal for linear-gaussian systems but this optimality is not achieved by OI or 3DVAR because the second moment is never updated. Thus, in addition to the linear-Gaussian requirement, 3DVAR and OI are optimal only to the degree the uncertainty in forecast distribution is modeled accurately.

Extensions to nonlinear systems take the usual form of linearizing nonlinear problems, and a linearization of the dynamics forms the basis for the extended Kalman filter [13]. However, linearized or linear, for large systems advancing the background error covariance is computationally expensive, if not prohibitive. The ensemble Kalman filter [8] (and many variants) provide a statistical alternative; an ensemble of forecasts is used to compute a reduced-rank estimate of the population covariance. Multiple simulations thus replace explicit propagation of the covariance, and the model is never linearized. For nonlinear dynamics, this approach appears to offer some advantages over the extended Kalman filter, but optimality is neither guaranteed nor observed and significant issues with regard to noise, fidelity and dispersal of the ensemble remain.

For nonlinear dynamical systems, which will transform initially Gaussian uncertainties into non-Gaussian uncertainties downstream, then, a new approach is required. A particle filter appears to be the the most promising. This filter implements the Bayes rule using Monte-Carlo methods but questions regarding appropriate representations of uncertainty and dimensionality remain. The particle filter is the subject of vigorous research in many fields.

---

and states.

In this chapter, we will study sequential filtering from a Bayesian point of view and relate it to contemporary practice, with entirely Gaussian assumptions in uncertainties and a linear relationship between observations and states. This will be a prelude to a discussion of 3DVAR using the NMC method, which is the method available with WRF. We will end the chapter with notes on configuring WRF-VAR for experiments conducted here, including code modifications to make the the distribution work.

3.2 Bayesian Formulation of Filtering

Let us assume a model state vector X_n at discrete time n is only a function of the state at a previous time $n-1$:

$$X_n = f(X_{n-1}, U_{n-1}) \tag{3.1}$$

where U_{n-1} is a vector of model inputs in the form of control variables and f is a nonlinear evolution function [2]. Observations are modeled linearly in this report and take the form

$$Y_n = HX_n + \eta. \tag{3.2}$$

The vector Y_n contains observations, H is a linear observation operator, and η is noise.

The goal is to estimate X_n^{true}, the true state of the atmosphere at discrete time t_n, by finding a model state that is most consistent with the observations. We assume that X_n^{true} can be captured by a statistic of the analysis's distribution. Thus we may posit a variable X_n representing a random model-state, such that $E[X_n] = X_n^{true}$. The goal is to find this analysis distribution and its statistics using one of the possibly many estimation methods.

In filtering, the estimation problem is reduced to quantifying the statistics of an analysis distribution, that is the distribution of model-state X_n conditioned on all

[2] We assume the model itself does not change with time, though states and control inputs obviously do.

observations Y_0 to Y_n. We quantify this distribution as

$$P(X_n|Y_{0:n}). \tag{3.3}$$

Assuming the observations are not correlated in time $P(Y_{0:n}) = \prod_{i=0}^{n} P(Y_i)$ and using the Markov property, Bayes rule can be used to express the posteriori probability 3.3 as

$$P(X_n|Y_{0:n}) = \frac{P(Y_n|X_n)P(X_n|Y_{0:n-1})}{P(Y_n)}. \tag{3.4}$$

The denominator is a distribution of observations which does not depend on model state and therefore can be ignored as just a normalization constant. Thus

$$\begin{aligned} P(X_n|Y_{0:n}) &\propto P(Y_n|X_n)P(X_n|Y_{0:n-1}) && (3.5)\\ &= P(Y_n|X_n)\int P(X_n|X_{n-1})P(X_{n-1}|Y_{0:n-1})dX_{n-1} && (3.6)\\ &= P(Y_n|X_n)P(X_n^f). && (3.7) \end{aligned}$$

Equation 3.5 expresses the recursive form of filtering, whilst Equation 3.6 expands it to properly include stochastic models (i.e. model error), and Equation 3.7 is the form we will use here, where $P(X_n^f)$ is the conditional prior or the forecast distribution. We will next assume a parametric (Gaussian) form of these distributions to estimate X_n^{true}.

3.3 Gaussian Assumption & Quadratic Objectives

If the distributions in Equation 3.7 are assumed to be Gaussian, we may write $X_n^f \sim N(X_n, B)$ and $\eta \sim N(0, R)$, where B and R are matrices of error covariances in the state and observations respectively. Thus, we can write:

$$P(X_n^f) \propto e^{-\frac{1}{2}[(X_n - X_n^f)^T B^{-1}(X_n - X_n^f)]} \tag{3.8}$$

and the probability of observations given state can be written

$$P(Y_n|X_n) \propto e^{-\frac{1}{2}[Y_n - H(X_n)]^T R^{-1}[Y_n - H(X_n)]} \tag{3.9}$$

The distributions in Equation 3.8 and 3.9 are assumed to be unbiased. That is, $X_n = E[X^f]$ and $HX_n = E[Y_n]$ is the mean of the conditional and prior distributions. We wish to quantify an unknown X_n as a statistic of two distributions, whose samples in the form of Y_n and X_n^f are given.

This solution may be written as a search for the maximimum a posterior probability, which is $X_n^a = \arg\max_{X_n} P(Y_n|X_n)P(X_n^f)$. Since we make the Gaussian assumption, our estimate of the system state is reduced to minimizing the negative logarithm of the above equations and the best estimate is thus $X_n^a = \arg\min_{X_n} J(X_n)$, where the $J(X_n)$ is written as

$$J(X_n) = \frac{1}{2}(Y_n - HX_n)^T R^{-1}(Y_n - HX_n) + \frac{1}{2}(X_n - X_n^f)^T B^{-1}(X_n - X_n^f) \tag{3.10}$$

If the dependence on time is dropped, then we may write the objective as

$$J(X) = (Y - HX)^T R^{-1}(Y - HX) + (X - X^f)^T B^{-1}(X - X^f) \tag{3.11}$$

The optimal estimate of the posterior (or analysis distribution) mean is the analysis, written as X^a, which in the case of a linear H is found by least squares to be

$$X^a = X^f + BH^T(HBH^T + R)^{-1}(Y - HX^f) \tag{3.12}$$

OI and the Kalman filter both utilize Equation 3.12. However, OI will use a constant isotopric B_0 matrix for all times, or may make a simple assumption about the behavior of $P(X^f)$ over time such as that the model integration increases initial error variance by a fixed amount. The Kalman filter updates the background error

covariance too. First define the Kalman gain as

$$K = BH^T(HBH^T + R)^{-1}. \tag{3.13}$$

Then, the posterior distribution is represented by the mean X^a and a covariance $B^a = (I - KH)B$. 3DVAR solves 3.11 iteratively, typically using the Conjugate Gradient algorithm. This has the advantage for a nonlinear observation operator, but here, since we assume the observation operator is linear, there is no difference from Equation 3.12. A key point however is that like OI, 3DVAR does not propagate the error covariance (or estimate it). Instead, the background error covariance, as B is called, will be determined using the NMC method [19] at every assimilation time. The ensemble Kalman filter can be seen as using an ensemble of forecasts and observation pairs. Each pair produces a single analysis state. The expectation of the analysis states is an estimate of truth, and the statistically computed covariance of analysis states is an estimate of the posterior covariance. The derivation and convergence of EnKF are beyond the scope of this document.

