5-2021

The Effect of Initial Conditions on the Weather Research and Forecasting Model

Aaron D. Baker
Stephen F Austin State University, bakeraarond@gmail.com

The Effect of Initial Conditions on the Weather Research and Forecasting Model

Creative Commons License

THE EFFECT OF INITIAL CONDITIONS ON THE WEATHER RESEARCH AND FORECASTING MODEL

by

Aaron Baker, B.S.

Presented to the Faculty of the Graduate School of

Stephen F. Austin State University

In Partial Fulfillment

of the Requirements

For the Degree of

Master of Science

STEPHEN F. AUSTIN STATE UNIVERSITY

May 2021

THE EFFECT OF INITIAL CONDITIONS ON THE WEATHER RESEARCH AND FORECASTING MODEL

by

Aaron Baker, B.S.

APPROVED:

Lynn Greenleaf, Ph.D., Thesis Director

Jeremy Becnel, Ph.D., Committee Member

Christopher Ivancic, Ph.D., Committee Member

Jacob Turner, Ph.D., Committee Member

Pauline M. Sampson, Ph.D.
Dean of Research and Graduate Studies

ABSTRACT

Modeling our atmosphere and determining forecasts using numerical methods has been a challenge since the early 20th Century. Most models use a complex dynamical system of equations that prove difficult to solve by hand as they are chaotic by nature. When computer systems became more widely adopted and available, approximating the solution of these equations, numerically, became easier as computational power increased. This advancement in computing has caused numerous weather models to be created and implemented across the world. However a challenge of approximating these solutions accurately still exists as each model have varying set of equations and variables to approximate.

The model under consideration in this paper is the Weather Research and Forecast model (WRF). This model takes input from data files as initial conditions, often from other models, and runs a simulation based on its own set of equations and conditions.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my thesis director, Dr. Lynn Greenleaf, for her support in my learning the mathematics behind meteorology since being an undergraduate student at SFA and continuing on graduate school.

Next, I would like to thank my committee members, Dr. Jeremy Becnel, Dr. Jacob Turner, and Dr. Christopher Ivancic. My project included subjects that were directly related to their fields, and they helped and guided me along the way.

To my parents for their support during my time at SFA. I cannot thank them enough for putting up with me and supporting me through my crazy ideas and plans, even if it meant staying in school for longer than expected.

I also want to thank my family at the Center for Teaching & Learning at SFA for putting up with me with the last three and a half years, and especially when I started working on my project. The entire staff in the department has all supported me in my endeavors and I cannot thank them enough.

To my friends that I've gotten to know during my time at SFA and the ones that I've known since high school, my accomplishments would not have been possible without their continuing support.

Lastly, a special thanks to meteorologists Chris Gibson, the Science and Operations Officer at the Missoula, Montana Weather Forecast Office and Armani Cassel, Meteorologist at the Shreveport, Louisiana Weather Forecast Office. I would not have been able to accomplish the project without their help and the resources they provided.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

Modeling the physical world involves the use of partial differential equations. This can usually be found when modeling fluid flows, electricity, mechanics, and optics. These equations are useful because most laws of physics, like Newton's equations of motion and the Navier-Stokes equations, can be described by relating to space and time derivatives. These derivatives represent natural things like velocity and acceleration and the relationship between the rate of change of these natural things can be used to study the world around us [6].

1.1 Motivation

Meteorology is a subset of the physical world that pertains to the fluid flow of the atmosphere. When it comes to forecasting the weather, the atmosphere is modeled by a system of partial differential equations that are then approximated to estimate the value of certain unknown variables given a set of boundary and initial conditions. The purpose of this paper is to analyze one of these models and to compare and contrast the output of various simulations based on the different initial conditions the model was given.

1.2 Introduction to Partial Differential Equations

A partial differential equation (PDE) is an equation containing partial derivatives. Unlike ordinary differential equations, the unknown function of the PDE depends on multiple variables such as time and space.

Some well-known PDEs include:

$$u_t = u_{xx} \qquad \text{heat equation in one dimension} \tag{1.1}$$

$$u_{tt} = u_{xx} + u_{yy} + u_{zz} \qquad \text{wave equation in three dimensions} \tag{1.2}$$

$$u_{tt} = u_{xx} + \alpha u_t + \beta u \qquad \text{telegraph equation} \tag{1.3}$$

In the examples above, the variable u is the dependent function and the function we use to differentiate. The variables x, y, z, and t are the independent variables we use to differentiate u with respect to. For instance, $u_t = \frac{\partial u}{\partial t}$ is the partial derivative of u with respect to t.

The order of a partial derivative in a PDE is determined by the maximum number of terms listed in the subscript of the variable. For example,

$$u_t = \frac{\partial u}{\partial t} \qquad \text{first order} \tag{1.4}$$

$$u_{tt} = \frac{\partial^2 u}{\partial t^2} \qquad \text{second order} \tag{1.5}$$

$$u_{ttt} = \frac{\partial^3 u}{\partial t^3} \qquad \text{third order} \tag{1.6}$$

Likewise, the order of the PDE is the order of the highest partial derivative in the PDE itself ($u_t = u_{xx}$ is a second order PDE since the order of u_{xx} is two).

Another classification of PDEs is the linearity. A PDE is considered to be linear if the dependent function, u and its derivatives appear in a linear fashion and can be modeled in the form of

$$Au_{xx} + Bu_{xy} + Cu_{yy} + Du_x + Eu_y + Fu = G \tag{1.7}$$

where x and y are the independent variables of the equation and A, B, C, D, E, F, and G can be constants or given functions of the independent variables. Some

examples between linear and nonlinear PDEs include:

$$e^{-t}u_{xx} + \sin t = u_{tt} \qquad \text{linear} \tag{1.8}$$

$$uu_{xx} + u_t = 0 \qquad \text{nonlinear} \tag{1.9}$$

$$u_{xx} + yu_{yy} = 0 \qquad \text{linear} \tag{1.10}$$

$$xu_x + yu_y + u^2 = 0 \qquad \text{nonlinear} \tag{1.11}$$

Notice that (1.8) and (1.10) are linear because the order of u in each term is one. Whereas (1.9) and (1.11) are nonlinear due to the uu_{xx} term in (1.9) and the u^2 term in (1.11).

PDEs can also be classified by homogeneity. For instance, if a PDE is modeled similar to (1.7), then it's considered to be homogeneous if the term $G = G(x, y) = 0$, otherwise it's considered to be nonhomogeneous. In equation 1.8, the PDE is considered to be nonhomogeneous since in this case, $G = G(t) = \sin t$ when rearranged to fit the form of (1.7) and t is an independent variable of the equation. Likewise, (1.9) - (1.11) are considered to be homogeneous since $G = 0$.

Determining the correct solution to a PDE or a system of PDEs depend on the various classifications mentioned above. Many analytical solutions have been developed and can be used to find the exact solution. However, due to constraints of boundary and initial conditions, there are times where the analytical solutions cannot be used and therefore the system of PDEs must be solved by finding an approximation numerically. Several methods have been developed to find the closest approximation with minimal error. These methods include finite differences, Monte Carlo, and Runge-Kutta time integration scheme.

2 NUMERICAL WEATHER PREDICTION

Forecasting the weather by means of numerical methods has been one of the most significant scientific advances in the last century. Forecasts prior to these advances were made primarily based on what had been observed in previous years and the forecaster's knowledge of how the weather usually is for that particular time of the year. However, the advances of thermodynamics resulted in the fundamental principles governing the flow of the atmosphere [8].

2.1 Early Advances

In the late 19^{th} century, the move to view the atmosphere as a moving fluid started to come about. In 1890, meteorologist Cleveland Abbe began this recognition by stating "meteorology is essentially the application of hydrodynamics and thermodynamics to the atmosphere" [15] and in fact hoped that through his proposal of taking a mathematical approach to forecasting, someone would take up the task of figuring out a way to do this using either a graphical, analytical, or numerical method [1].

It would not take long for someone to step up to this task and propose the set of equations needed to solve numerically. Norwegian scientist Vilhelm Bjerknes determined that forecasts should be completed in two steps: a diagnostic step and a prognostic step [2]. The diagnostic step would contain the current observed conditions of the atmosphere at the initial time step, while the prognostic step would use the laws of motion to calculate how the state changes over time. This last step was to be taken by assembling a set of seven independent equations, one for each dependent variable that describes the atmosphere: pressure, temperature, density, humidity,

and the three components of velocity. These seven equations consist of the three hydrodynamic equations of motion, also known as the conservation of momentum, the continuity equation, or the conservation of mass, the conservation equation for water mass, the conservation of energy, or the first law of thermodynamics, and the equation of state.

2.2 The First Forecast

Progress wouldn't be made on Bjerknes' equations until 1913 when English Quaker Scientist Lewis Fry Richardson started working at the Meteorological Office. Richardson had learned of Bjerknes' work and decided to attempt a way to solve them, as Richardson was not a fan of the current method of forecasting. During this time, forecasting was done based on previous years' data and was not very accurate.

Richardson was then appointed Superintendent of Eskdalemuir Observatory in Scotland, and during his time there, Richardson began serious work on weather prediction. Richardson took the idea of taking the equations that Bjerknes suggested, and put them together as a system of equations and used his finite-difference method to solve the system. In Richardson's first run, the forecast failed miserably. Richardson notes in his book that in a six hour forecast, the pressure rose about 145 millibars, where, when compared to the actual observations, the pressure remained steady. Richardson then speculated that this error was not caused by errors in the values of the initial conditions, but because of the unnatural distribution of the initial conditions. To fix this, Richardson suggested that the data be smoothed so that the distribution of data became more natural. The solution would be to modify the analysis to restore the balance between the wind fields and pressure, a process called initialization.

2.3 From Human Computers to Machines

While the results of Richardson's forecast were not widely received, they were not forgotten. Mathematician John von Neumann became interested in turbulent fluid flows in the mid 1930s and recognized that advances in this area could be greatly accelerated if there were a readily available means to solve the complex equations for these flows, numerically. A solution to this issue would require very fast automatic computing machinery. Von Neumann then helped develop the design and construction of an electronic computer at the Institute for Advanced Studies (IAS) at Princeton University. The Electronic Computer Project was comprised of four groups, with one of the groups dedicated to meteorology. Von Neumann recognized that weather forecasting would be an ideal problem for the IAS to solve. Aware of Richardson's work, as well as the further work done by Courrant, Friedrichs, and Lewy that had progressed solving PDEs numerically, von Neumann made the next step in the advancement of weather forecasting by creating a formal proposal to the United States Navy requesting financial backing for the establishment of a Meteorology Project that was later accepted. The Meteorological Research Project was established in July, 1946.

Not long after the establishment of the Meteorological Research Project, a Conference on Meteorology was arranged at the IAS to discuss how to implement the equations Richardson used so that the IAS Computer could solve them. However, it was then realized a smaller time step would be needed to minimize the errors and the computational power of the IAS computer would not be sufficient, leaving member Jule Charney to find a way to find a solution.

The proposed solution would find a way to simplify the seven equations developed by Bjerkens and Richardson and eliminate gravity waves. This new system of equations would be known as the quasi-geostrophic system as it would use the barotropic

vorticity equation. By 1950, the group had completed the mathematical analysis needed to design an algorithm to approximate these new equations. Arrangements were made to run the forecast on the Electronic Numerical Integrator and Computer (ENIAC) in Aberdeen, Maryland. The team made four 24-hour forecasts, and each 24 hour integration took about 24 hours of computation allowing the team to just keep pace with the weather.

With the good initial results of the ENIAC computations, interest and expectations began to grow for a operationally useful computer forecast to be implemented. Karl-Heinz Hinkelmann tackled the problem of finding the ideal initial conditions for the primitive equations integrations. Hinklemann proposed that if the right initialization parameters were set, then the errors would decrease and the accuracy would increase. Work on this problem by several people help create the first operational model for Deutscher Wetterdienst, the weather forecasting office for Germany in 1966 with the National Meteorological Center in Washington implementing a multi-level model later that year. These advancements of the models and computing power have helped create multiple types of atmospheric models that are now in use today.

2.4 NWP Today

Today there are many models in use by several meteorological agencies across the world that all have their specific use cases. Some of these include the Global Forecast System (GFS), the North American Mesoscale Forecast System (NAM), The High-Resolution Rapid Refresh (HRRR), the European Centre for Medium-Range Weather Forecasts (ECMWF), and the Weather Research and Forecasting Model (WRF).

The two most well-known models are the GFS and the ECMWF, also known as the American and European models, respectively. These models are used to forecast the weather across the world on a global scale. The NAM and HRRR are called mesoscale

(fine scale) models. These models are useful in determining small weather features like severe thunderstorms, due to the finer grid resolution that these models are able to use compared to the global models [5]. Updates to these models are applied as warranted. For instance, the National Centers for Enviromental Prediction (NCEP) have recently updated the GFS to version 16 in March 2021. Doubling the number of vertical layers from 64 levels to 127, extending the model from the upper stratosphere (about 55 kilometers in height) to the mesopause (about 80 kilometers in height) [7].

3 THE WEATHER RESEARCH AND FORECASTING MODEL

3.1 Introduction

A special model used today is the Weather Research and Forecasting Model (WRF) [12], a community driven weather model that is hosted by the National Center for Atmospheric Research (NCAR) and allows contributions to the model from the active user base worldwide. As the name implies, the WRF is an atmospheric model designed for both research and NWP that has grown to provide specialty capabilities for a range of Earth system prediction applications, such as air chemistry, hydrology, wildland fires, hurricanes, and regional climate [11]. In its lifetime, the community has developed the model to run on various types of computing systems. Ranging from major super computing clusters down to the low power project computer board of the Raspberry Pi [4].

3.2 WRF Development

Development of the WRF started around 1995 when National Centers for Environmental Prediction (NCEP) started to get an interest of developing a nonhydrostatic model for operating in finer scales. The idea of this new model took shape on the premise that there could be a beneficial synergy in a NWP model shared by research and operational groups and to move past the limitations of other models, such as the then widely used Pennsylvania State University - NCAR Mesoscale Model Version 5 (MM5), and create a platform for an extensive research community to develop the needed capabilities that operations could readily exploit. This would then facilitate the "research to operations" development of a model, while leading to sharpened

research and development efforts on identified needs for operation.

The original members of the partnership that formed to develop the WRF were NCAR, the National Oceanic and Atmospheric Administration (NOAA), the U.S. Air Force, the Naval Research Laboratory, the University of Oklahoma, and the Federal Aviation Administration (FAA). The initial focus of development was the model's dynamical solver and the related numerics, and when compared to previous models, the solver became superior in higher-order numerical accuracy and properties of scalar conservation. Next, a software framework was created for the model's dynamics, physics, and the input/output. Work was also done to create the preprocessors, to handle the domain and input preparation, and the data assimilation for the solver. Finally, through all the efforts of the original partners, the first release of the WRF was in December 2000.

3.3 The WRF Solver

The current version of the WRF is version 4 released in 2019. The main solver is known as the Advanced Research WRF (ARW) and is a configuration of the WRF system that features the ARW dynamics solver together with other compatible components to produce a simulation [12]. While smaller version updates have been released, the overall governing equations, boundary and initial conditions remain the same. In this section we will introduce the equations, boundary, and initial conditions as described by the authors of the WRF Version 4 Technical Manual [12].

3.3.1 Governing Equations

The WRF uses equations that are formulated using a terrain-following hydrostatic-pressure vertical coordinate, denoted as η and is defined by:

$$\eta = \frac{p_d - p_t}{p_s - p_t} \tag{3.1}$$

where p_d is the hydrostatic component of the pressure of dry air, and p_s and p_t refer to values of p_d along the surface (lower) and top (upper) boundaries in height, respectively. Here the coordinate definition of (3.1) is the traditional sigma coordinate used in many hydrostatic atmospheric models where η varies from a value of 1 at the surface to 0 at the upper boundary of the model domain.

In the version 4 release, the vertical coordinate was changed to a more generalized form to allow the influence of terrain to be removed more rapidly as height is increased. This hybrid sigma-pressure vertical coordinate is given by

$$p_d = B(\eta)(p_s - p_t) + [\eta - B(\eta)](p_0 - p_t) + p_t \tag{3.2}$$

where p_0 is a reference sea-level pressure, and $B(\eta)$ defines the relative weighting between the terrain-following sigma coordinate and a pure pressure coordinate, such that η corresponds to the sigma coordinate defined in (3.1) for $B(\eta) = \eta$ and reverts to a hydrostatic pressure coordinate for $B(\eta) = 0$. To smoothly transition the coordinate between the surface and the upper levels, $B(\eta)$ is defined by the third order polynomial

$$B(\eta) = c_1 + c_2\eta + c_3\eta^2 + c_4\eta^3 \tag{3.3}$$

subject to the boundary conditions

$$B(1) = 1, \qquad B_\eta(1) = 1, \qquad B(\eta_c) = 0, \qquad B_\eta(\eta_c) = 0,$$

such that

$$c_1 = \frac{2\eta_c^2}{(1-\eta_c)^3}, \quad c_2 = \frac{-\eta_c(4 + \eta_c + \eta_c^2)}{(1-\eta_c)^3}, \quad c_3 = \frac{2(1 + \eta_c + \eta_c^2)}{(1-\eta_c)^3}, \quad c_4 = \frac{-(1+\eta_c)}{(1-\eta_c)^3}$$

where the subscript η in B_η denotes differentiation, and η_c is the specified value of η at which it becomes a pure pressure coordinate. Now the vertical coordinate metric is defined as

$$\mu_d = \frac{\partial p_d}{\partial \eta} = B_\eta(\eta)(p_s - p_t) + [1 - B_\eta(\eta)](p_0 - p_t) \tag{3.4}$$

Since $\mu_d \Delta\eta = \Delta p_d = -g\rho_d \Delta z$ is proportional to the mass per unit area within a grid cell, the flux forms of the prognostic variables are defined as

$$\mathbf{V} = \mu_d \mathbf{v} = (U, V, W), \qquad \Omega = \mu_d \omega, \qquad \Theta_m = \mu_d \theta_m, \qquad Q_m = \mu_d q_m$$

where $\mathbf{v} = (u, v, w)$ are the covariant velocities in the horizontal and vertical directions, $\omega = \dot{\eta}$ is the contravariant vertical velocity, $\theta_m \approx \theta(1.61 q_v)$ is the moist potential temperature and Q_m represents the the mixing ratios of moisture variables (water vapor, cloud water, rain water, etc.), denoted as $q_m = q_v, q_c, q_r, \ldots$.