3.4 3DVAR in WRF-Var

We use 3DVAR because it is the most widely used method for data assimilation in NWP and is packaged with WRF. The cost function that WRF-Var seeks to solve is an adaptation of Equation 3.11:

$$J(X) = (Y - HX)^T(R + F)^{-1}(Y - HX) + (X - X^f)^T B^{-1}(X - X^f) \tag{3.14}$$

where F, the representivity error-covariance matrix, estimates the errors caused by H to change from analysis space to observation space [26]. Control variables are defined for the background state to efficiently approximate B using a preconditioner U

$$X' = U\mathbf{v} \tag{3.15}$$

where X' is the analysis increment, U represents the various stages of covariance modeling, and $\mathbf{v}$ is the control variable. With a suitably defined U, UU^T approximates B. The incremental cost function is thus written

$$J(\mathbf{v}) = \mathbf{v}^T\mathbf{v} + (Y_0' - HU\mathbf{v})^T(R+F)^{-1}(Y_0' - HU\mathbf{v}). \tag{3.16}$$

This is the form of the quadratic objective in WRF-Var [26]. The cost function is minimized through either the conjugate gradient method or quasi-Newton method.

Control variables used are streamfunction ψ, velocity potential ϕ, unbalanced surface pressure p_u, temperature T, and relative humidity r [26]. The analysis keeps imbalance to a minimum because the first guess is either a WRF generated forecast file or a REAL generated initial condition file. Balance is also imposed between mass and wind geostrophically and cyclostrophically, when statistically relevant based on regression coefficients. The transform is used for computational efficiency only in WRF-Var as the WRF model variables are again used for a subsequent forecast.

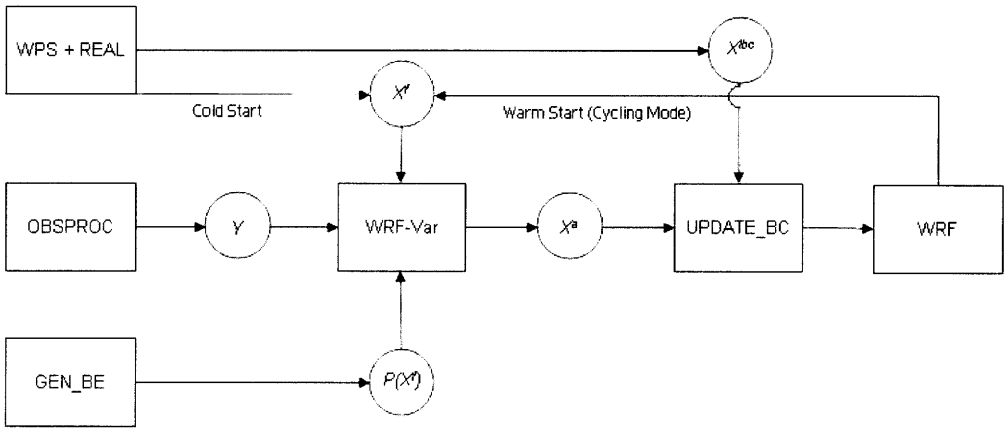

Figure 3-1: WRF-Var in the WRF system [28]

The WRF-Var system is shown in Figure 3-1. WRF-Var is run using a *cshell* script, where environmental variables are set according to start/end time, domain, quality control flags, etc. WRF-Var outputs an analysis file that is the same format as the *wrf_input* initialization file created by REAL. An additional utility UPDATE_BC is run to update the boundary condition file (originally created by running REAL).

With those two files another forecast can now be run with WRF, completing the assimilation cycle.

3.4.1 Observation Processing in WRF-Var

In order to be ingested by WRF-Var, observations must be in a proprietary format called *little_r* [28]. Observations accepted include U and V components, pressure, temperature, and relative humidty by either pressure or height. Also included are quality control flags and station information. This file is processed by the program 3DVAR_OBS, which outputs a file of similar format except with additional quality control, domain, and instrumental error data. WRF-Var itself also outputs information on how many and what kind of observations were assimilated, as a final diagnostic.

Standard uncertainties for sounding data are included in a file called *obserr.txt*. Instrumental errors are included for various air, water, and surface observation types as well as satellite retrievals. For radiosonde data, wind sensor errors are included on pressure levels of WRF forecast files and are based on those used in the FNMOC NOGAPS and ECMWF forecast models. Uncertainties can be used in quality control by throwing out those observations that differ from the first-guess (or forecast) by a large multiple of the observation error. Because we use synthetic observations (in an identical-twin experiment), such quality control measures are turned off.

It should be noted that the state estimates are a function of the relative difference between error uncertainties of observations and forecasts, and not their absolute values. Although the process for determining an optimal estimate is well established, mechanisms for determining the underlying error covariances are not. Typically, a combination of physical reasoning, statistical exploration and experimentation is used.

A common problem with forecast error covariances is that whilst they have useful correlation structure, the amplitude (or variances) are too small, thus resulting in an unjustifiably strong confidence in a forecast. Observations are either rejected (if quality control is in effect) or have minimal impact on the analysis. *Inflation* is a commonly used method by which the variances are increased to the point that

observations start to yield updates.

Equivalently, and more conveniently, observation errors can be tuned, and this facility is provided with WRF-Var using a method proposed by Desroziers and Ivanov [7]. The utility TUNE included with WRF-VAR calculates an expectation of the second half of Equation 3.16, based on randomly perturbing observations [28]. If there are enough observations, a reliable tuning factor can be calulated at an analysis time to adjust the relative strength of observation and forecast uncertainty. We use this method to optimize observational errors in the Katrina experiment.

3.4.2 First Guess & Background Error Statistics

WRF-Var uses a *wrf_input* format file for the first guess, which has different variables than those used by WRF as an initial condition. In cold start, a file in *wrf_input* format is created by REAL. For warm start, an option is added to the main WRF namelist for WRF to output a forecast file of the necessary format. WRF-Var overwrites the *wrf_input* file with the analysis result (thus completing the cycle).

Climatological background error covariances are included as the default with WRF-Var for North America during winter, but errors for the correct season and resolution can be estimated by either the NMC method or the ensemble method using the utility GEN_BE. The NMC method is preferred as it is operationally more prevalent, as well as relatively being computationally inexpensive. For this method at least a month of 24 hour and 12 hour forecasts both ending at the same time in 12 hour intervals are recommended. B is then estimated by

$$B \approx \overline{[X^f(T+24) - X^f(T+12)][X^f(T+24) - X^f(T+12)]^T} \tag{3.17}$$

where the overbar represents a long term average.

Implementation of this method begins with a cumbersome logistical step. Accumulated forecast files are split with a utility called TIME_SPLIT to extract the forecast at a single time, the time at which the simulation ends. The files are then ready to be ingested by the first stage of GEN_BE.

GEN_BE is made up of 5 separate stages run by *cshell* script. The first stage, STAGE0 reads in forecast files starting at lagged initial time (12h) but valid at the same forecast time. The fields are converted to WRF-Var fields and then differences are calculated. These difference fields are used to calculate differences for stream-function and velocity potential based on the U and V difference fields by solving the Poisson equation. Difference fields are outputted into binary files named diff.*date* to be read by the rest of the stages.

STAGE1 bins the difference data based on latitude of the Y-direction and removes bias from the difference fields [26]. STAGE2 performs a regression against the stream-function (by bin) in order to define unbalanced U, V, T, and P. STAGE3 calculates vertical local and global eigenvalues and eigenvectors for the control variables using an eigendecomposition. For regional applications like the Katrina example, STAGE4 calculates horizontal length scales by processing difference data as a function of grid-point separation and least-squares fitting this curve to a Gaussian. All mentioned data are combined into a single background statistic file called *be.cv_5* (where 5 stands for regional statistics calculated from WRF fields) by a utility called DAIGS. The same background statistics file is used for all analysis times.

3.5 WRF-Var Code Edits

There were numerous code alterations done in setting up WRF-Var both for the purposes of fixing apparent bugs in the source code and making WRF-Var function better on *ecovision.mit.edu*.

Compilation errors were fixed in both MAP (a utility to view the location of ingested observations) by making the following change to the */3DVAR_OBSPROC/configure.user* file lines 61-62:

```
LIBS = -L$(NCARG_ROOT)/lib -lncarg -lcgm -lncarg_gks -lncarg_c       \
       -L$(PGI)/linux86/lib -lpgftnrtl -lpgc -lX11 -lgfortran \
```

changed to:

```
LIBS = -L$(NCARG_ROOT)/lib -lncarg -lcgm -lncarg_gks -lncarg_c      \
       -L/usr/X11R6/lib -lX11 -L$(PGI)/linux86/lib -lpgftnrtl -lpgc \
       -L/usr/lib64 -lg2c -L/usr/lib64 lgfortran
```

The utility TUNE for tuning observational errors had a runtime error for calculating the expected J_o over multiple time periods, so the associated DO loops were removed from line 219, 345, 526, 679, and 1219.

Another change was made in */wrfvar_v2.2beta/da_3dvar/src/DA_Obs/da_use_ob_errfac.inc* to specify that only U and V errors were to be tuned. Lines 131-141 (below) commented out:

```
iv % sound(n) % v(k) % error = iv % sound(n) % v(k) % error * &
                                              iv % sound_ef_v
iv % sound(n) % t(k) % error = iv % sound(n) % t(k) % error * &
                                              iv % sound_ef_t
iv % sound(n) % q(k) % error = iv % sound(n) % q(k) % error * &
                                              iv % sound_ef_q
iv % sonde_sfc(n) % u % error = iv % sonde_sfc(n) % u % error * &
                                                 iv % synop_ef_u
iv % sonde_sfc(n) % v % error = iv % sonde_sfc(n) % v % error * &
                                                 iv % synop_ef_v
iv % sonde_sfc(n) % t % error = iv % sonde_sfc(n) % t % error * &
                                                 iv % synop_ef_t
iv % sonde_sfc(n) % p % error = iv % sonde_sfc(n) % p % error * &
                                                 iv % synop_ef_p
iv % sonde_sfc(n) % q % error = iv % sonde_sfc(n) % q % error * &
                                                 iv % synop_ef_q
```

It was also ambiguous where the errfac.dat file was to be located, so the statement in */wrfvar_v2.2beta/da_3dvar/src/DA_Obs/da_read_errfac.inc* on line 25:

```
open( fac_unit, status='old', file = 'errfac.dat', iostat=ierr )
```

was changed to:

```
open( fac_unit, status='old', file = &
'/home/align/wrf_new/wrfvar_v2.2beta/gen_be/errfac.dat', iostat=ierr )
```

The other errors were all associated with GEN_BE. A compilation issue is solved for gen_be_stage0 with this change in */wrfvar_v2.2beta/external/io_int/makefile* on line 4:

```
FFLAGS  = $(FCFLAGS)
```

changed to:

```
FFLAGS  = -Mpreprocess $(FCFLAGS)
SFC     = pgf90
```

A runtime issue encountered was that STAGE0 defines the character strings as only length 80 to hold the names of WRF forecast inputs, so the inputs are now located in a root directory of shorter length. STAGE0 also tried to open the same forecast file twice instead of the appropriate pair, resulting in difference files filled with 0 values. The way in which STAGE0 generated the file names was changed in the file */wrfvar_v2.2beta/da_3dvar/src/DA_Gen_Be_Stats/DA_Statistics_Step0.inc* to call for the appropriate file names. Line 86:

```
call DA_Make_Filename(file_date1, fcst_date, number, filename)
```

was changed to:

```
call DA_Make_Filename(file_date1, file_date1, number, filename)
```

while line 112:

```
      call DA_Write_Diff (unit, file_date, number, diff, xp, &
                          ids,ide, jds,jde, kds,kde,  &
                          ims,ime, jms,jme, kms,kme,  &
```

was changed to:

```
call DA_Write_Diff (unit, file_date1, number, diff, xp, &
                    ids,ide, jds,jde, kds,kde,  &
                    ims,ime, jms,jme, kms,kme,  &
                    its,ite, jts,jte, kts,kte, xb12 )
```

The same file was edited in experimentation with other methods of generating background errors. Additionally because STAGE0 uses many of the same framework as WRF-Var, there were some instances where WRF-Var inputs had to be supplied just for runtime to go smoothly. An "ANALYSIS_DATE" variable had to be defined in */wrfvar_v2.2beta/da_3dvar/src/DA_Setup_Structures/DA_setup_firstguess_wrf_var.inc* for STAGE0 even though no analysis was being calculated. Line 50 had this added only for compiling gen_be, not wrfvar:

```
an_date=xb_date
```

The program also would not run unless a WRF 3DVAR input file was linked, even though it was not used. Another change made to */wrfvar_v2.2beta/da_3dvar/Module_wrf_3dvar_io.F* so that the proper date would be passed to the difference file being outputted. The following was added to line 78:

```
current_date=in_date
```

STAGE1 contained an error where the difference files for input were not properly declared as old files, and so they were being overwritten instead of opened. The appropriate OPEN statements were thus edited in */wrfvar_v2.2beta/gen_be/gen_be_stage1.f.* Line 121 and 248 changed from:

```
open (iunit, file = trim(dat_dir)//'/'//filename, form='unformatted')
```

to:

```
open (iunit, file = trim(dat_dir)//'/'//filename, &
              form='unformatted', status='old')
```

Finally, OPEN statements were changed in each of the 4 stages (in directory */wrfvar_v2.2beta/gen_be/)* to accept difference files from any date, not just in temporal

succession as was allowed before. A logical variable "johnfile" was declared and the following statements were used every time a file was opened:

```
inquire(file=trim(dat_dir)//'/'//filename,exist=johnfile)
if (johnfile) then
     -program continues-
end
```

Chapter 4

Identical Twin Experiment

As stated in the Chapter 1 introduction, the WRF-Var system provides a platform for experimentation in data assimilation with different mesoscale problems. Here we work through the process by simulating an important and common problem, a position error in a tropical cyclone forecast.

The framework utilized is an identical-twin experiment with a simulated observation network. Each twin is a simulation of Hurricane Katrina. One, however, has a constant, barotropic 2 m/s westerly wind imposed to simulate a perturbation run relative to what may be viewed as the *true* Hurricane Katrina simulation. The "truth" is used as a source of observations (after adding noise).

We take this approach because identical twin experiments are accepted as a valid source for methodological development and testing in data assimilation. Further, although Katrina itself was well-forecasted, we are interested in examining the effect position errors has on DA and this is achieved by comparing the perturbed run to the simulated.

In this chapter, we discuss the generation of appropriate background error statistics, creation of synthetic observations, and the mechanics of running the identical twin experiment. We hope to illustrate both the opportunity for experimentation now afforded by a working state-of-the-art assimilation package at MIT and the failing of 3DVAR in regards to position errors (as mentioned in Section 1.1). We also show a field alignment solution for WRF fields from the identical twin experiment. The

encouraging results imply that this algorithm can be implemented into an already existing data assimilation package such as WRF-Var.