Using the variables defined above, the flux-form Euler equations can be written as

$$\partial_t U + (\nabla \cdot \mathbf{V} u) + \mu_d \alpha \partial_x p + (\alpha/\alpha_d)\partial_\eta p \partial_x \phi = F_U \tag{3.5}$$

$$\partial_t V + (\nabla \cdot \mathbf{V} v) + \mu_d \alpha \partial_y p + (\alpha/\alpha_d)\partial_\eta p \partial_y \phi = F_V \tag{3.6}$$

$$\partial_t W + (\nabla \cdot \mathbf{V} w) - g[(\alpha/\alpha_d)\partial_\eta p - \mu_d] = F_W \tag{3.7}$$

$$\partial_t \Theta_m + (\nabla \cdot \mathbf{V} \theta_m) = F_{\Theta_m} \tag{3.8}$$

$$\partial_t \mu_d + (\nabla \cdot \mathbf{V}) = 0 \tag{3.9}$$

$$\partial_t \phi + \mu_d^{-1}[(\mathbf{V} \cdot \nabla \phi) - gW] = 0 \tag{3.10}$$

$$\partial_t Q_m + (\nabla \cdot \mathbf{V} q m) = F_{Q_m} \tag{3.11}$$

with the diagnostic equation for dry hydrostatic pressure

$$\partial_\eta \phi = -\alpha \mu_d \tag{3.12}$$

and the diagnostic relation for the full pressure (dry air and water vapor)

$$p = p_o \left(\frac{R_d \theta_m}{p_0 \alpha_d} \right)^{\gamma}, \tag{3.13}$$

where α_d is the inverse density of air $(1/\rho)$ and α is the inverse density taking into account the full parcel density $\alpha = \alpha_d(1 + q_v + q_c + q_r + \dots)^{-1}$.

In (3.5) - (3.11) the subscripts x, y, and η denote differentiation, $\nabla \cdot \mathbf{V}a = \partial_x(Ua) + \partial_y(Va) + \partial_\eta(\Omega a)$, and $\mathbf{V} \cdot \nabla a = U\partial_x a + V\partial_y a + \Omega\partial_\eta a$ (where a is a generic variable), $\gamma = c_p/c_v = 1.4$ is the ratio of the heat capacities for dry air, R_d is the gas constant for dry air, and p_0 is a reference surface pressure (typically 10^5 Pascals). The right-hand-side (RHS) terms F_U, F_V, F_W, and F_{Θ_m} represent forcing terms arising from model physics, turbulent mixing, spherical projections, and the earth's rotation.

3.3.2 Map Projections

A map projection is a way to transpose the earth's surface onto a flat surface. This requires a systematic transformation of the latitudes and longitudes of locations on the earth's surface into locations on a plane [13]. One such projection is the Lambert conformal, which seats a cone on the earth and projects the surface conformally onto the surface of the unrolled cone. This isotropic transformation requires $(\Delta x/\Delta y)|_{earth} =$ constant everywhere on the grid. To transform the governing equations, a map-scale factor, denoted as m, is defined as the ratio of the distance in computational space to the corresponding distance on the earth's surface:

$$(m_x, m_y) = \frac{(\Delta x, \Delta y)}{\text{distance on the earth}} \tag{3.14}$$

For the purposes of this paper, we are only looking at the Lambert conformal. Due to the projection being isotropic, $m_x = m_y = m$ and typically only varies with latitude.

The WRF solver includes the map-scale factors in the governing equations by

redefining the moments variables as

$$U = \mu_d u/m, \qquad V = \mu_d v/m, \qquad W = \mu_d w/m, \qquad \Omega = \mu_d \omega/m$$

Likewise, the RHS variables in (3.5) - (3.7) also contain the Coriolis and curvature terms along with mixing terms and physical forcings. Using the map-scale factors from (3.14), the Coriolis and curvature terms are cast in the following form:

$$F_{U_{cor}} = \left(f + u\frac{\partial m}{\partial y} - v\frac{\partial m}{\partial x} \right) V - \left(\frac{u}{r_e} + e\cos\alpha_r \right) W \tag{3.15}$$

$$F_{V_{cor}} = -\left[\left(f + u\frac{\partial m}{\partial y} - v\frac{\partial m}{\partial x} \right) U + \left(\frac{v}{r_e} - e\sin\alpha_r \right) W \right] \tag{3.16}$$

$$F_{W_{cor}} = e\left(U\cos\alpha_r - V\sin\alpha_r\right) + \frac{1}{r_e}\left(uU + vV\right) \tag{3.17}$$

where α_r is the local rotation angle between the y-axis and the meridians, ψ is the latitude, $f = 2\Omega_e \sin\psi$, $e = 2\Omega_e \cos\psi$, Ω_e is the angular rotation rate of the earth, and r_e is the radius of the earth. The terms containing the map-scale factors represent the horizontal curvature terms, those containing r_e relate to vertical (earth-surface) curvature and those with e and f are the Coriolis force.

3.3.3 Perturbation Form of the Governing equations

The WRF uses perturbation variables to reduce the truncation errors in the horizontal pressure gradients and the machine rounding errors in the vertical pressure gradient and buoyancy calculations. These new variables are defined as perturbations from a hydrostatically-balanced reference state, and (denoted as overbars) are defined as a function of height only and that satisfy the governing equations for an atmosphere at rest. That is, the reference state is in hydrostatic balance and is strictly only a function of $\overline{z}$. Therefore, $p = \overline{p}(\overline{z}) + p'$, $\phi = \overline{\phi}(\overline{z}) + \phi'$, $\alpha = \overline{\alpha}_d(\overline{z}) + \alpha'_d$, and $\mu_d = \overline{\mu}_d(x, y) + \mu'_d$. Because the η coordinate surfaces are generally not horizontal, the reference profiles $\overline{p}$, $\overline{\phi}$, and $\overline{\alpha}$ are functions of (x, y, η). The hydrostatically balanced

portion of the pressure gradients in the reference sounding can be removed without approximation to the equations using these perturbation variables. The momentum equations (3.5) - (3.7) can now be written as

$$\partial_t U + m[\partial_x(Uu) + \partial_y(Vu)] + \partial_\eta(\Omega u) + (\alpha/\alpha_d)[\mu_d(\partial_x\phi' + \alpha_d\partial_x p' + \alpha'_d\partial_x\overline{p}) + \partial_x\phi(\partial_\eta p' - \mu'_d)] = F_U \tag{3.18}$$

$$\partial_t V + m[\partial_x(Uv) + \partial_y(Vv)] + \partial_\eta(\Omega v) + (\alpha/\alpha_d)[\mu_d(\partial_y\phi' + \alpha_d\partial_y p' + \alpha'_d\partial_y\overline{p}) + \partial_y\phi(\partial_\eta p' - \mu'_d)] = F_V \tag{3.19}$$

$$\partial_t W + m[\partial_x(Uw) + \partial_y(Vw)] + \partial_\eta(\Omega w) - m^{-1}g(\alpha/\alpha_d)[\partial_\eta p' - \overline{\mu}_d(q_v + q_c + q_r)] + m^{-1}\mu'_d g = F_W \tag{3.20}$$

and the mass conservation equation (3.9) and geopotential equation (3.10) become

$$\partial_t\mu'_d + m[\partial_x U + \partial_y V] + m\partial_\eta\Omega = 0 \tag{3.21}$$

$$\partial_t\phi' + \mu_d^{-1}[m(U\partial_x\phi + V\partial_y\phi) + m\Omega\partial_\eta\phi - mgW] = 0 \tag{3.22}$$

and the diagnostic equation for dry hydrostatic pressure, (3.12) becomes

$$\partial_\eta\phi' = -\overline{\mu}_d\alpha'_d - \alpha_d\mu'_d \tag{3.23}$$

The conservation equations for the potential temperature (3.8) and the scalar moisture (3.11) do not need to be redefined using a perturbation technique, but will still have changes made to reflect the map projection. These equations become

$$\partial_t\Theta_m + m[\partial_x(u\theta_m) + \partial_y(V\theta_m)] + m\partial_\eta(\Omega) = F_{\Theta_m} \tag{3.24}$$

$$\partial_t Q_m + m[\partial_x(Uq_m) + \partial_y(Vq_m)] + m\partial_\eta(\Omega q_m) = F_{Q_m} \tag{3.25}$$

(3.18) - (3.25), and the equation of state (3.13) represent the equations used in the WRF solver. The Coriolis terms (3.15) - (3.17), mixing terms, and parameterized physics are all included in the RHS terms defined in these equations.[1]

[1]For more information about the various mixing terms and parameterized physics settings used in the WRF, please see Chapters 4 and 8 of the WRF Version 4 Technical Manual [12]

3.4 The WRF Preprocessing System – Initial and Boundary Conditions

The WRF may be run with user-defined initial conditions for idealized simulations, or it may be run using interpolated data files from either an external analysis or forecast for real-data cases. For the scope of this paper, we will talk about initializing the WRF using real-data cases as defined by the authors of the technical manual.

Initial conditions for real-data cases are pre-processed through a separate package called the WRF Preprocessing System (WPS). Output from the WPS is passed to the ARW real-data preprocessor, which generate initial and lateral boundary conditions. Here we will discuss the steps taken to build these conditions using a real-data case. Although the WPS is outside of the ARW, it is appropriate to show how the system converts the raw meteorological and static terrestrial data into data that can be used for the solver.

3.4.1 Use of the WPS by ARW

The WPS is a set of programs that takes terrestrial and meteorological data that is usually in the GRIB[2] format and transforms them for input to the ARW pre-processor program for real-data cases. Figure 3.1 shows the flow of data for the WPS. Text in the rectangular boxes indicates program names. The first step that the WPS takes is to define a physical grid (which includes projection type, location on the globe, number of grid points, nest locations, and grid distances) and to interpolate static field to the prescribed domain. Independent of the domain configuration, an external forecast is processed by the WPS GRIB decoder, which diagnoses required field and reformats GRIB data into an internal binary format. When given a specified domain,

[2]GRIB or GRIdded Binary is a data format developed by the World Meteorological Organization (WMO) that is commonly used to store historical and forecast weather data. For more information please see GRIB FM 92-IX in the WMO Manual on Codes No.306

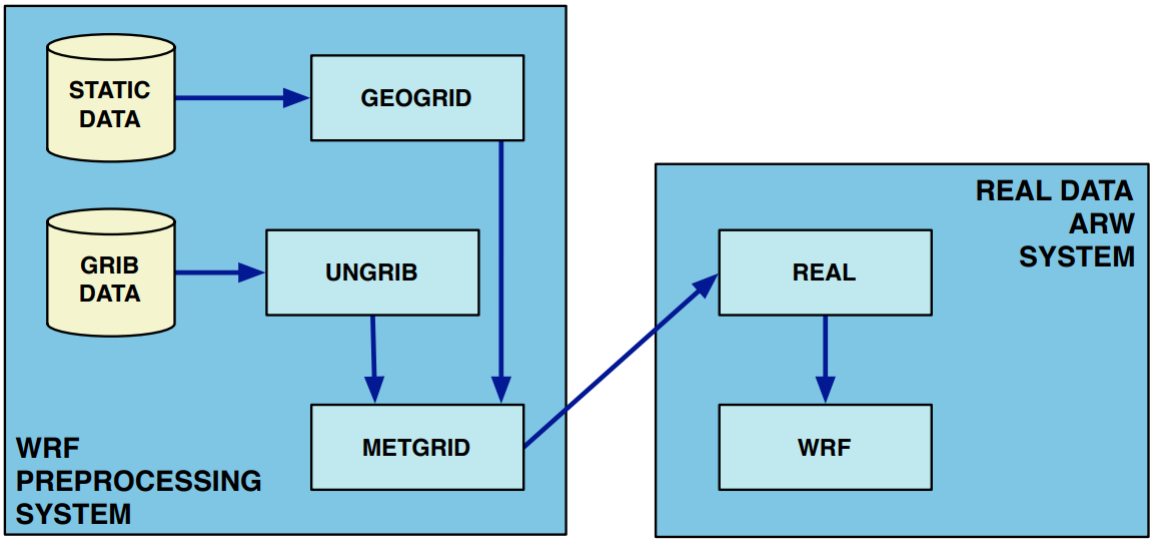

Figure 3.1: Schematic showing data flow and program components in WPS, and how WPS feeds initial data to ARW (From Figure 5.1 in [12])

the WPS horizontally interpolates meteorological data onto the projected domain. The output data supplies a complete three-dimensional snapshot of the atmosphere on the selected model grid's horizontal staggering, at the selected time slices.

Input to the ARW real-data processor from WPS contain surface and three-dimensional fields that include temperature (K), relative humidity (%), geopotential height (m), pressure (Pa), and the horizontal components of wind speed (m/s, rotated to the model projection). The two-dimensional static terrestrial fields include land/water mask, map scale factors, map rotation angle, annual mean temperature, and latitude/longitude. After WPS processing, the two-dimensional time-dependent fields from the external model include surface at sea-level pressure (Pa), snow depth (m), skin temperature (K), and sea surface temperature (K).

3.4.2 Reference State

The meteorological data are partitioned into reference and perturbation fields. For real-data cases, the reference state is defined by terrain elevation and several user-definable constants that include

- p_0 (10^5 Pa) reference sea level pressure;
- T_0 (usually 270 to 300 K) reference sea level temperature;
- A (50 K) temperature difference between the pressure levels of p_0 and p_0/e;
- T_{min} (200 K) minimum temperature permitted;
- γ_{strat} (-11 K) standard stratosphere lapse rate;
- p_{strat} (0 Pa) pressure at which stratospheric warming begins.

Using these parameters, the dry reference state surface pressure is

$$\overline{p}_s = p_0 \exp\left(\frac{-T_0}{A} + \sqrt{\left(\frac{T_0}{A}\right)^2 - \frac{2\phi_{sfc}}{AR_d}}\right) \tag{3.26}$$

From (3.26), the three-dimensional reference pressure (dry hydrostatic pressure p_d) is computed as a function of the vertical coordinate η levels and the model top p_t:

$$\overline{p}_d = B(\eta)(\overline{p}_s - p_t) + [\eta - B(\eta)](p_0 - p_t) + p_t. \tag{3.27}$$

With (3.27), the reference temperature, though not permitted to get colder than T_{min}, is a straight line on a skew-T plot, defined as

$$\overline{T}_d = \max\left(T_{min}, T_0 + A \ln \frac{\overline{p}_d}{p_0}\right).$$

For vertical locations where the $\overline{p}_d < p_{strat}$, the reference profile warms:

$$\overline{T}_d = T_{min} + \gamma_{strat} \ln \frac{\overline{p}_d}{p_{strat}}$$

The isobaric temperature and the stratospheric correction supply a reasonable reference temperature up to approximately 100 Pa. From the reference pressure and the final reference temperature, the reference potential temperature is then defined as

$$\overline{\theta}_d = \overline{T}_d \left(\frac{p_0}{\overline{p}_d} \right)^{\frac{R_d}{C_p}} \tag{3.28}$$

and the reciprocal of the reference density using (3.27) and (3.28) is given by

$$\overline{\alpha}_d = \frac{1}{\overline{\rho}_d} = \frac{R_d \overline{\theta}_d}{p_0} \left(\frac{\overline{p}_d}{p_0} \right)^{-\frac{C_v}{C_p}} . \tag{3.29}$$

Using (3.27), the base state coordinate metric is given as

$$\overline{\mu}_d = \frac{\partial \overline{p}_d}{\partial \eta} = B_\eta(\eta) \left(\overline{p}_s - p_t \right) + \left[1 - B_\eta(\eta) \right] \left(p_0 - p_t \right) . \tag{3.30}$$

From (3.29) and (3.30), the reference state geopotential defined from the hydrostatic relation is

$$\delta_\eta \overline{\phi} = -\overline{\alpha}_d \overline{\mu}_d .$$

3.4.3 Vertical Interpolation and Extrapolation

The ARW real-data processor vertically interpolates the three-dimensional input field using functions of dry pressure. Input data from the WPS contain both a total pressure and a moisture field (typically relative humidity is used). Starting at the top of each column of input pressure data, integrated moisture is subtracted from the pressure field, step-wise, down to the surface. This total dry surface pressure p_s defines the vertical coordinate metric

$$\mu_d = \overline{\mu}_d + \mu'_d = B_\eta(\eta)(p_s - p_t) + [1 - B_\eta(\eta)](p_0 - p_t). \tag{3.31}$$

With the ARW vertical coordinate η, the model lid p_t, and the column dry pressure known at each (i, j, k) location, the three-dimensional arrays are interpolated.