4.1 Background Errors

The background errors are perhaps the most important component of 3DVAR, as they are used to spread the influence of observations on surrounding model grid nodes and impose balance constraints. For purpose of optimizing the performance of WRF-Var in this experiment, background errors are generated from 4 years of past August forecasts (2004-2007) using the same domain as the Katrina simulation. The domain of study is shown in Figure 4-1.

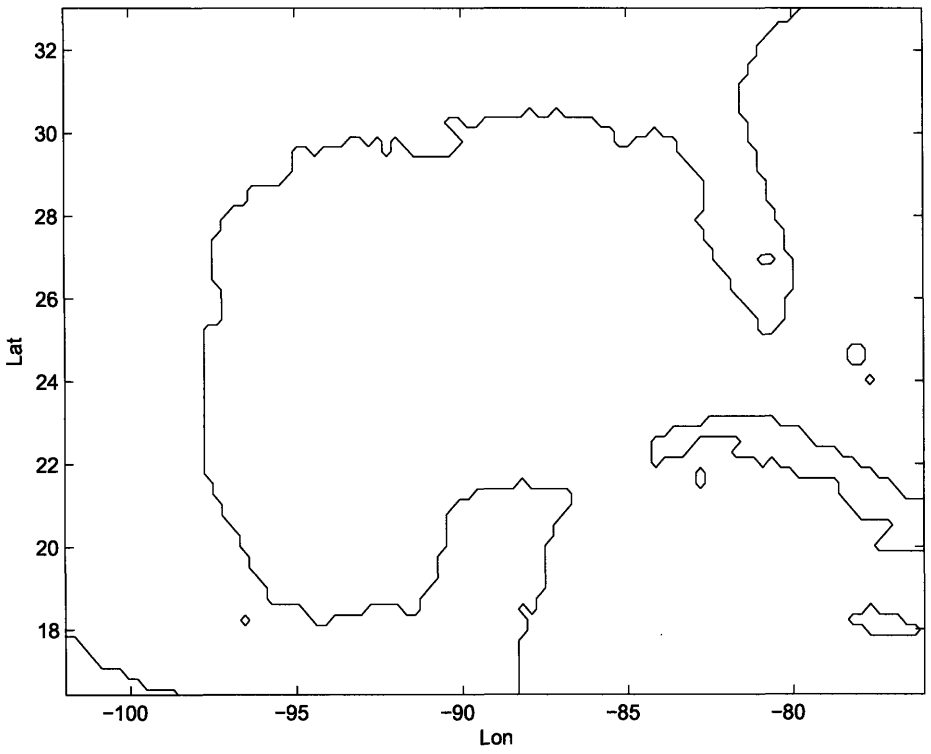

Figure 4-1: Domain for Katrina simulation. SW to NE corner is $16.5^oN, 102^oW$ to $33^oN, -76^oW$, center point is $25^oN, 89^oW$.

The Katrina simulation itself takes place over 2 days (August 28 and 29). Flow

dependent statistics for this *particular* storm would be optimal, but because we do not implement an ensemble forecast here, and the inappropriateness of state-dependent covariances in the presence of position errors is already obvious [6], we utilize a plausibly more robust source of background error covariance by generating climatologically relevant errors using the NMC method from past Augusts.

The NMC method is applied to pairs of forecast files ending at the same time as described in Section 3.4.2. For creating difference files, 12 and 24 hour forecast files every 12 hours are archived for each August of years 2004-2007. These files are WRF forecasts using AVN grib data for initialization and boundary conditions.

Difference files are used to calculate background statistics. In order to obtain statistically meaningful results, a sample size of 50 or more difference files is fed into a WRF program called GEN_BE. Samples are chosen randomly from the 4 years and the background statistics file is used for every assimilation time in the experiment.

A way of distilling some relevant information from all of the NMC characterization of background uncertainty is by looking at the eigenvectors and eigenvalues of covariance components in the vertical. Eigenvectors are a look into the vertical structure of modes of oscillation for model variables. Eigenvalues for these eigenvectors, shown in Figure 4-2 for control variables are a representation of the variance for each of these vertical modes. Thus, these variances can be compared roughly at a glance to the error variances for observations (once converting to the same variable).

As an example, taking a value of 1.8×10^{11} for a mode 1 eigenvalue of velocity potential from Figure 4-2, an order of magnitude approximation for the variance in U is found through $U = \frac{\partial}{\partial x}\phi$ where $\partial x = 30$ km and $\phi = \sqrt{1.8 \times 10^{11}}$. This comes out to a variance in U of around 10 m/s. The observation errors are of comparable scale, around 3 m/s. Furthermore, the similarity in eigenvalues from both the 50 sample and full case show that the subsample provides a reasonable representation of the the background statistics.

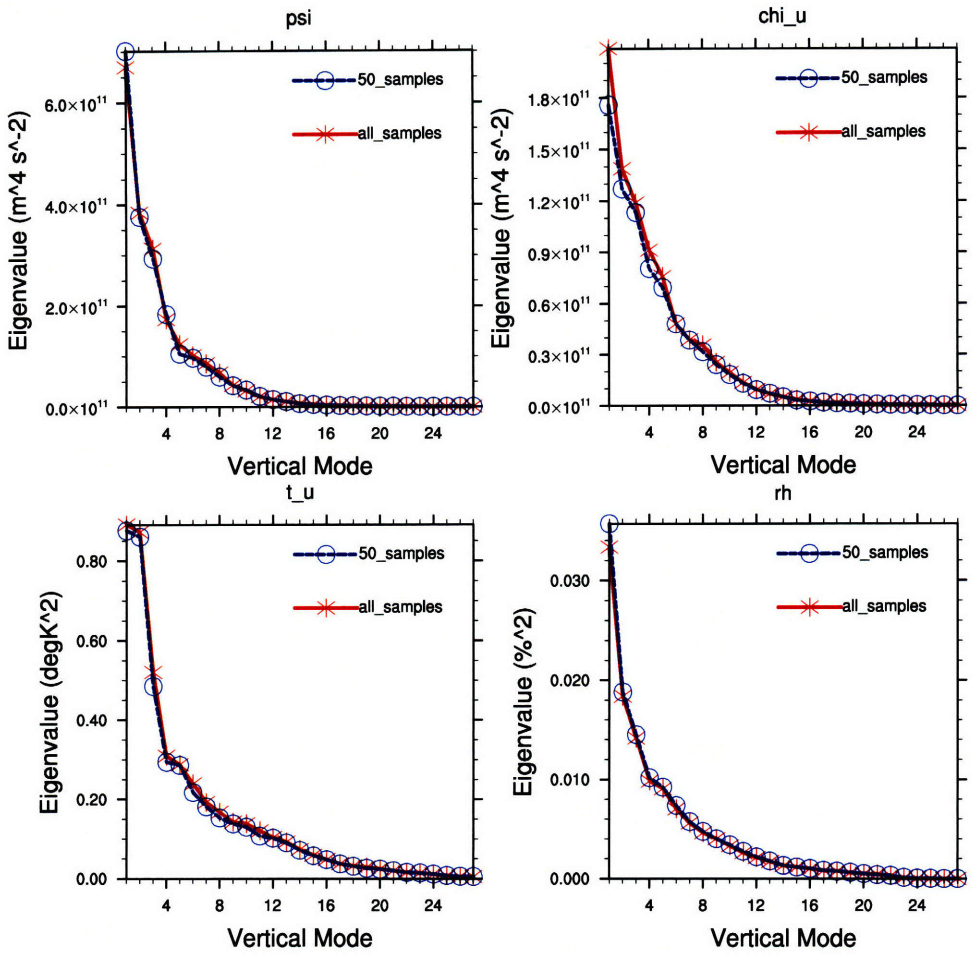

Figure 4-2: Eigenvalues for WRF-Var control variables from NMC method

4.1.1 Analysis Increment

For understanding the spatial structure of the background error covariance estimated by NMC at any level, a single sounding observation is assimilated into a perturbed run. The sounding is located in the center of the domain as shown in Figure 4-3.