These vertical calculations are always interpolations in the free atmosphere, however when near the model surface, it is possible that there will be an inconsistency between the input surface pressure and the ARW surface pressure which may lead to an extrapolation. The default behavior for extrapolating the horizontal winds and the relative humidity below the known surface keeps the values constant, with no gradient. For potential temperature, a default value of 6.5 K/km lapse rate is applied. Vertical interpolation of the geopotential field is optional and is handled separately. Since a known lower boundary condition exists, no extrapolation is required.

3.4.4 Perturbation State

In the real-data preprocessor, first a topographically defined reference state is computed, then the input three-dimensional data are vertically interpolated in dry pressure space. With the potential temperature θ and mixing ratio q_v available on each η level, pressure, density, and height diagnostics are handled. The perturbation μ'_d, given the reference value $\overline{\mu}_d$ defined in (3.30) and the vertical coordinate metric μ_d defined in (3.31), is defined as

$$\mu'_d = \mu_d - \overline{\mu}_d. \tag{3.32}$$

Starting with the reference state fields (3.27) - (3.30) and the perturbation equation (3.32), the perturbation fields for pressure and inverse density are diagnosed. The pressure perturbation includes moisture, and is diagnosed from the hydrostatic equation

$$\delta_\eta p' = \mu'_d \left(1 + \overline{q_v}^\eta\right) + \overline{q_v}^\eta \overline{\mu}_d,$$

which is integrated down from the model top (where $p' = 0$) to recover p'. The total dry inverse density is given as

$$\alpha_d = \frac{R_d}{p_0}\theta_m \left(\frac{p'_d + \overline{p}_d}{p_0}\right)^{-\frac{C_v}{C_p}},$$

which defines the perturbation field for the inverse density

$$\alpha'_d = \alpha_d - \overline{\alpha}_d.$$

The perturbation geopotential is diagnosed from the hydrostatic relation

$$\delta_\eta \phi' = -(\mu_d \alpha'_d + \mu'_d \overline{\alpha}_d)$$

by upward integration, using the terrain elevation as the lower boundary condition.

3.4.5 Lateral Boundary Conditions

For real-data cases, the specified lateral boundary condition for the coarse grid is supplied by an external file that is generated by the real-data processor. This file contains records for the fields u, v, θ, $q - v$, ψ', and μ'_d that are used by ARW to constrain the lateral boundaries. The lateral boundary file holds one less time period than was processed by WPS. Each of these variables has both a valid value at the initial time of the lateral boundary time, and a tendency term to get to the next boundary time period. For example, assuming a three-hourly availability of data from WPS, the first time period of the lateral boundary file for u would contain data for both coupled u (map scale factor and μ_D interpolated to the variable's staggering) at the 0 hour time step ($0h$)

$$U_{0h} = \left. \frac{\overline{\mu_d} x_u}{\overline{m}^x} \right|_{0h},$$

and a tendency value defined as

$$U_t = \frac{U_{3h} - U_{0h}}{3h},$$

which would take a grid point from the initial value to the value at the three-hour simulation time. The horizontal momentum fields are coupled both with μ_d and the

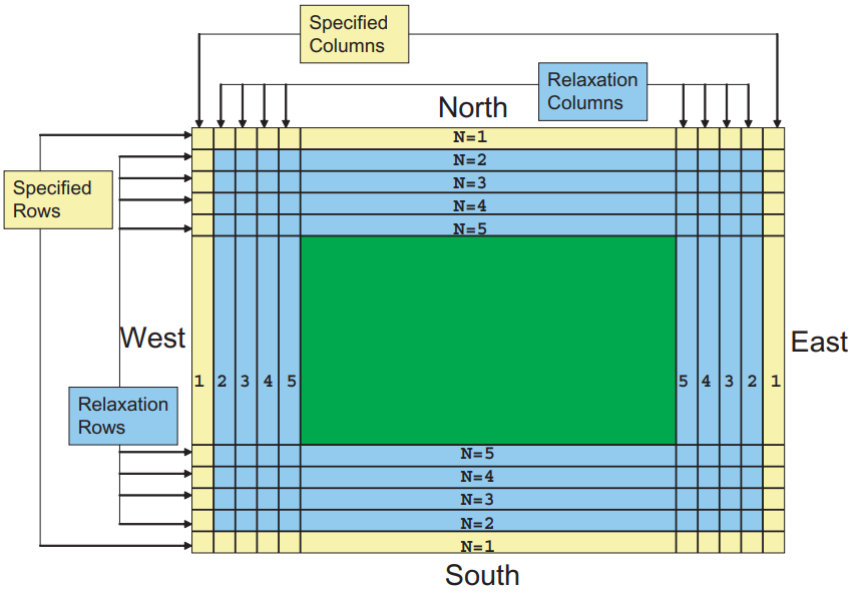

Figure 3.2: Specified and relaxation zones for a grid with a single-specified row and column, plus four rows and columns for the relaxation zone. These are typical values used for a specified lateral boundary condition for a real-data case. In this figure, the weight of the relaxation term would be identically zero for the fifth row or column in from the boundary edge. [12]

inverse map factor. The other three-dimensional fields are coupled only with μ_d. The μ'_d lateral boundary field is not coupled.

Each lateral boundary field is defined along the four sides of the rectangular grid (referred to as the north, south, east, and west sides). Boundary values and tendencies for vertical velocity and the non-vapor moisture species are included in the external lateral boundary file, but act as place-holders for the nested boundary data for the fine grids. The width of the lateral boundary zone along each of the four sides is user-selectable at run-time as shown in figure 3.2

The coarse grid specified lateral boundary is comprised of both a specified and a relaxation zone. For the coarse grid, the specified zone is determined entirely by

temporal interpolation from an external forecast or analysis (supplied by WPS). The width of the specified zone is run-time configurable, but is typically set to 1. That is, the last row and column along the outer edge of the outer-most grid is entirely specified by temporal interpolation, using data from an external model. The second region of the lateral boundary for the coarse grid is the relaxation zone, which is where the model is nudged or relaxed towards the large-scale forecast (rows 2-5 in figure 3.2). The size of the relaxation zone is a run-time option.

3.4.6 Model Discretization

The ARW solver has two different types of discretizations to approximate the solutions of (3.13) and (3.18) - (3.25) developed in subsection 3.3.3. The first is a temporal discretization that uses a time-split integration scheme. Slow or low-frequency meteorologically significant modes are integrated using a third-order Runge-Kutta time integration scheme, while the high-frequency acoustic modes are integrated over smaller time steps to maintain numerical stability. The horizontally propagating acoustic modes (which includes the external mode present in the mass-coordinate equations using a constant pressure upper boundary condition) and gravity waves are integrated using a forward-backward time integration scheme, and vertically propagating acoustic modes and buoyancy oscillations are integrated using a vertically implicit scheme that use the acoustic time step.

The second is a spatial discretization in which the ARW uses a C grid staggering for the variables as shown in figure 3.3. That is, normal velocities are staggered one-half grid length from the thermodynamic variables. The variable indices, (i, j, k) indicate variable locations with $(x, y, \eta) = (i\Delta x, j\Delta y, k\Delta\eta)$. We will denote the points where θ is located as being mass points, and locations where u, v, and w are defined as u points, v points, and w points, respectively. Not shown are the moisture variable q_i,

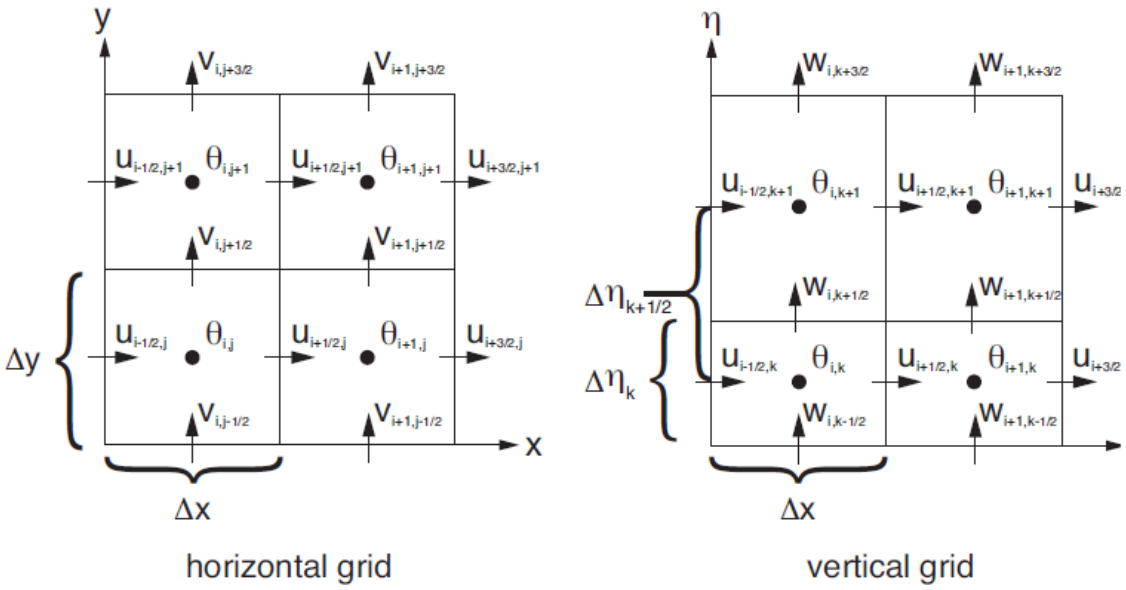

Figure 3.3: Horizontal and vertical grids of the ARW [12]

and the coordinate metric μ, defined at the mass points on the discrete grid, and the geopotential ϕ that is defined at the w points. The diagnostic variables used in the model, the pressure p, and inverse density α, are computed at mass points. The grid lengths Δx and Δy are constants in the model formulation; changes in the physical grid lengths associated with the various projections to the sphere are accounted for using the map factors from subsection 3.3.2. The vertical grid length $\Delta\eta$ is not a fixed constant; it is specified in the initialization. The user is free to specify the η values of the model levels subject to the constraint that $\eta = 1$ at the surface, $\eta = 0$ at the model top, and η decreases monotonically between the surface and model top. For a detailed description of this process, please see Chapter 3 of the WRF Version 4 Technical Manual [12].

4 FORECAST VERIFICATION

4.1 Introduction

After a forecast, the approximated solutions to the governing equations, is generated from a weather model, the accuracy and usefulness need to be tested. This can help see where errors start to form and give ideas on how to improve the model. This is done using a technique called verification, the assessment and quantification of the relationship between a matched set of forecasts and observations [14]. Verification is done using various statistical methods, and while many techniques are in use today, for the purposes of this paper, we will use the basic methods introduced in 1989 by Stanski, Wilson, and Burrows.

4.2 Verification Framework

4.2.1 Question Types and Stratification

Stanski, Wilson, and Burrows introduced a general model to follow when stepping through the verification process and a general outline can be found in Figure 4.1. The authors emphasized that the verification goal be set before the process is started and stated the two types of these goals – administrative or scientific. Administrative goals usually include justification on purchasing new equipment, introducing new forecast products, monitor the overall quality of a forecast model and to track the change in quality over time. They can also answer questions such as "Is the accuracy of the forecasts improving?" and "Are objective temperature forecasts better than subjective ones?" Scientific goals are usually for identifying strengths and weaknesses of forecast

products that can be useful to provide the necessary information needed for further research and development of the forecast models and products. They can also answer questions such as "What does a temperature forecast of $-10°$ Celsius really mean?" or "How accurately can I forecast extreme rainfalls?"

Once the goal is set, the sample data set may need to be stratified by separating the events of the sample into two or more groups according to a selection rule and then carrying out the verification for each group separately, to help meet the intended purpose and can be done in multiple ways.

One type of stratification is known as external stratification and is done by selecting rules that are independent of the element being verified. A popular choice for this type is selecting by time of day or by season. It can be done at any time during the verification process before computation and may be done for either administrative or scientific purposes. However, from the example scientific questions mentioned earlier, additional stratification may need to be done on the sample data. As an an example, suppose that the question that is trying to be answered regards whether or not a model can accurately forecast extreme events. In this case, the data must be stratified between extreme and non-extreme events. This process is called internal stratification since the rules are determined according to the purpose of the verification, using the element that is being verified.

Internal Stratification can be done in one of two ways. The first is stratification according to the observation. This process defines categories according to the observed values of the weather element. Then, verification measures can be calculated for each category of the observations, and the statistics are said to be conditioned on the observation. An example of this is a conditional distribution of forecasts given a specific value or range of values of the observation. The second way is to stratify according to the forecast. This is like the previous, but the categories are defined using the forecasted values instead and are said to be conditioned on the forecast.

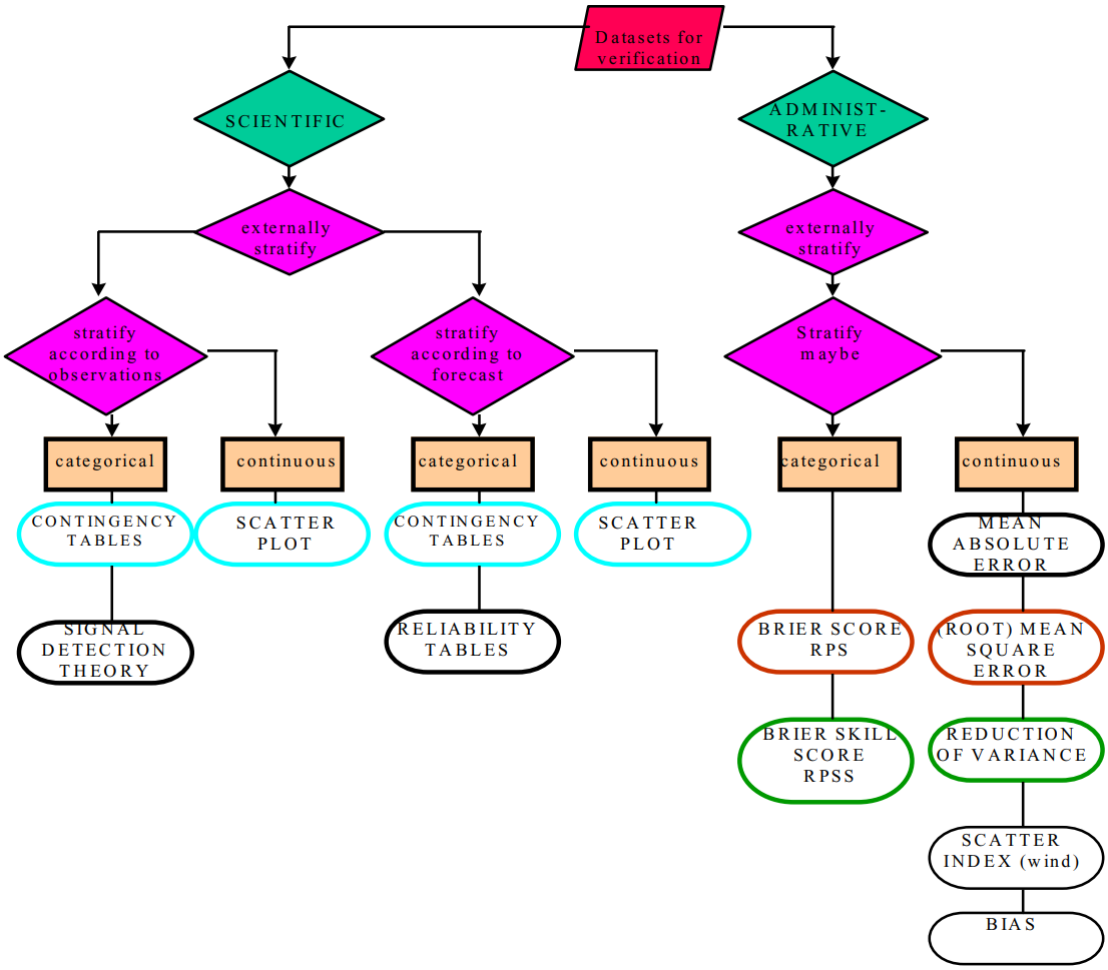

Figure 4.1: A general model for verification introduced by Stanski, Wilson, and Burrows (1989) [14].

The type of stratification to choose depends on the question that is to be answered and many will require both types for a complete answer. Furthermore, as suggested by the diagram in Figure 4.1, different verification measures (denoted by the oval boxes at the bottom) imply one or the other type of stratification and different information about the product at hand.

When answering an administrative question, stratifying internally is rarely done. Because of its nature, most of the time just having a simple summary statistic is all that is needed. This is done to represent the quality of the data in few numbers as possible or to facilitate comparisons or identify trends within the model and is associated with summary scoring rules shown in Figure 4.1. However, using summary statistic scores only gives a general idea of how a model forecast worked. When used in a scientific setting, it will often create more questions than answering the original one. For example, the summary statistic can only state how well a model performed compared to another and cannot state under which conditions that model might favor the other.

4.2.2 Predictand Types

The next step in the verification process is to decide the appropriate measure to answer the question at hand. The rectangular boxes in Figure 4.1 show that the choice depends on the nature of the forecasts to be verified. For this purpose, there are two distinct types used: continuous and categorical. Continuous predictands are those elements where a specific value or range of values is forecasted. This includes weather elements like temperature and wind speed and both are nearly forecasted this way. Categorical predictands are those elements for which the forecast is of the occurrence of the event in one of two or more mutually exclusive and exhaustive categories of the element. This includes elements like occurrence of measurable precipitation and

precipitation type.

4.2.3 Adjusting for NWP Models

Verification for NWP models start with a data set consisting of forecast and observation data matched spatially and temporally. When matching the data spatially, observations must either be analyzed to the grid of the forecast data, or the forecast data must be interpolated in space to observation points. The former is the option most often used, but can draw some criticism as the interpolated observations aren't as representative of the truth as they were before interpolation and will have the potential to show bias towards the model in the statistical results.

As mentioned previously, the process requires decisions be made about the purpose of the verification before designing a system. Administrative verification answers questions about trends in model accuracy and skill and can be used to compare the accuracy of different models. Scientific verification involves answering questions about spatial and/or temporal variations in the performance of a model that can be used as feedback to improve the model being tested on. The verification data set must be carefully stratified according to the characteristics of the features that are to be examined. Spatial stratification can be done for both Administrative or Scientific verification. When done for administrative purposes, the stratification is usually done over different portions of the model domain that correspond to administrative regions of the country, whereas for scientific purposes, the areas are often chosen to reflect different climatological regimes such as mountainous areas.

When stratifying externally, the dataset is divided according to season, model run time, or another selection rule that is independent of the parameter being verified. In most cases, stratification by season is chosen the most showing differences in performance characteristics between seasons.

4.2.4 Forecast Quality

The Stanski, Wilson, and Burrows note that no single verification measure provides complete information about the quality of a forecast model. All give information about one or more aspects of the quality, also known as attributes, of a forecast model and a verification system will often describe the attributes that are most pertinent to fulfilling the goal of what is being verified. These attributes are briefly defined below:

1. Accuracy is a general term indicating the level of agreement between forecast weather and true weather as represented by observations. The difference between an observed value and the forecast value is the error. The smaller the error is, the higher the accuracy.

2. Skill, or relative accuracy is defined as the accuracy of a forecast relative to the accuracy of forecasts produced by some standard procedure. Common standards, which are considered to have no skill (such as forecasting from observations alone) are climatology, persistence, and chance. Skill scores provide a means of accounting for variations in accuracy which have nothing to do with the forecaster's ability to forecast.

3. Reliability is equivalent to bias; and is simply the average agreement between the stated forecast value of an element and the observed value. Often, reliability can be improved by giving the bias feedback to the forecaster and affording the opportunity to remove it from future forecasts.

4. Resolution is the ability of the forecast to sort or resolve the set of sample events into subsets with different frequency distributions. Resolution is related to the standard deviation or variance of the observations stratified by the forecast.

5. Sharpness, an attribute of forecasts alone, refers to the tendency to forecast the extreme values. For probability forecasts, there is the tendency toward

forecasting 0% and 100% probability. It represents the tendency to "go out on a limb" and is directly related to the variance of the distribution of forecasts.

6. Uncertainty is the variance of the observations in the verification sample and does not depend on the forecasts in anyway. It is considered to be the "difficulty" of the forecast set. Greater variance implies larger or more frequent changes in the weather element being verified, and when viewed as a time series, these elements are harder to forecast than more persistent weather situations. It is variations in the uncertainty between datasets that make it hazardous to compare verification statistics that are sensitive to uncertainty.

4.3 Verification Methods

The most common verification methods that Stanski, Wilson, and Burrows mention are listed at the bottom ovals in Figure 4.1. The methods that will be used for this paper are explained in the following subsections.

4.3.1 Bias

The Bias, or the Mean (Algebraic) Error, indicates the average direction of the deviation from observed values, but may not reflect the magnitude of the error. However, the bias does give a measure of overall reliability. The bias is given by:

$$\text{BIAS} = \frac{1}{N}\left[\sum_{i=1}^{N}(F_i - O_i)\right], \tag{4.1}$$

where N is the total number of forecasts, F_i is the i^{th} forecast value and O_i is the corresponding observation value. A positive bias indicates that the forecast value exceeds the observed value on average, whereas a negative bias value indicates the opposite.

4.3.2 Mean Absolute Error

The Mean Absolute Error (MAE) is similar to the calculation in 4.1 in Section 4.3.1, however now the only change is that the absolute value is now being taken of the difference inside of the summation:

$$\text{MAE} = \frac{1}{N}\left[\sum_{i=1}^{N}|(F_i - O_i)|\right], \tag{4.2}$$

where the definitions of N, F_i, and O_i are the same as 4.1. The MAE is a linear score which gives the "average" magnitude of the errors, but not the direction of the deviation but does give a measure of overall accuracy.

4.3.3 Time-Mean Absolute Error

When verifying models, the Bias and MAE are normally used to compare the accuracy and reliability of each forecast hour. However, it is also good to know the overall accuracy for the entire simulation run. To do this, the time-averaged mean absolute error (TMAE) is determined and is done by averaging the calculated MAEs for each hour of the entire simulation run as defined in 4.3 below:

$$\text{TMAE} = \frac{1}{M}\left[\sum_{t=1}^{M}\text{MAE}_t\right] \tag{4.3}$$

Where the definitions of N, F_i, and O_i are the same as 4.1, and M denotes the number of time steps in the simulation run, and MAE_t denotes the MAE at the t^{th} time step [16].

5 ANALYSIS

5.1 Experimental Setup

The purpose of the experiment was to test the differences in the output of the WRF when different real-case inputs are used. For this experiment, each simulation consisted of two different runs of the model, one run used inputs from the NAM and the second run used inputs from the GFS. The software used to run the model is the Unified Environmental Modeling System (UEMS) [10] created by Robert A. Rozumalski, the NWS National Science Operations Officer and Science and Training Resource Coordinator at UCAR. The UEMS is a complete, full-physics, NWP package that contains the WRF system into a single user-friendly, forecasting system. Due to the scalability of the WRF, a desktop workstation with a multi-core processor was used running in a virtual machine. See Table 5.1 for detailed information about the system[1].

Ten different simulations were run during a time period of December 31, 2020 to February 14, 2021 using either the NAM and GFS files from 00z and 12z[2].

5.2 Steps of the UEMS

The UEMS is comprised of multiple script files that are used at various steps of the WRF process. The first is the WRF Domain Wizard (dwiz), a graphical interface to help create a custom domain over the area of interest. As shown in figure 5.1, the

[1]Due to software issues during the time period, UEMS and WRF were reinstalled to the versions listed as (x.x.x). This did not affect previous data as the version changes only fixed minor bug fixes unrelated to the settings used.

[2]XXz refers to the hour in Zulu time and is the same as the Coordinated Universal Time (UTC).

Processor	AMD Ryzen 5 1600 Six-Core CPU
RAM Capacity	32 GB
Operating System	Proxmox Virtual Environment
Number of Virtual Machine CPUs	12 threads
Virtual Machine RAM	32 GB
Virtual Machine Operating System	Linux Mint 20
UEMS Version	19.8.1 (21.3.1)
WPS & WRF Version	4.1.2 (4.2.2)

Table 5.1: System Information

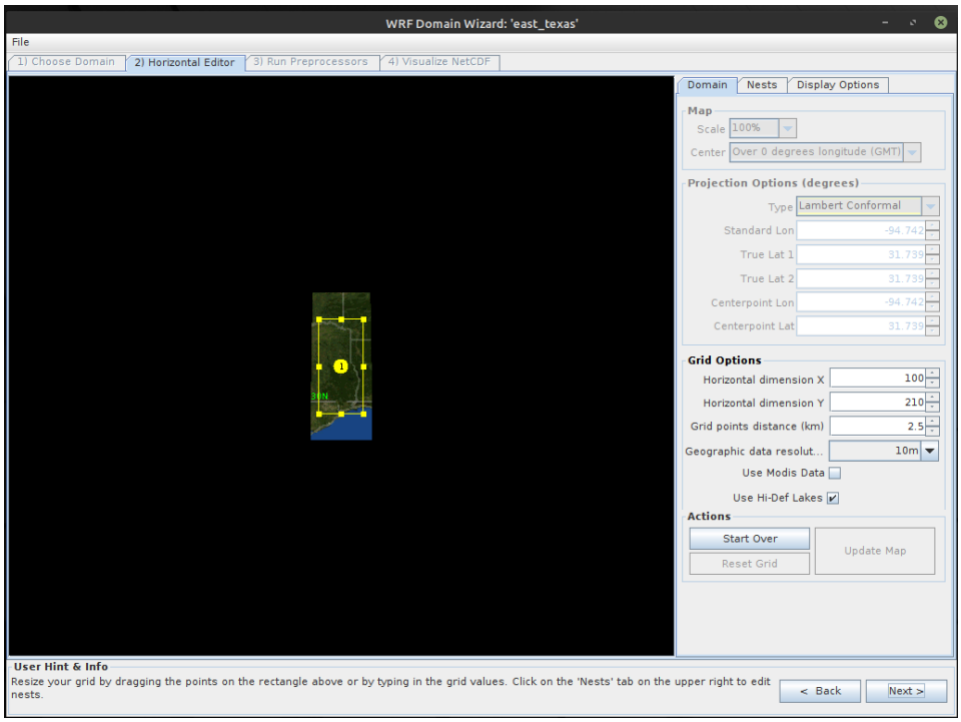

Figure 5.1: Screenshot of the WRF Domain Wizard from the UEMS

domain created for this experiment, was centered over Nacogdoches, Texas with a grid size of 100 km by 210 km, using a grid point size of 2.5 km. After all the settings

have been set, the wizard will then create the required files needed for the WPS.

The next step in the process is to download and process the input files. By default the UEMS will download the latest outputs from the model of choice and run them through the WPS. For example, if the current time that the script is executed is at 02z and the latest model run was at 00z, the UEMS will download all the files from the 00z run. Execution of this script is also when the forecast timing is selected, by default, it will use a 24 hour forecast period.

After the UEMS create the necessary files from the WPS, the main WRF simulation is ready to run. The third script will run the WRF simulation, and scale across the set number of processing threads. For this experiment, the WRF used 10 out of the 12 available processing threads.

Finally, after the UEMS completes the WRF simulation, the final script will then post process the output files (one for each forecast hour) into the GRIB format for reading and analysis.

5.3 Methodology

As mentioned in the previously, the domain used was a 100 km by 210 km grid with a grid point every 2.5 km. For the input files, the GFS contained a file for every 3 hours up to 24 whereas the NAM files contained a file for each hour, up to 24 hours. The default WRF settings for the UEMS were used as shown in Table 5.2. Information about the settings can be found in the WRF Technical manual [12]. The simulation ran the default 24 hour forecast period, and the outputs were analyzed. Maps were created using the Python package MetPy [9], and data points were obtained using the external `wgrib2` tool included with the UEMS package.

For the purpose of this paper, the temperature, and the relative humidity (both using the 2 meter above ground level variant) were analyzed. The data were stratified

Number of Domains	1
Horizontal Grid Spacing (km)	2.5
Microphysics Scheme	Purdue Lin
Planetary Boundary Layer	Yonsei University Scheme
Cumulus Scheme	Kain-Fritsch
Land-Surface Scheme	NOAH Land Surface Model
Long Wave Radiation	RRTM Scheme
Short wave Radiation	Dudhia Scheme

Table 5.2: WRF Configuration (Default Settings in UEMS)

by each model run, each collecting data from 34 NOAA observation stations located in 34 counties in Eastern Texas. A list of the stations and counties can be found in Table 5.3 with a corresponding map in Figure 5.2.

5.4 WRF Verification

In this section we will cover the verification analysis of the WRF runs, and compare the GFS inputs with the NAM inputs for each variable forecasted. Each analysis will contain the bias and mean absolute error (MAE) for all 34 counties. While the TMAEs and cumulative biases for each run are listed in Table 5.4, we will discuss each forecast run individually.

5.4.1 2 Meter Temperature

The first simulation analyzed was the 12z run on December 31, 2020. On average, the GFS input run trended higher than what was observed where as the NAM run was just above the observed temperatures, and towards the end of its run, was matching the observed with negligible error as shown in Figure 5.3. This also caused the bias of

Map Number	County	StationID	Station Type	Station Name	Latitude	Longitude
1	Lamar	KPRX	NWS/FAA	Paris Cox Field	33.63	-95.45
2	Red River	CKST2	RAWS	Clarksville	33.63	-95.09
3	Bowie	TEXT2	RAWS	Texarkana	33.37	-94.05
4	Delta	E4423	ARPS	Yantis	32.9	-95.7
5	Hopkins	KSLR	NWS/FAA	Sulphur Springs Municipal Airport	33.16	-95.62
6	Titus	KOSA	NWS/FAA	Mt. Pleasant Regional Airport	33.1	-94.96
7	Cass	DENT2	RAWS	Linden	33.01	-94.36
8	Wood	KJDD	NWS/FAA	Wood County Airport	32.74	-95.5
9	Upshur	KJXI	NWS/FAA	Gilmer Municipal Airport	32.7	-94.95
10	Harrison	KASL	NWS/FAA	Marshall	32.52	-94.31
11	Van Zandt	TWB71	TWDB	Vaca Moo Airport	32.43	-95.71
12	Smith	KTYR	NWS/FAA	Tyler Pounds Airport	32.36	-95.4
13	Gregg	KGGG	NWS/FAA	East Texas Regional Airport	32.39	-94.71
14	Henderson	KF44	NWS/FAA	Athens Municipal Airport	32.16	-95.83
15	Rusk	KRFI	NWS/FAA	Rusk County Airport	32.14	-94.85
16	Panola	K4F2	NWS/FAA	Panola County Airport	32.18	-94.3
17	Anderson	KPSN	NWS/FAA	Palestine Municipal Airport	31.78	-95.71
18	Cherokee	KJSO	NWS/FAA	Cherokee County Airport	31.87	-95.22
19	Nacogdoches	KOCH	NWS/FAA	Nacogdoches Regional Airport	31.58	-94.72
20	Shelby	KF17	NWS/FAA	Center Municipal Airport	31.83	-94.16
21	Houston	RTCT2	RAWS	Ratcliff	31.39	-95.14
22	Angelina	KLFK	NWS/FAA	Angelina County Airport	31.23	-94.75
23	Sabine	SSRT2	RAWS	Sabine South	31.28	-93.84
24	Walker	KUTS	NWS/FAA	Huntsville Municipal Airport	30.74	-95.59
25	San Jacinto	CPGT2	RAWS	Coldsprings	30.52	-95.09
26	Tyler	WVLT2	RAWS	Woodville	30.74	-94.43
27	Jasper	KJAS	NWS/FAA	Jasper County Airport	30.89	-94.03
28	Newton	KRBT2	RAWS	Kirbyville	30.63	-93.83
29	Montgomery	KCXO	NWS/FAA	Montgomery County Airport	30.36	-95.41
30	Liberty	K6R3	NWS/FAA	Cleveland	30.36	-95.01
31	Hardin	F3104	APRS	Kountze	30.34	-94.28
32	Harris	KIAH	NWS/FAA	Houston Untercontinential Airport	29.98	-95.36
33	Chambers	HILT2	RAWS	Anahuac	29.67	-94.44
34	Jefferson	KBPT	NWS/FAA	Southeast Texas Regional Airport	29.95	-94.03

Table 5.3: Station List Used for Verification

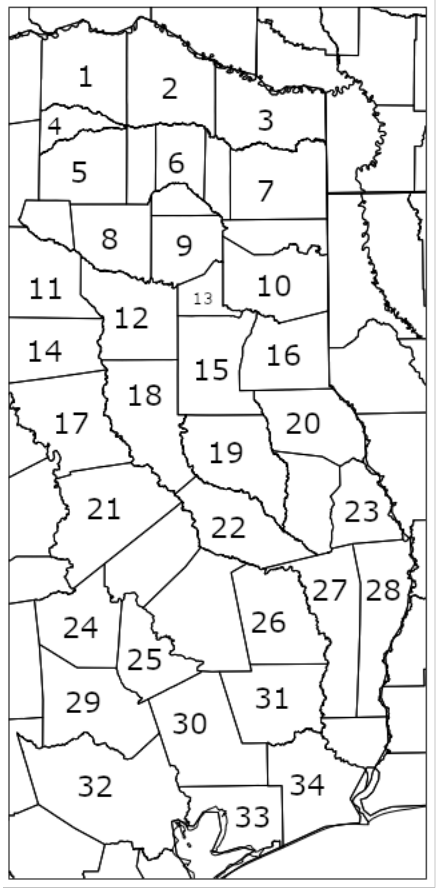

Figure 5.2: Map of counties used for verification. The number in each county represents the number in Table 5.3

the NAM to be minimal reaching a maximum of just over 2.1°F towards the model's favor, but reaching a bias as low as −0.6°F towards the observation's favor. The bias of the GFS favored the model the entire run and reached a maximum of 5.9°F and a minimum, which occurred at the initial start of the run, of 1.3°F. Likewise, the error rates of the run are close together. However, after the third forecast hour (the fourth data point on the graph), the error trend of the GFS rises significantly up to 5.3°F and slowly climbing to the maximum of 6.3°F before slowly falling as the simulation progresses. The MAE of the NAM reached a maximum of 3.0°F and falls close to

	12/31/20 12z		1/6/21 12z		1/7/21 12z		1/10/21 00z		1/25/21 12z	
	GFS	NAM	GFS	NAM	GFS	NAM	GFS	NAM	GFS	NAM
Temperature Bias (°F)	3.9815	0.7262	0.3435	0.3618	-1.3195	-1.3592	0.7931	0.7203	-0.0766	-0.1156
Temperature MAE (°F)	4.3817	1.9948	1.9776	2.8986	2.9088	2.5984	2.1107	1.8035	3.2957	3.1080
Relative Humidity Bias (%)	-1.5783	-0.4053	-0.0211	-0.8076	0.4160	0.6817	-5.7183	-4.2312	-7.1771	-2.7820
Relative Humidity MAE (%)	5.4132	4.3942	6.0136	8.1383	7.1233	6.8040	9.4203	8.3186	14.2649	12.4408
	2/8/21 12z		2/9/21 12z		2/10/21 12z		2/11/21 12z		2/14/21 00z	
	GFS	NAM	GFS	NAM	GFS	NAM	GFS	NAM	GFS	NAM
Temperature Bias (°F)	-0.0261	-0.5705	0.9162	-0.4719	2.3003	1.3533	4.3871	3.4004	1.0607	0.8510
Temperature MAE (°F)	3.2489	2.4949	5.6106	2.7930	3.3781	2.1797	4.6625	3.5468	2.3692	1.6095
Relative Humidity Bias (%)	1.0427	1.3767	-10.1897	-8.3817	-9.0276	-6.4272	-13.7592	-15.3634	-14.2454	-11.4004
Relative Humidity MAE (%)	6.5315	6.3682	13.3682	11.0527	10.5734	7.9272	14.9991	16.9327	15.4214	12.6115

Table 5.4: TMAE and Cumulative Bias Values for all forecast runs.