Surface pressure is plotted for truth to show the observation in relation to Katrina. The resulting analysis increment from WRF-Var is shown in Figure 4-4.

The influence of the observation is consistent with other efforts and reasonable in the analysis when taking into account that the U field has been balanced geostrophically and cyclostrophically in WRF-Var. This gives credence to the background error method and the minimization of the cost function.

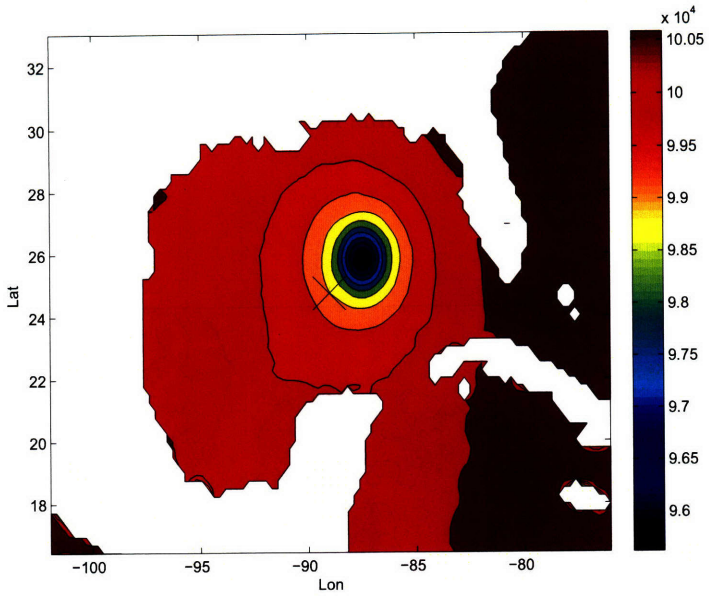

Figure 4-3: Single observation location for analysis increment. Surface pressure (PA) shown to indicate relative position of Katrina at 12 hours

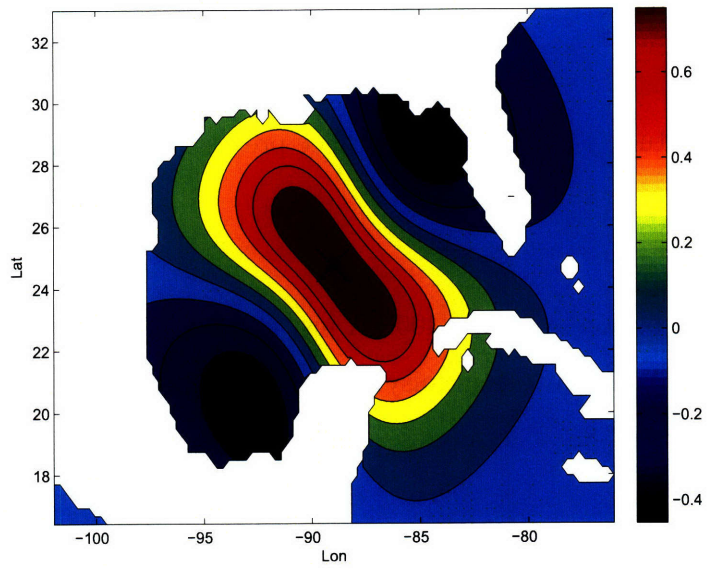

Figure 4-4: Analysis increment (analysis minus forecast) of U (m/s) for single observation

4.1.2 Amplitude Perturbation

Another test of the WRF-Var system is an amplitude perturbation of an atmospheric variable, as 3DVAR is expected to perform well for amplitude errors. Surface pressure

is perturbed (200 PA is added) from the initial condition file at the domain center point. After a forecast run is completed the perturbation has grown in amplitude and area. The difference between the two runs at 12 hours is shown in Figure 4-5.

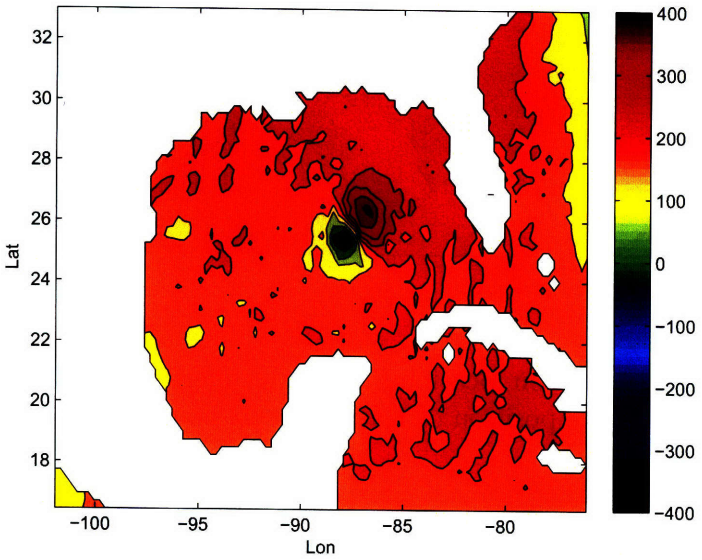

Figure 4-5: Amplitude perturbation in surface pressure (PA) (forecast minus truth)

Now the perturbation is around 400 PA just north-east of the center. An analysis is produced from WRF-Var using noisy U and V component observations from the same grid as Figure 4-8. The analysis increment in surface pressure is shown in Figure 4-6.

The analysis reduces the forecast run perturbation by about 200 PA, or half. The analysis compared to the real run is shown in Figure 4-7.

The analysis shows that WRF-Var finds a solution midway between truth and the forecast state. This is the expected result for similar confidence (error variances) in observation and background states, as discussed in Section 3.4.1. Thus this example gives confidence to the WRF-Var system as we have configured it.

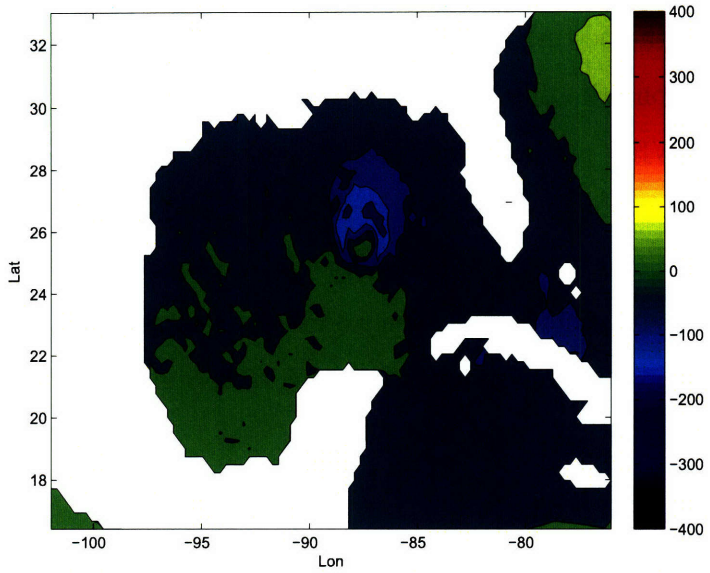

Figure 4-6: Surface pressure analysis increment (PA) for amplitude perturbation example