0°F by the end of the simulation.

On the 12z run for January 6, 2021, here the roles have swapped in that the bias and error of the GFS are minimal and compared to the NAM as the GFS followed very close to what was observed as shown in Figure 5.4. The bias (MAE) of the NAM reached a maximum of 3.2°F (5.1°F) while the GFS reached a maximum of 2.6°F (1.5°F).

The 12z run on January 7, 2021 as shown in Figure 5.5 is an interesting model run. Both the GFS and the NAM were in agreement with each other for this day, but after the 12^{th} forecast hour, the models both under perform as they indicated a significant cool down after dark, but the rate that the observed temperatures fell was not as fast, only reaching a minimum of around 40°F. The bias and MAE indicate the closeness of the two models, but both reach a maximum error rate of around 5.3°F.

The 00z run on January 10, 2021 includes the major snow event that occurred in East Texas. The snow started falling during the afternoon period, which would have been towards the end of the run, around 18z. The 2 meter temperatures reflect the difference of snow versus no snow. For instance, shown in Figure 5.6 The models were in agreement at the start of the run, however they start to overestimate the

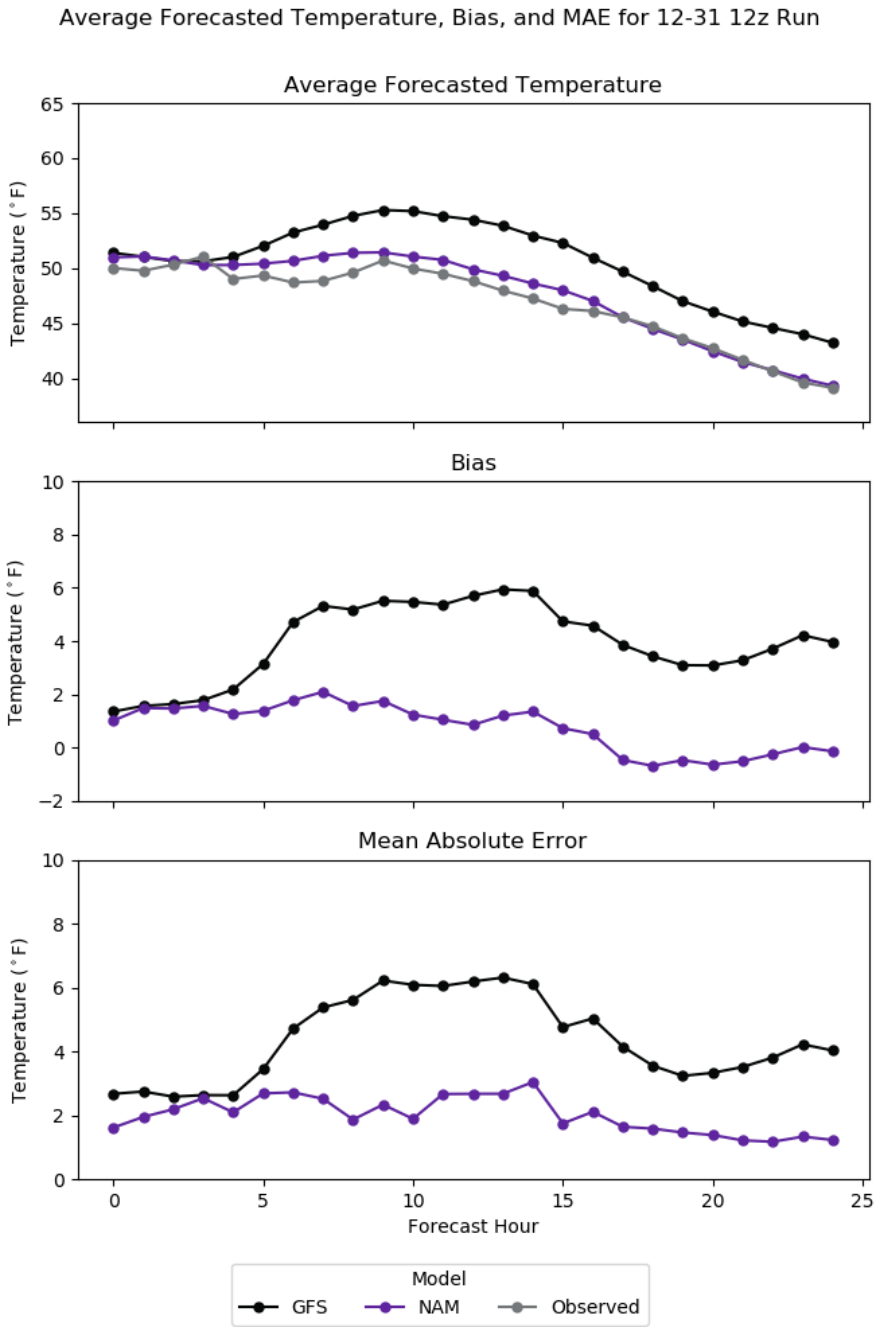

Figure 5.3: Averaged Forecasted Temperatures (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 12-31 12z run of the WRF.

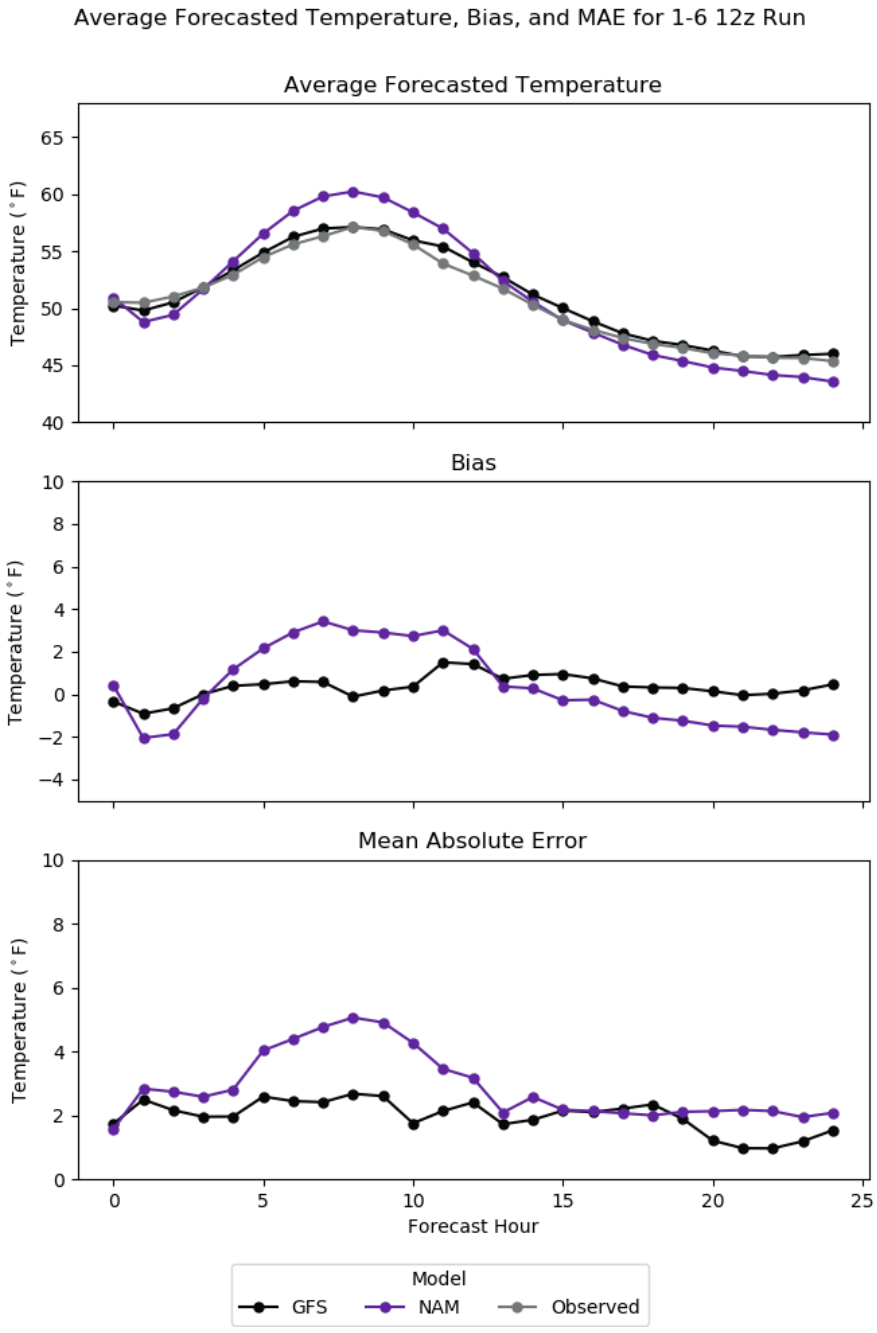

Figure 5.4: Averaged Forecasted Temperatures (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 1-6 12z run of the WRF.

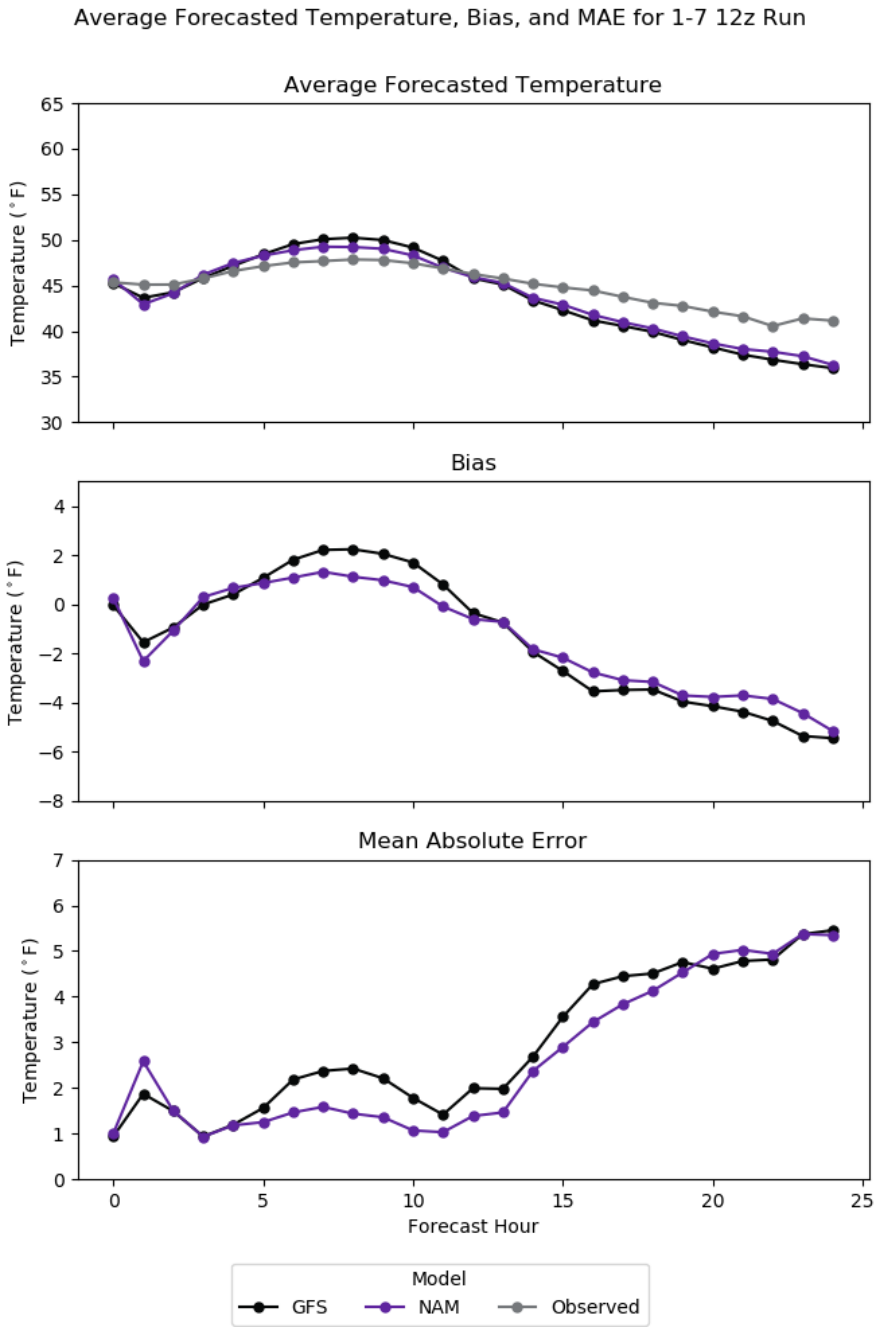

Figure 5.5: Averaged Forecasted Temperatures (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 1-7 12z run of the WRF.

temperatures as the snow started to fall during the last 6 hours of the run. The overestimation was in the NAM's favor by only reaching a maximum bias of 2.9°F while the GFS reached a higher bias of 3.9°F. The MAE of the GFS reached a maximum of 4.0°F, while the NAM only reached a maximum of 2.6°F.

The first simulation that the GFS and NAM were in agreement of the observed was the 12z run on January 25, 2021. The bias for the runs stayed within 2°F above and below zero, and the MAE of both reached a maximum of $4.4 - 4.6$°F shown in Figure 5.7.

Next is the 12z run on February 8, 2021. Like the previous simulation, the GFS and NAM were mostly in agreement with each other, and at times, switch between above the observed and below. The NAM stayed mostly below up until the end where it had a period of time that it was above the observed, but the GFS stayed just on par with the observed for most of the run, both shown in Figure 5.8. This trend is shown in the bias and MAE with the bias of both staying 3°F above and below zero, and the MAE of both reaching a maximum of around 5°F.

The 12z run the next day on February 9, 2021 has the runs not in agreement with each other no longer. At the beginning of the run, the GFS was well below the observed, and stayed that way until the sixth forecast hour, where as the NAM stayed below the observed until the very end. Both the GFS and the NAM did not account the rate at which the temperatures fell after sunset as, similar to the the run on January 7^{th}, but the NAM was still close to the observed throughout the entire run as indicated by the bias of the NAM only staying within 2°F above and below zero and only reaching a maximum error rate of around 4.8°F, while the GFS reached a maximum error of 7.2°F.