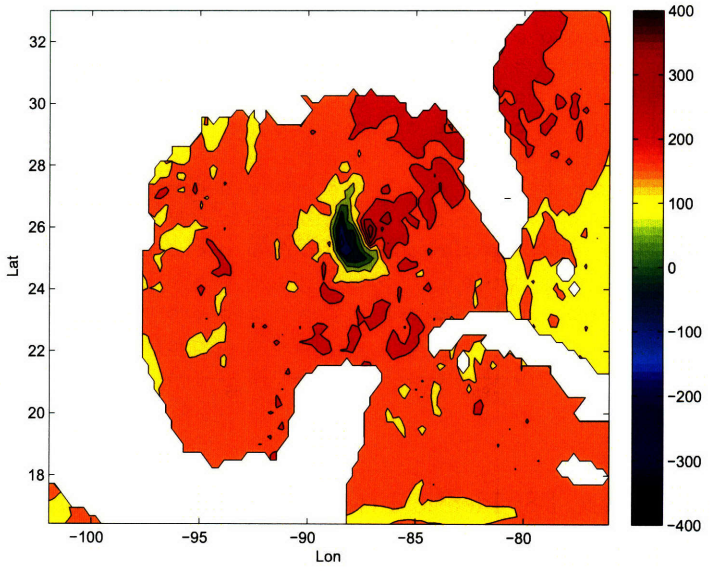

Figure 4-7: Analysis minus truth of surface pressure (PA) for amplitude perturbation example

4.2 Perturbation Run

Here, in contrast, the perturbed run is designed to create a position error. A 2 m/s barotropic wind is added to the U component at every grid point to "blow" Katrina

off course eastward. This change is entered by a subroutine at the end of the WPS procedure. Thus, the perturbation is included in the initialization file and also exists in the boundary condition files to sustain the forcing at all times.

Because the WRF preprocessing program REAL assumes atmospheric data is balanced it does not impose any balance itself. Thus, we use another subroutine to add a perturbation to the geopotential height to geostrophically balance the wind perturbation. This prevents low frequency oscillations from dominating the movement of Katrina and keeps the simulation from being nonphysical.

4.3 Observations

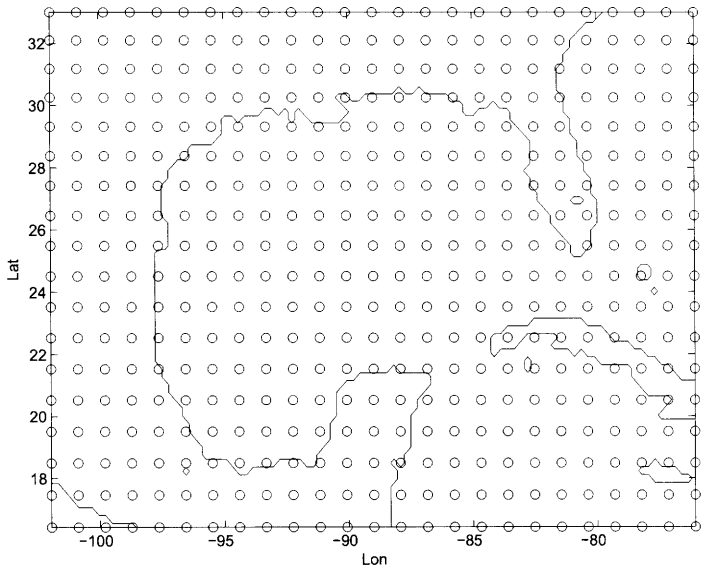

Figure 4-8: Identical twin observation locations

Observations are taken from the 2-day simulation at every 4^{th} gridpoint as shown in Figure 4-8. The first assimilation cycle for the identical twin is at 12 hours to allow for damping of transients from the initialization process. Subsequent assimilations are performed from observations at 6 hour intervals. At these times, a matlab script gathers data from the *netcdf* output and puts the data into *little_r* format. The observations are then processed by OBSPROC for ingestion into WRF-Var. Random noise

is added by selecting the "add observation noise" option in WRF-Var. Observation errors are chosen as the standard sounding errors, and no observations are thrown out for being too far away from the average state. Observation error statistics can be tuned by using TUNE.f90 at the first assimilation time. Details of the identical twin materials and scripts are found in appendix A.

4.4 WRF-Var Twin Analysis

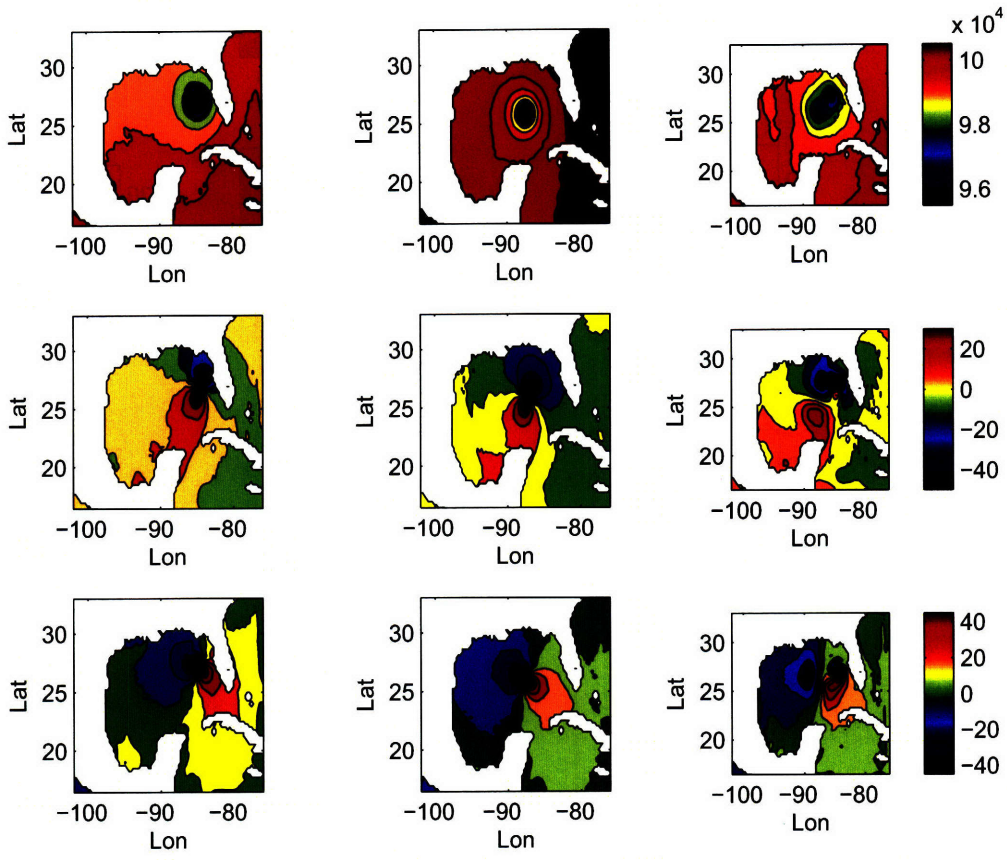

Figure 4-9: Identical twin plots. Rows: surface pressure (PA), U component near 900 mbar (m/s), V component near 900 mbar (m/s). Columns: forecast, truth, analysis

Forecast, truth, and analysis fields for the identical twin experiment are shown in Figure 4-9 (original observation errors) and Figure 4-10 (observation errors scaled by

a factor of about 2.5). The forecast is run for 12 hours with a 2 m/s perturbation in U component. By this time the Katrina forecast shows a position error eastward of about half the diameter of the cyclone's surface pressure signature. The background surface pressure is generally different for the forecast because of the geostrophic balance imposed along with the wind perturbation.