The 12z run on February 10, 2021 saw a repeat of February 8^{th}, that the models were in close agreement with each other, however as shown in Figure 5.10 the models overestimated again of what was observed and always had a bias greater than zero

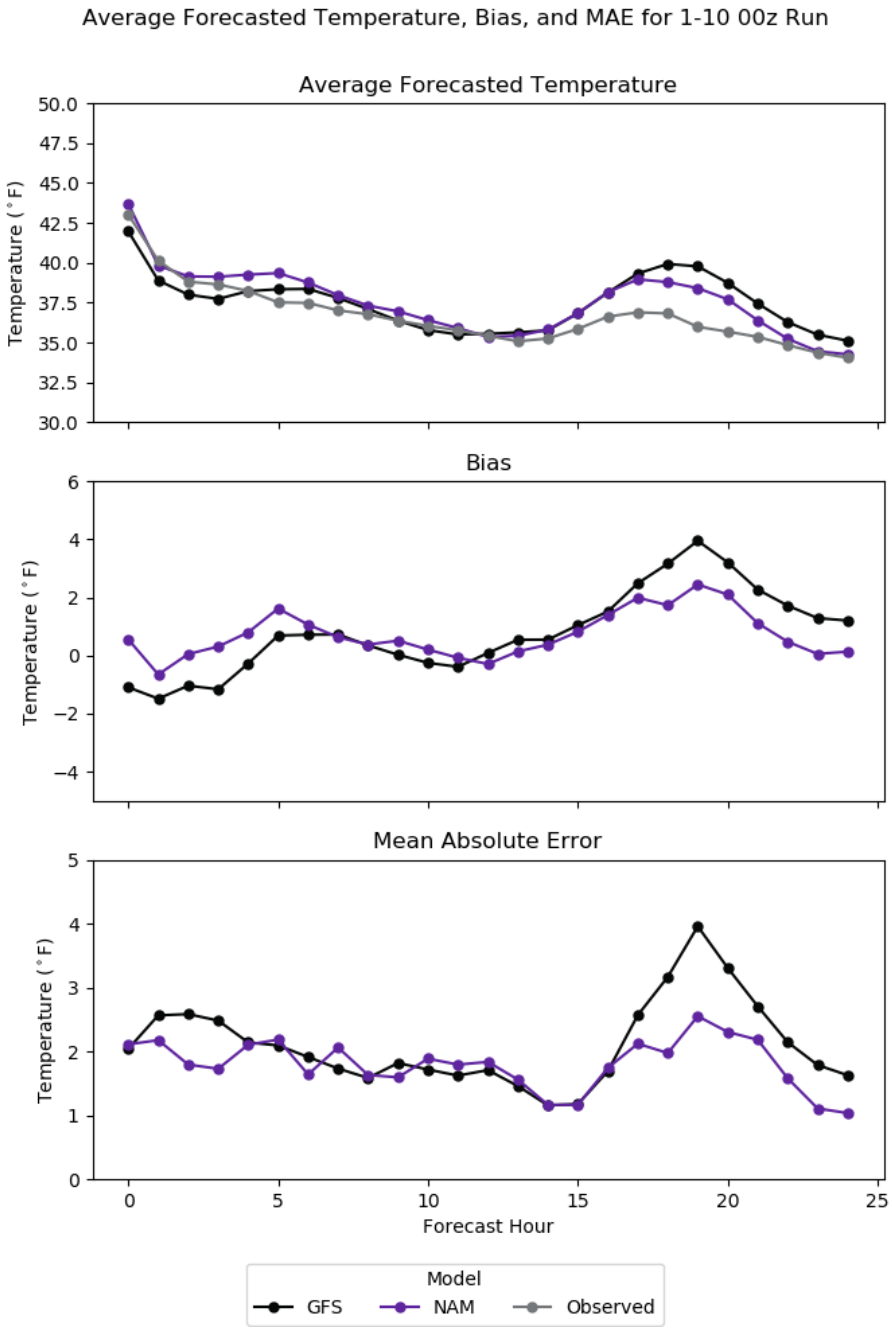

Figure 5.6: Averaged Forecasted Temperatures (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 1-10 00z run of the WRF.

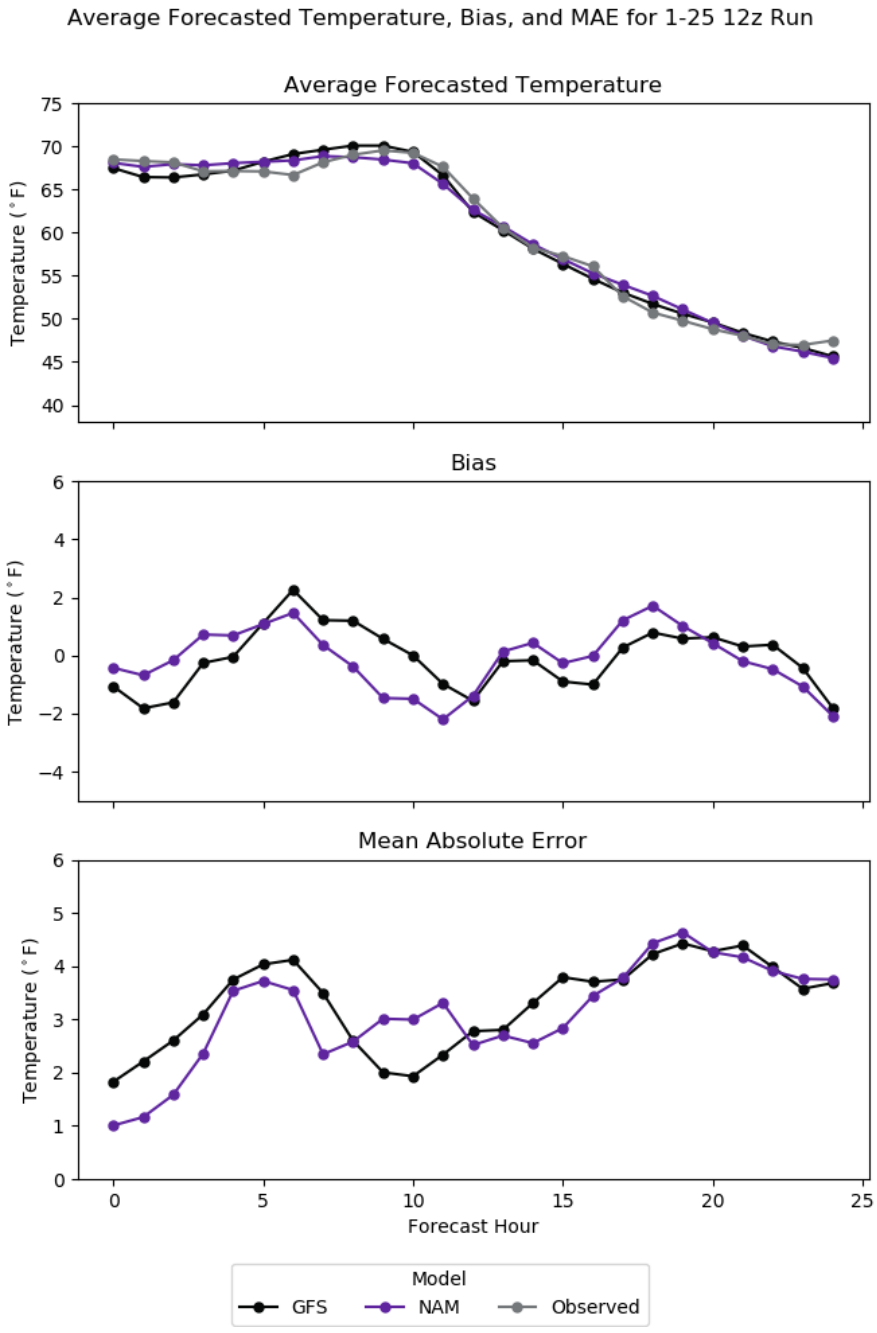

Figure 5.7: Averaged Forecasted Temperatures (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 1-25 12z run of the WRF.

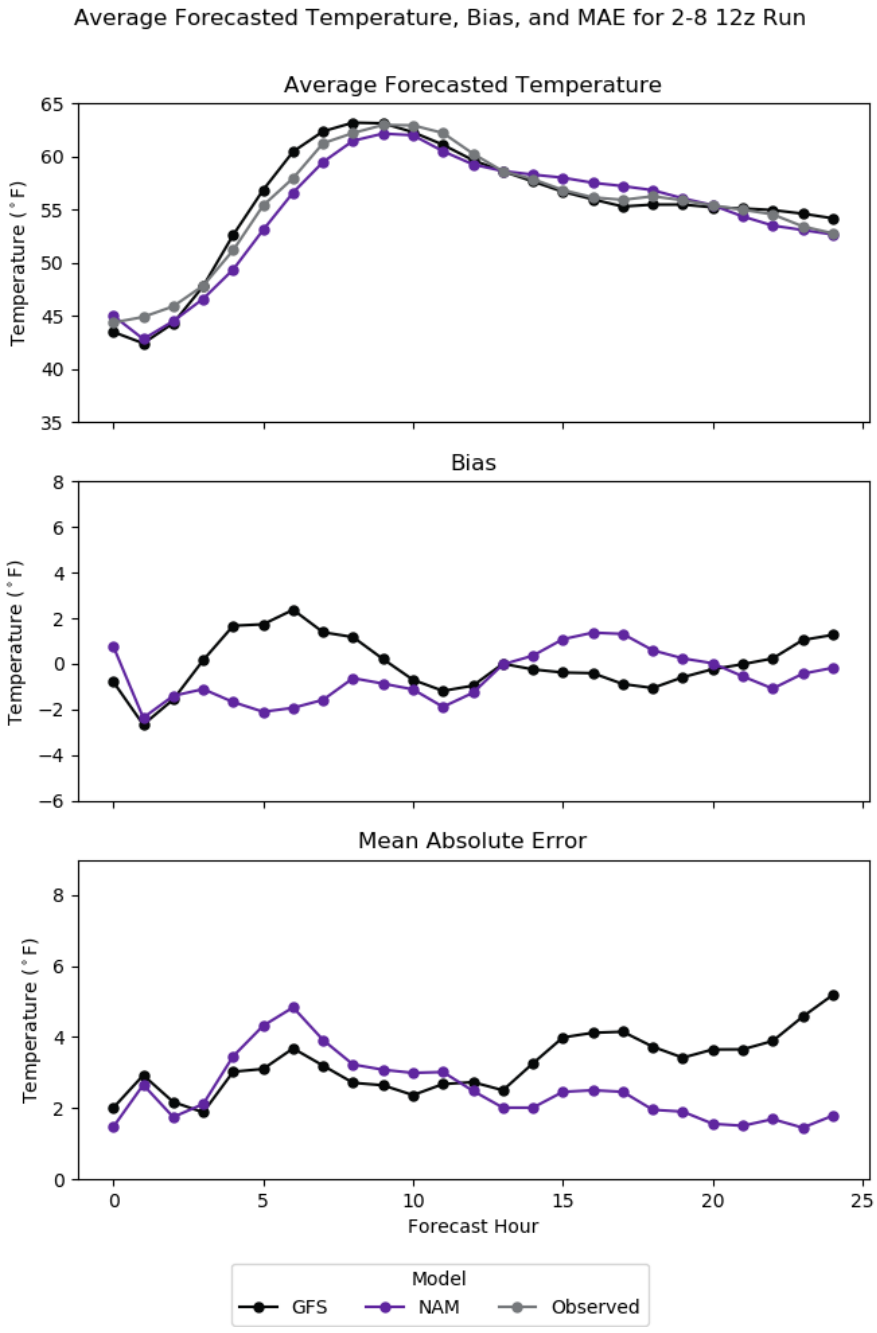

Figure 5.8: Averaged Forecasted Temperatures (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 2-8 12z run of the WRF.

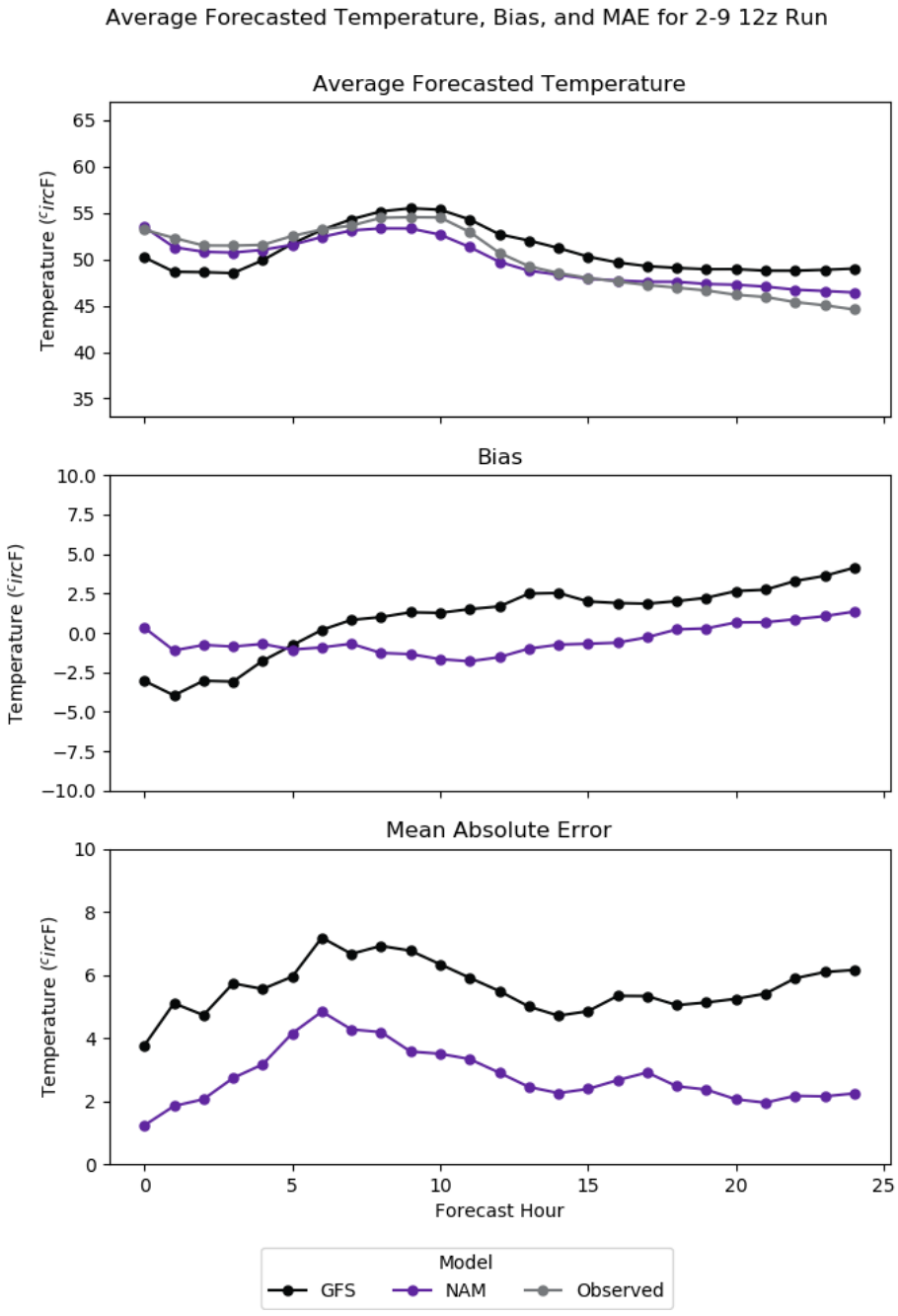

Figure 5.9: Averaged Forecasted Temperatures (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 2-9 12z run of the WRF.

and saw maximum MAEs of 5.0° in the GFS and 3.0°F in the NAM.

The same happens again for the 12z run on February 11, 2021 where both GFS and NAM were in agreement, but after the first three forecast hours, the runs estimated that the temperatures would rise during the day, however, due to clouds, rain, and freezing precipitation, the temps only fell during the day and into the night. This is shown in Figure 5.11. The bias and MAE of both reached a maximum of 7°F, one of the largest departures from the observed in the set of model runs.

The final simulation for this paper was the 00z run on February 14, 2021. This run included the start of a multi day snow and ice event for the area. Both the GFS and the NAM were in agreement with each other again and were also in agreement of what was observed shown in Figure 5.12. The bias of the NAM was tighter as it stayed mostly around 1°F with a couple of points around 2°F, while the spread of the bias for the GFS was higher, but only reaching a maximum of around 4°F and a minimum of around -1°F. Likewise the MAE for both tell the same story, with the GFS seeing the maximum value between the two, and the NAM values staying below the GFS values.

5.4.2 2 Meter Relative Humidity

In the ten simulations mentioned in the previous subsection, the temperature is one of the values that directly come from the approximations set of the equations mentioned previously in Chapter 3. While this paper does not talk about the runs individually, it can be seen in Figures 5.13 - 5.22 that the WRF did not forecast the relative humidity very accurately regardless if the input was from the NAM or from the GFS. As noted by Coaouch et al., relative humidity is not an output of the WRF, but is inferred, or calculated, from the water vapor mixing ratio, temperature, and surface pressure [3]. Here it can be said that most of the time, the WRF will

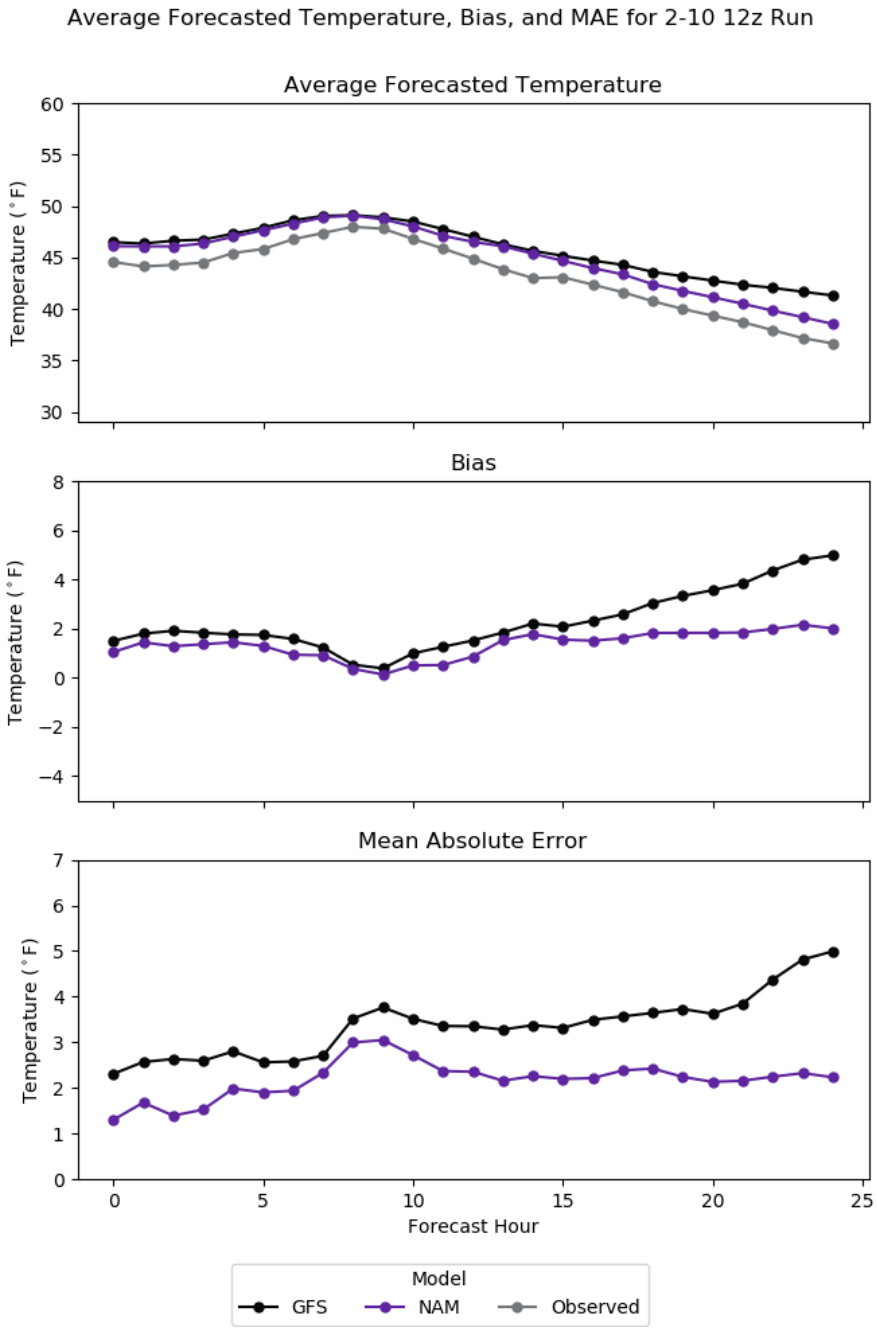

Figure 5.10: Averaged Forecasted Temperatures (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 2-10 12z run of the WRF.

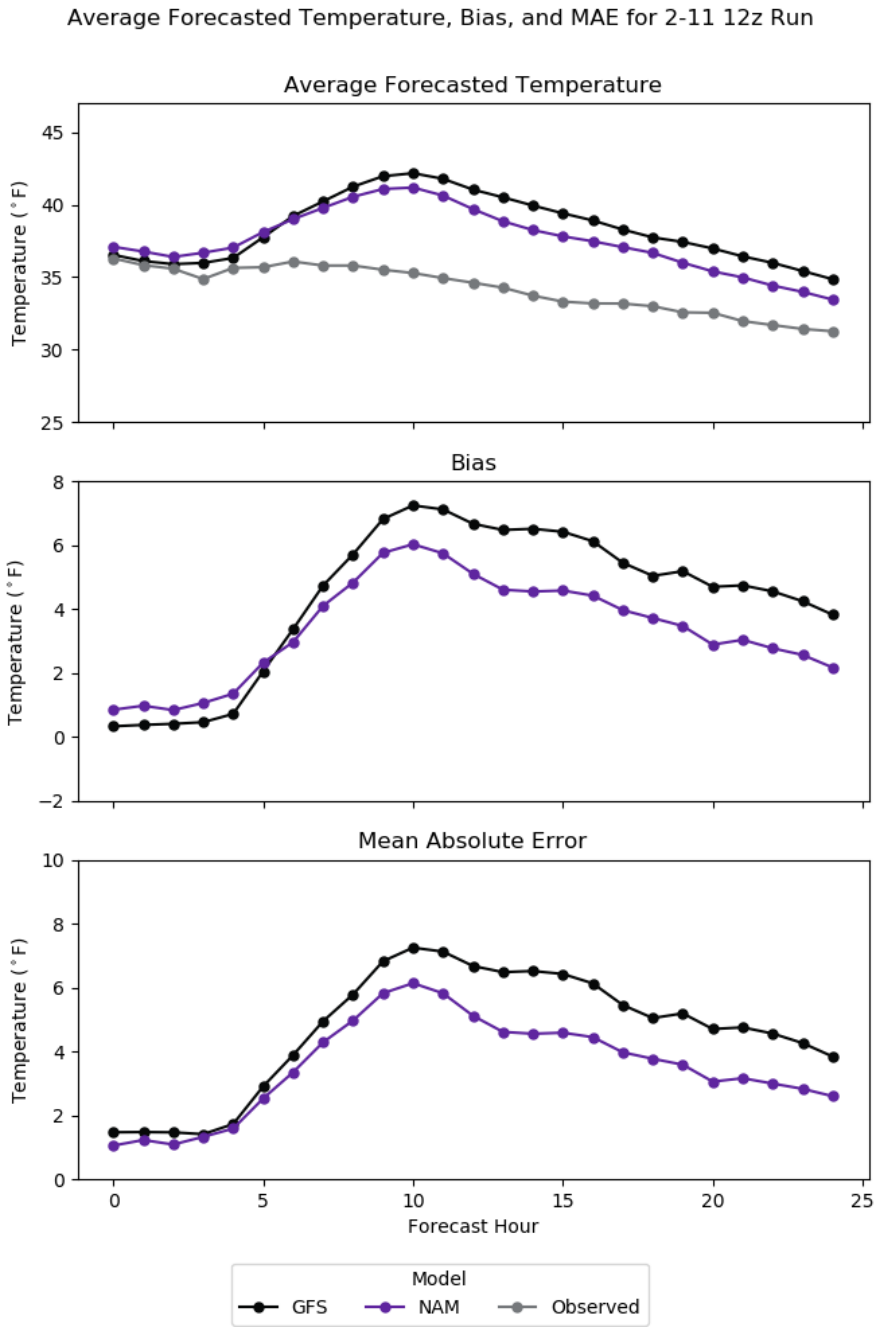

Figure 5.11: Averaged Forecasted Temperatures (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 2-11 12z run of the WRF.

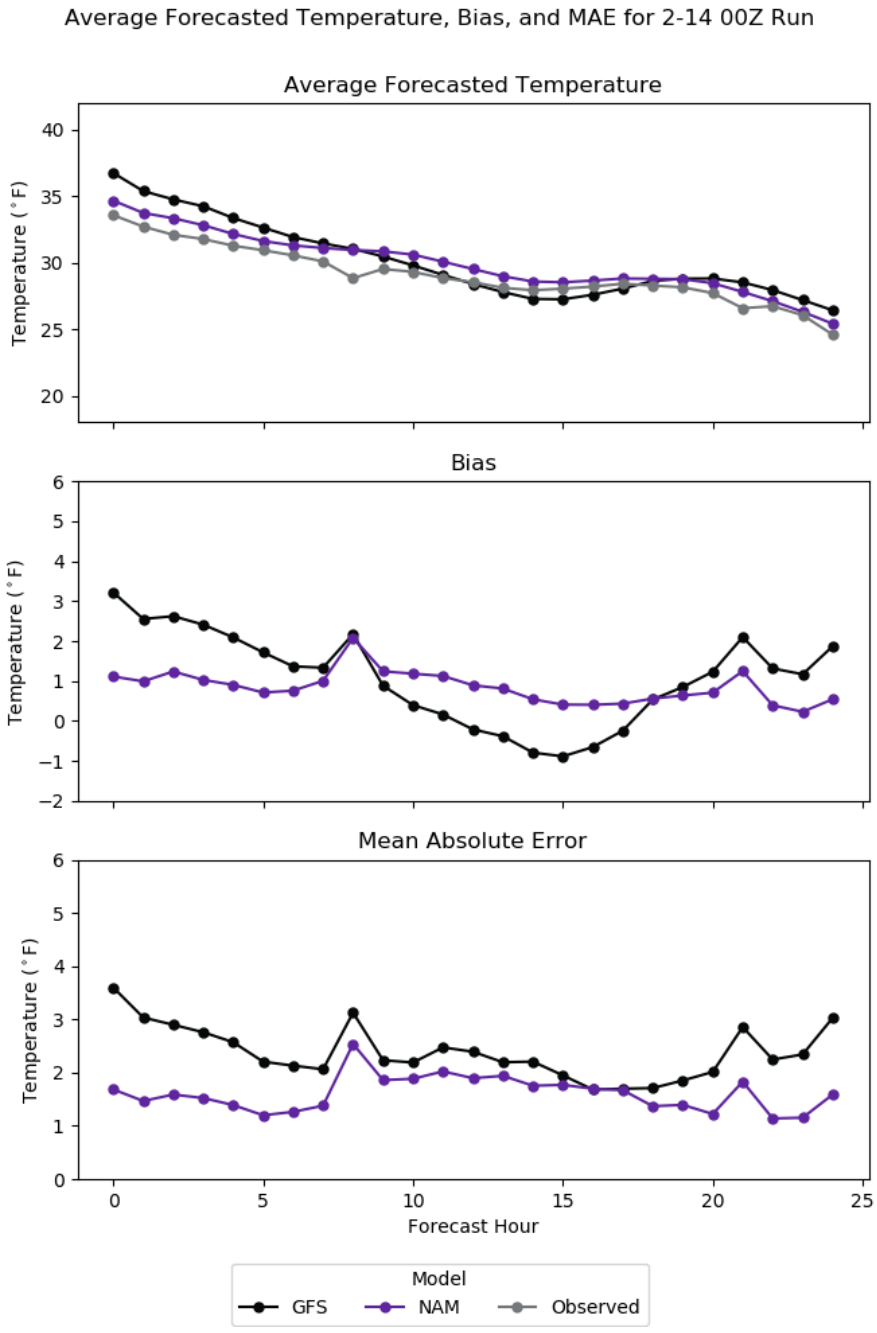

Figure 5.12: Averaged Forecasted Temperatures (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 2-14 00z run of the WRF.

bias towards a lower relative humidity as there are multiple runs where the forcasted values from both GFS and NAM are significantly below what was observed and can see MAEs over 20%.

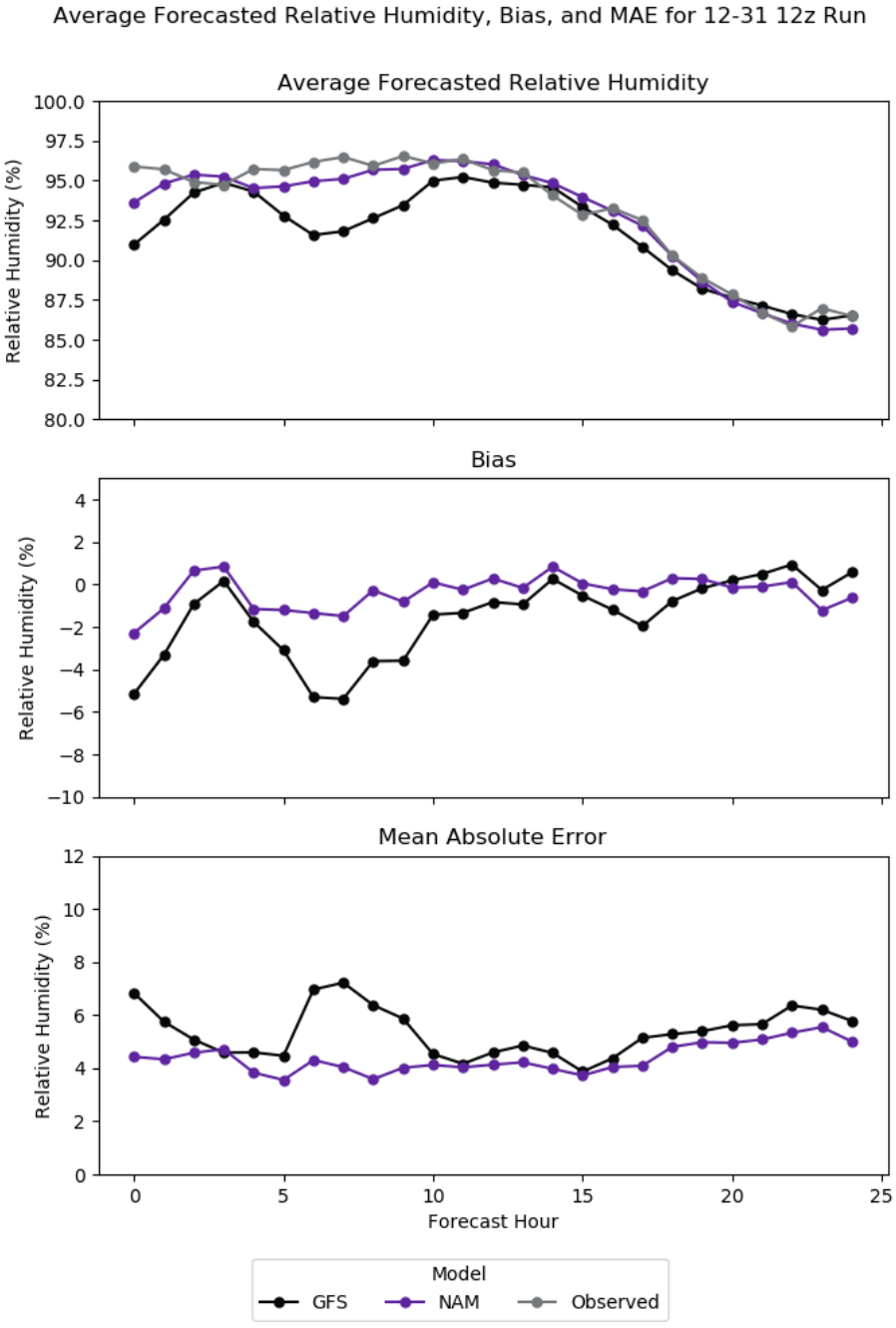

Figure 5.13: Averaged Forecasted Relative Humidity (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 12-31 12z run of the WRF.

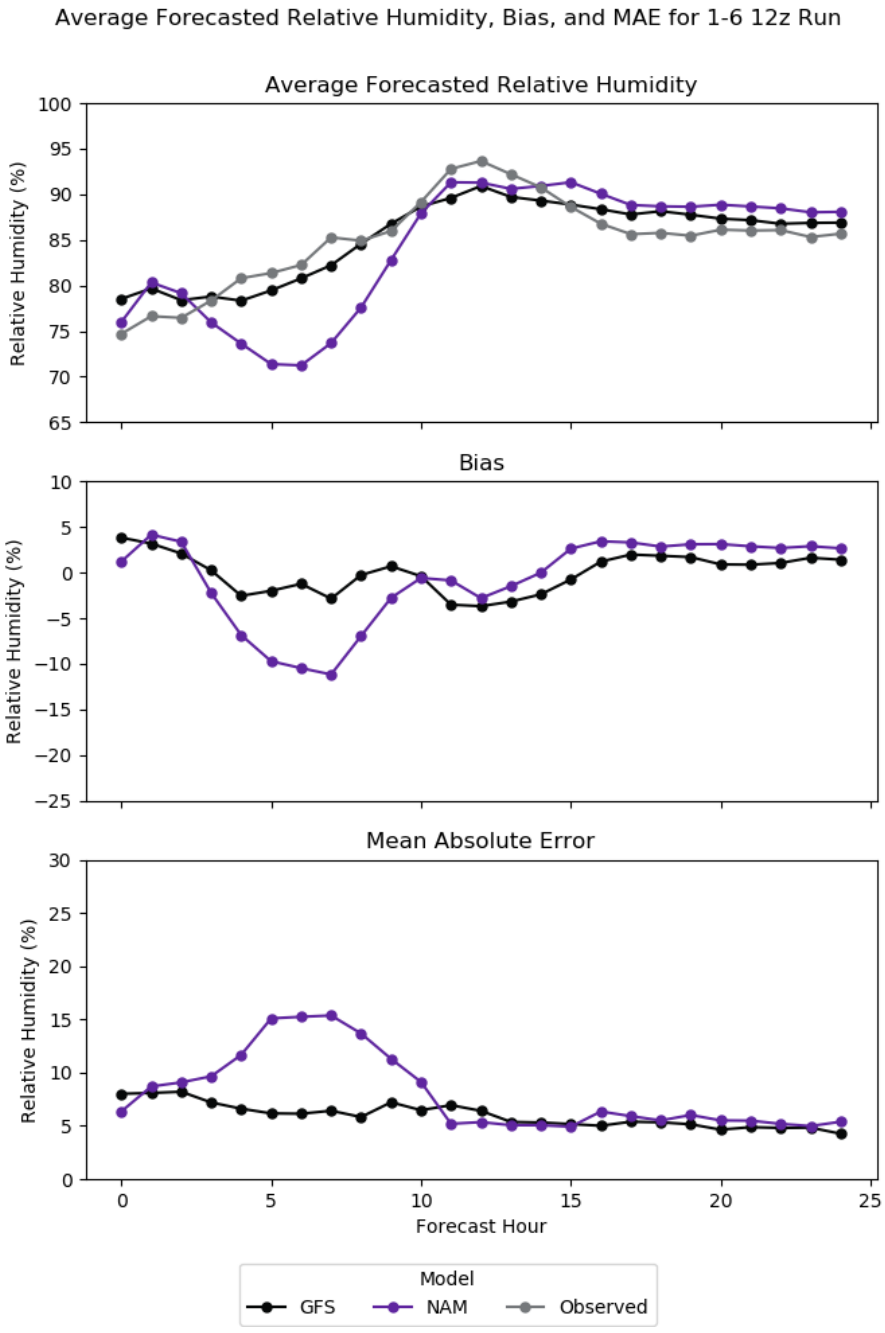

Figure 5.14: Averaged Forecasted Relative Humidity (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 1-6 12z run of the WRF.