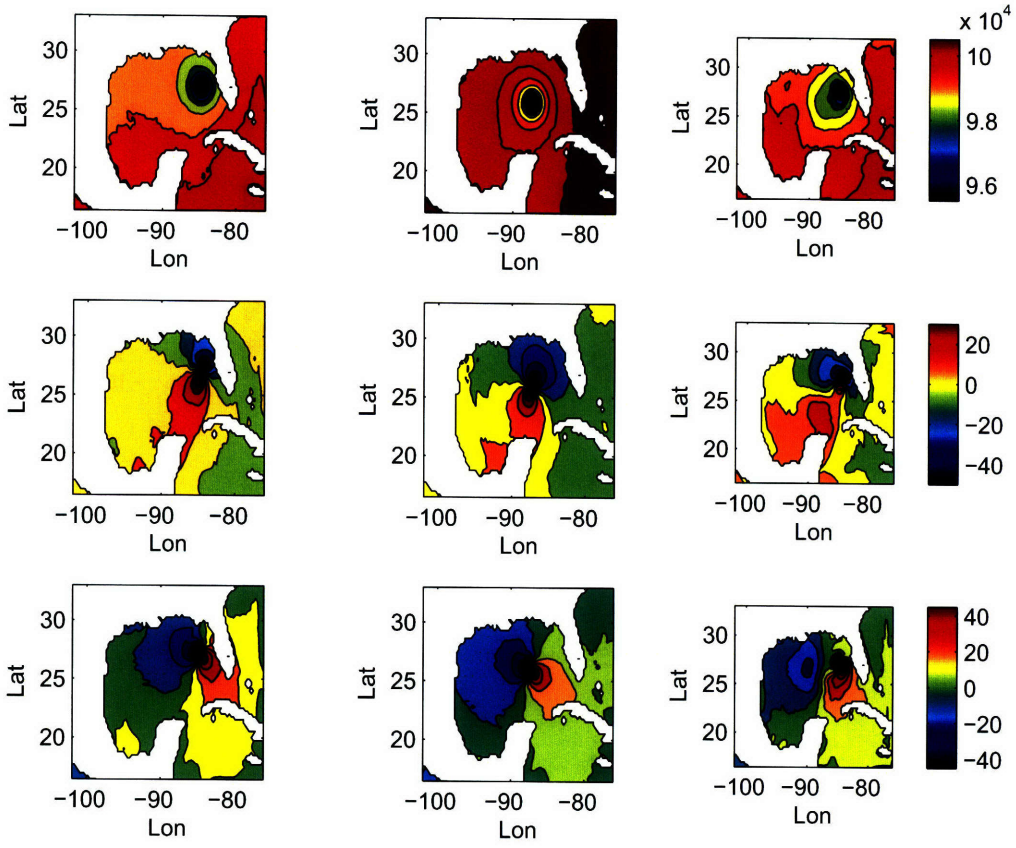

Figure 4-10: Same as Figure 4-9 except with tuned observation errors.

The analysis in the U and V fields is smeared between the forecast and truth for both cases.[1] The tuning factor of 2.5 multiplied by all observation errors in the second example results in a higher confidence in the forecast, but this obviously does nothing to make the analysis more reasonable. The resulting surface pressure for both shows

[1] "Analysis" pressure is actually pressure calculated at 1 model time step (180s) as there is no pressure increment and the wind analysis increments are not balanced in surface pressure.

a wider and weaker tropical cyclone. It is clear from this experiment why methods such as bogussing have been developed.

4.5 Field Alignment Description

Field alignment, as discussed in Ravela et al. [21], allows for position adjustments in addition to amplitude adjustments. The goal is to create a cost function that explicitly represents position errors. The new objective is written

$$\begin{aligned} J(X,\mathbf{q}) &= \frac{1}{2}(X(\mathbf{p}) - X^f(\mathbf{p}^T B(\mathbf{q})^{-1}(X(\mathbf{p}) - X^f(\mathbf{p})))) \\ &\quad + \frac{1}{2}(Y - HX(\mathbf{p}))^T R^{-1}(Y - HX(\mathbf{p})) \\ &\quad + L(\mathbf{q}) - \frac{1}{2}ln(|B(\mathbf{q})|) \end{aligned} \tag{4.1}$$

where $\mathbf{q}$ is the displacement field, $X(\mathbf{p} := \mathbf{r} - \mathbf{q})$ represents displacement of X by $\mathbf{q}$ ($\mathbf{r}$ are the position indices), and $L(\mathbf{q})$ is an energy function which expresses constraints on the displacement field.

Unfortunately, Equation 4.1 is highly nonlinear and although a direct solution using an iterative minimization technique can be implemented [21], approximations yield a more practical solution.

Instead of solving displacements and amplitudes simultaneously through Equation 4.1, an approximation involving the Euler-Lagrange equations can be made to solve them sequentially. This cost function is minimized in two steps, the first is a position adjustment step and the second is an amplitude adjustment step. The displacement equation is

$$w_1 \nabla^2 \underline{q}_i + w_2 \nabla(\nabla \cdot \underline{q}_i) + \left[\nabla X^{fT}|_{\mathbf{p}} H^T R^{-1} \left(H\left[X^f(\mathbf{p})\right] - Y\right)\right]_i = 0 \tag{4.2}$$

Here q_i is the displacement at grid node i. Equation 4.2 is solved iteratively, and a poisson equation is solved at each iteration to refine the displacement estimate.

The amplitude equation is applied after solving Equation 4.2. Let us define the

thus displaced grid nodes as $\hat{\mathbf{p}}$. Then, the amplitude equation may be written as:

$$X^a = X^f(\hat{\mathbf{p}}) + BH^T(HBH^T + R)^{-1}\left(Y - H\,X^f(\hat{\mathbf{p}})\right) \tag{4.3}$$

Notice that the background error covariance is the covariance after alignment. In our case, because B is climatologically determined, it is unchanged as a resulf of alignment and used as is.

The amplitude adjustment step is 3DVAR, as discussed, and here we focus on the position adjustment step and integrate it with MIT's WRF-Var framework.

The positions of all fields are adjusted directly in model space. Note that all scalar and vector model fields are aligned using the observations to define a single displacement vector at each node [21]. Figure 4-11 shows the analysis by field alignment, depicted here for surface pressure. Now, instead of feature distortion, there is a new position for Hurricane Katrina between truth and forecast.

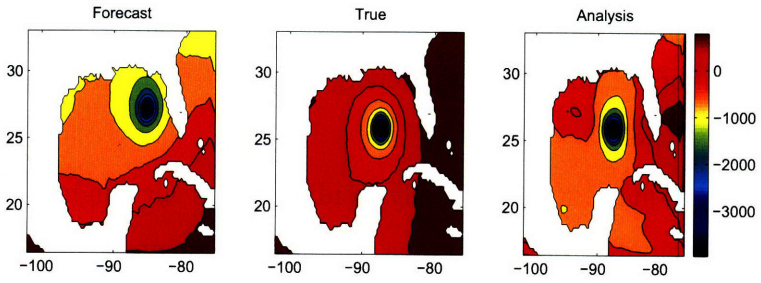

Figure 4-11: Application of field alignment algorithm. Perturbation pressure (PA) is shown.

Chapter 5

Conclusions

A fully working version of the WRF-Var system is now available for data assimilation experimentation with both real and idealized cases. This is a state-of-the-art mesoscale forecast model, the first available at MIT. Along with the ability to now experiment with currently used DA methods, real-time forecasting is also possible with addition of real data inputs.

A simple position error identical twin experiment using the system shows a clear weakness with traditional amplitude adjustment methods like 3DVAR. Implementation of the two step field alignment algorithm appears a very promising solution as shown with the alignment of WRF variable fields from the identical twin.

Appendix A

Identical-Twin Experiment Mechanics

All materials for running the twin experiment including input data, edited code, namelists, and configuration files are found on *ecovision.mit.edu* in the directory */home/align/wrf_new/auto_run_twin*. The *run* folder contains *cshell* scripts for the single observation and amplitude adjustment experiment, tuning observation errors, creating WRF forecasts for background statistics, and the identical twin experiment. The script *twin.csh* runs through all the steps for the identical twin, which includes configuration and compilation of code (if fresh WRF code is being used). For the twin itself, start time, number of cycles, and length of time between cycles are defined first. A forecast is then run for the perturbed case until the first analysis time. Then observations are taken from the real run for that time and WRF-Var is run to produce an analysis. After updating the boundary conditions, the next cycle runs another forecast starting from the analysis and the assimilation cycle repeats. *Netcdf* output from each forecast run and each assimilation is outputted into a single directory. This master script gives the ability to perform different experiments quickly, a monumental improvement over trudging through the individual WRF steps manually.

Edited WRF code is located in the *code* folder, along with the matlab script *obs.m* for creating *little_r* format observation files from Katrina truth and *randfile.c* for generating a random difference file names for background statistics. Static land

data, GFS analysis data for Augusts of 2004-2007, and SST data for the same time period are found in *data*. Namelists for creating forecasts for background statistics and configuration files for compiling WRF, WRF-Var, and GEN_BE are found in *namelists*. Other namelists for use in experiments are created by *twin.csh* as it runs. Finally output from the identical twin experiment is moved by *twin.csh* into the folder *output*.

Bibliography

[1] The wmo manual on codes, section: A guide to the code form fm 92-ix ext. grib. WMO Publication No. 306, 1995.

[2] G. Alexander, J Weinman, and J Schols. The use of digital warping of microwave integrated water vapor imagery to improve forecasts of marine extratropical cycles. *Monthly Weather Review*, 126:1469–1495, 1998.

[3] A. Bennett. *Inverse Methods in Physical Oceanography.* Cambridge University Press, 1992.

[4] K. Brewster. Phase-correcting data assimilation and application to storm-scale numerical weather prediction. part i: Method description and simulation testing. *Monthly Weather Review*, 131:480–492, 2003.

[5] A. Bryson and Y. Ho. *Applied Optimal Control: Optimization, Estimation, and Control.* Hemisphere Publishing Corporation, 1975.

[6] R. Dee and A. da Silva. Data assimilation in the presence of forecast bias. *Quarterly Journal of the Royal Meteorological Society*, 124(545):269–296, 1998.

[7] G. Desroziers and S. Ivanov. Diagnosis and adaptive tuning of observation-error parameters in a variantional assimilation. *Quarterly Journal of the Royal Meteorological Society*, 127:1433–1452, 2001.

[8] G. Evensen. The ensemble kalman filter: Theoretical formulation and practice implementation. *Ocean Dynamics*, 53:342–367, 2003.

[9] G. Evensen and P. Van Leeuwen. An ensemble kalman smoother for nonlinear dynamics. *Monthly Weather Review*, 128:1852–1867, 2000.

[10] A. Gelb. *Applied Optimal Estimation.* The MIT Press, 1974.

[11] R. Hoffman and C. Grassotti. A technique for assimilating ssm/i observations of marine atmospheric storms: Tests with ecmwf analyses. *Journal of Applied Meteorology*, 35:1177–1188, 1996.

[12] R. Hoffman, Z. Liu, J. Louis, and C. Grassotti. Distortion representation of forecast errors. *Monthly Weather Review*, 123:2758–2770, 1995.

[13] K. Ide and M. Ghil. Extended kalman filtering for vortex systems. part i: Methodology and point vortices. *Dynamics of Atmospheres and Oceans*, 27:301–332, 1997.

[14] C. Jones and B. MacPherson. A latent heat nudging scheme for assimilation of precipitation data into an operational mesoscale model. *Meteorological Applications*, pages 269–277, 1997.

[15] E. Kalnay. *Atmospheric modeling, in: Data Assimilation and Predictability.* Cambridge University Press, 2003.

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

[17] E. Lorenz. Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, 20(2):130–141, 1963.

[18] A. Mariano. Contour analysis: A new approach for melding geophysical fluids. *Journal of Atmospheric and Oceanic Technology*, 7:285–295, 1990.

[19] D. Parrish and J. Derber. The national meteorological center's spectral statistical-interpolation analysis system. *Monthly Weather Review*, 120:1747–1763, 1992.

[20] H. Rauch. Solutions to the linear smoothing problem. *IEEE Transactions on Automatic Control*, 8:371–372, 1963.

[21] S. Ravela, K. Emanuel, and D. McLaughlin. Data assimilation by field alignment. *Physica(D)*, 230:127–145, 2007.

[22] S. Ravela, J. Marshall, C. Hill, A. Wong, and S. Stransky. Realtime observatory for laboratory simulation of planetary flows. *Experiments in Fluids (under review).*

[23] S. Ravela, J. Marshall, C. Hill, A. Wong, and S. Stransky. Real-time observatory for laboratory simulation of planetary circulation. In Y. Shi, G. Albada, J. Dongarra, and P. Sloot, editors, *Computational Science - ICCS 2007, Part I*, number 4487 in Lecture Notes in Computer Science, pages 1155–1162, Beijing, China, May 2007. ICCS, Springer.

[24] S. Ravela, J. Marshall, C. Hill, A. Wong, and S. Stransky. Tracking rotating fluids in realtime using snapshots. In *Proceedings of IEEE Computer Vision and Pattern Recognition (article in press)*, 2008.

[25] R. Rew, G. Davis, S. Emmerson, and H. Davies. Netcdf user's guide for c, an interface for data access, version 3. 1997.

[26] W. Skamarock, J. Klemp, J. Dudhia, D. Gill, D. Barker, W. Wang, and J. Powers. A description of the advanced research wrf version 2. 2007.

[27] H. Thiebaux, P. Julian, and G. DiMego. Areal versus collocation data quality control. In *Preprints*, International Symposium on Assimilation of Observations in Meteorology and Oceanography, pages 255–260, Clermont-Ferrand, France, 1990. WMO.

[28] W. Wang, D. Barker, J. Bray, C. Bruyere, M. Duda, J. Dudhia, D. Gill, and J. Michalakes. User's guide for advanced research wrf (arw) modeling system version 2.2. 2007.

[29] C. Wunsch. *The Ocean Circulation Inverse Problem.* Cambridge University Press, 1996.

[30] Q. Xiao, Y. Kuo, Y. Zhang, D. Barker, and D. Won. Experiments of a typhoon bogussing scheme in the mm5 3d-var cycling system. Number 15C.4 in 26th Conference on Hurricanes and Tropical Meteorology, Miami, FL, 2004. AMS.