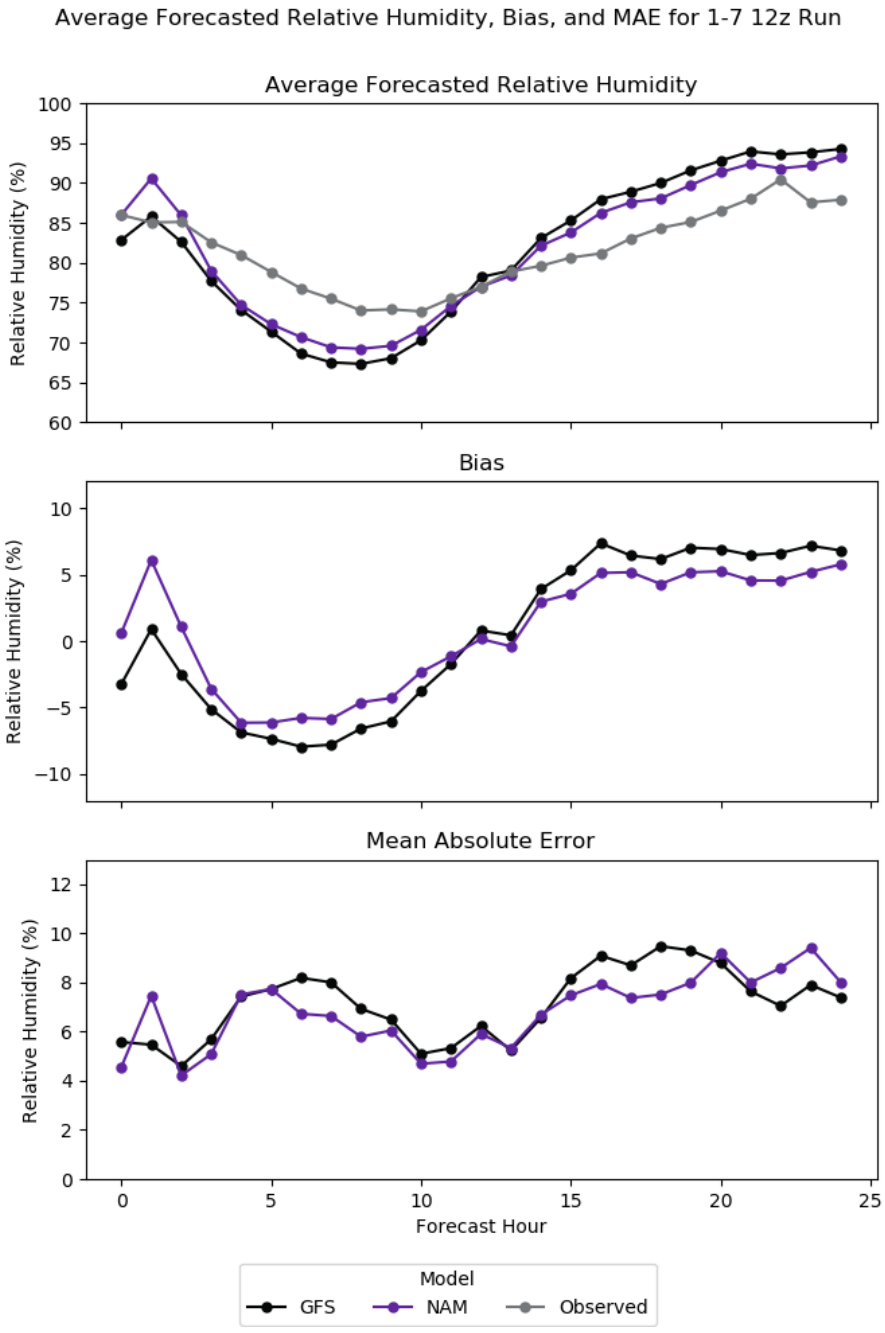

Figure 5.15: Averaged Forecasted Relative Humidity (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 1-7 12z run of the WRF.

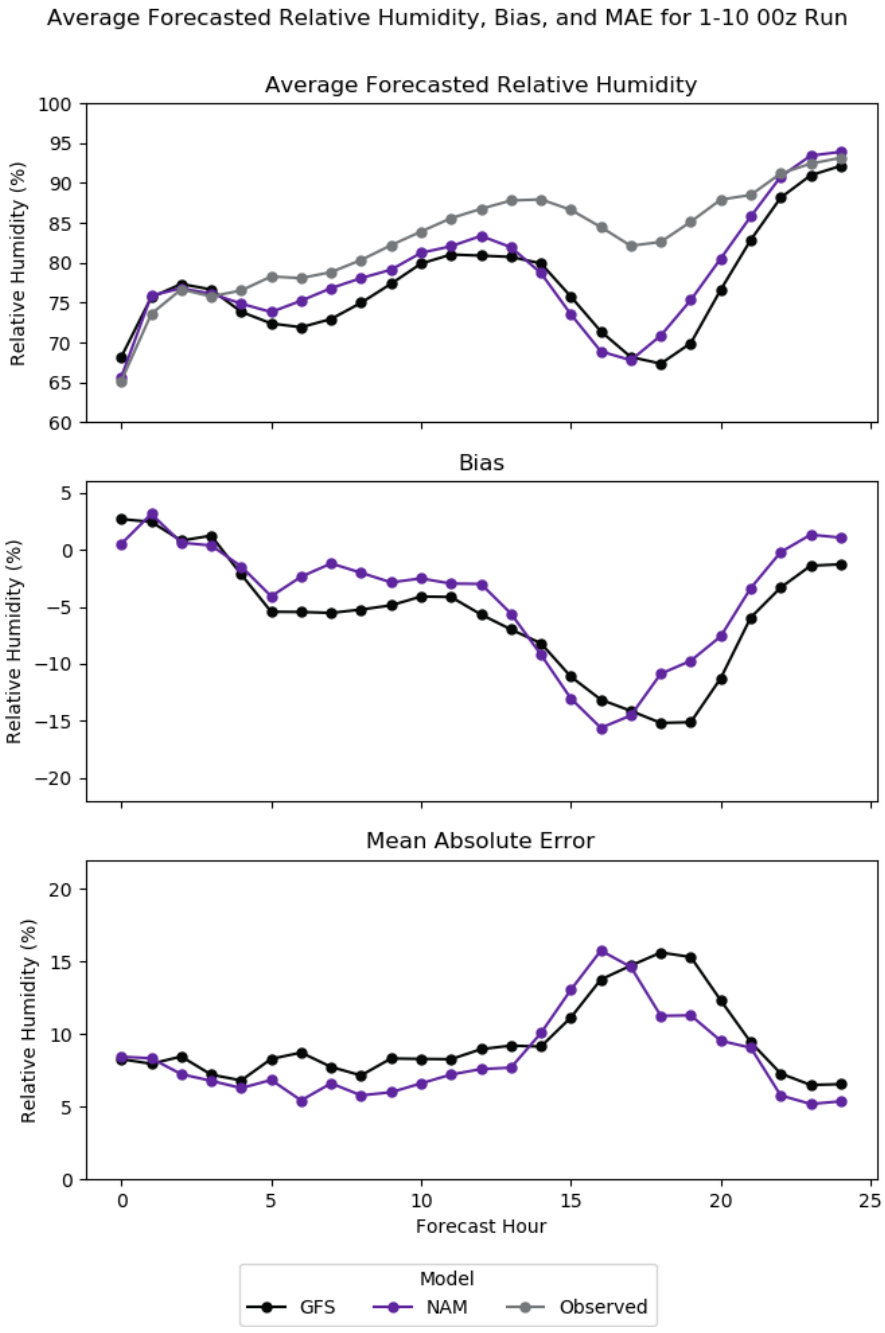

Figure 5.16: Averaged Forecasted Relative Humidity (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 1-10 00z run of the WRF.

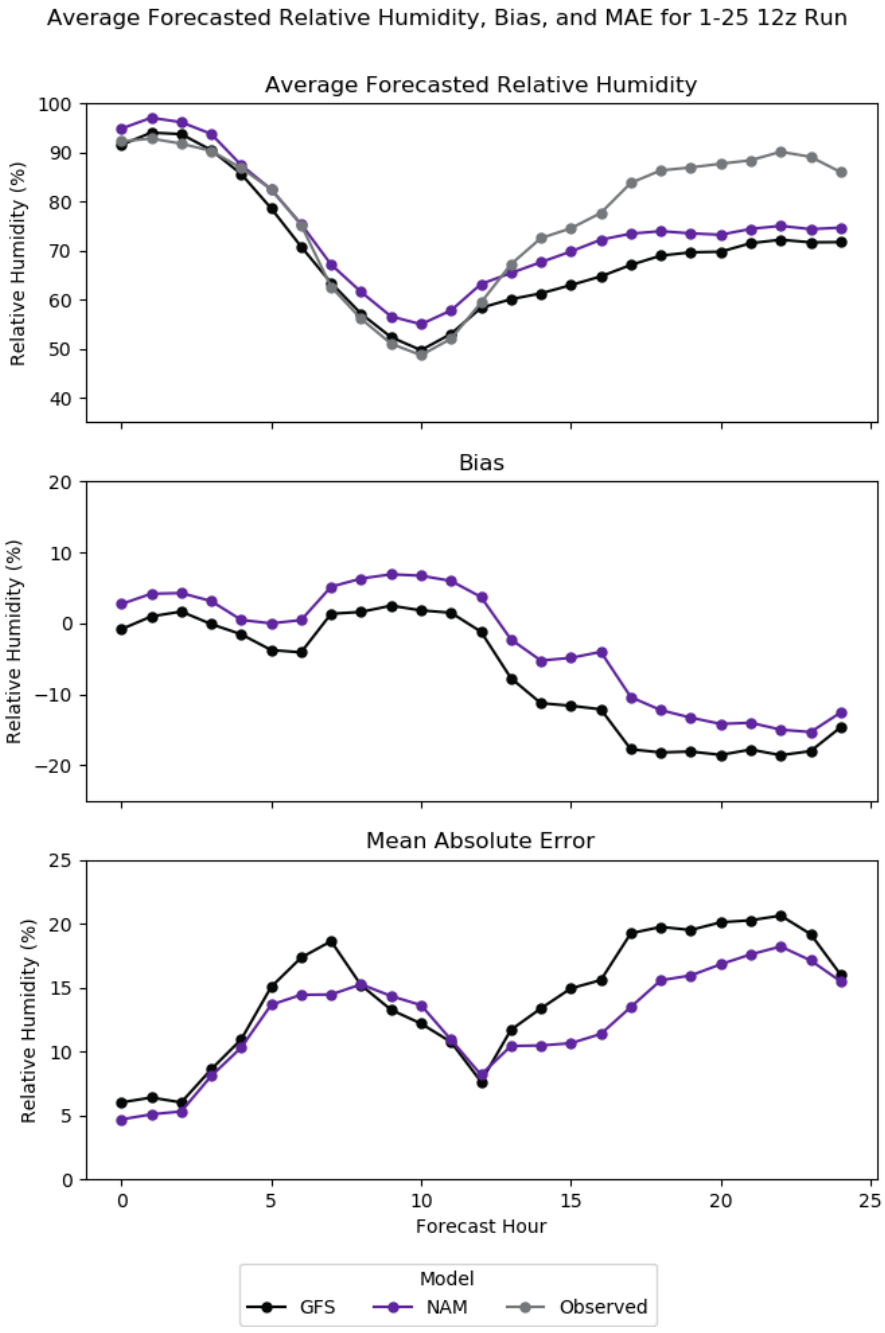

Figure 5.17: Averaged Forecasted Relative Humidity (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 1-25 12z run of the WRF.

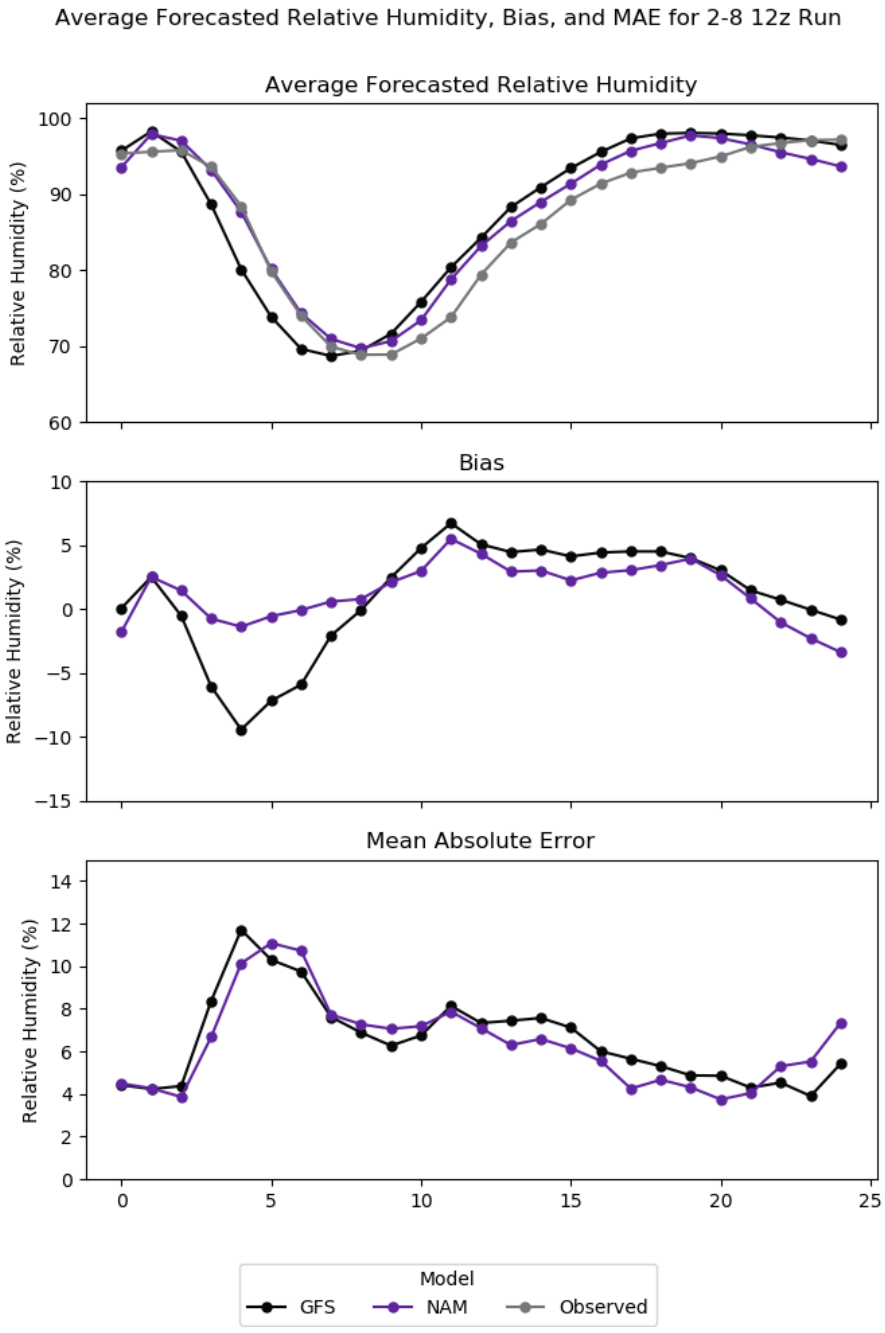

Figure 5.18: Averaged Forecasted Relative Humidity (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 2-8 12z run of the WRF.

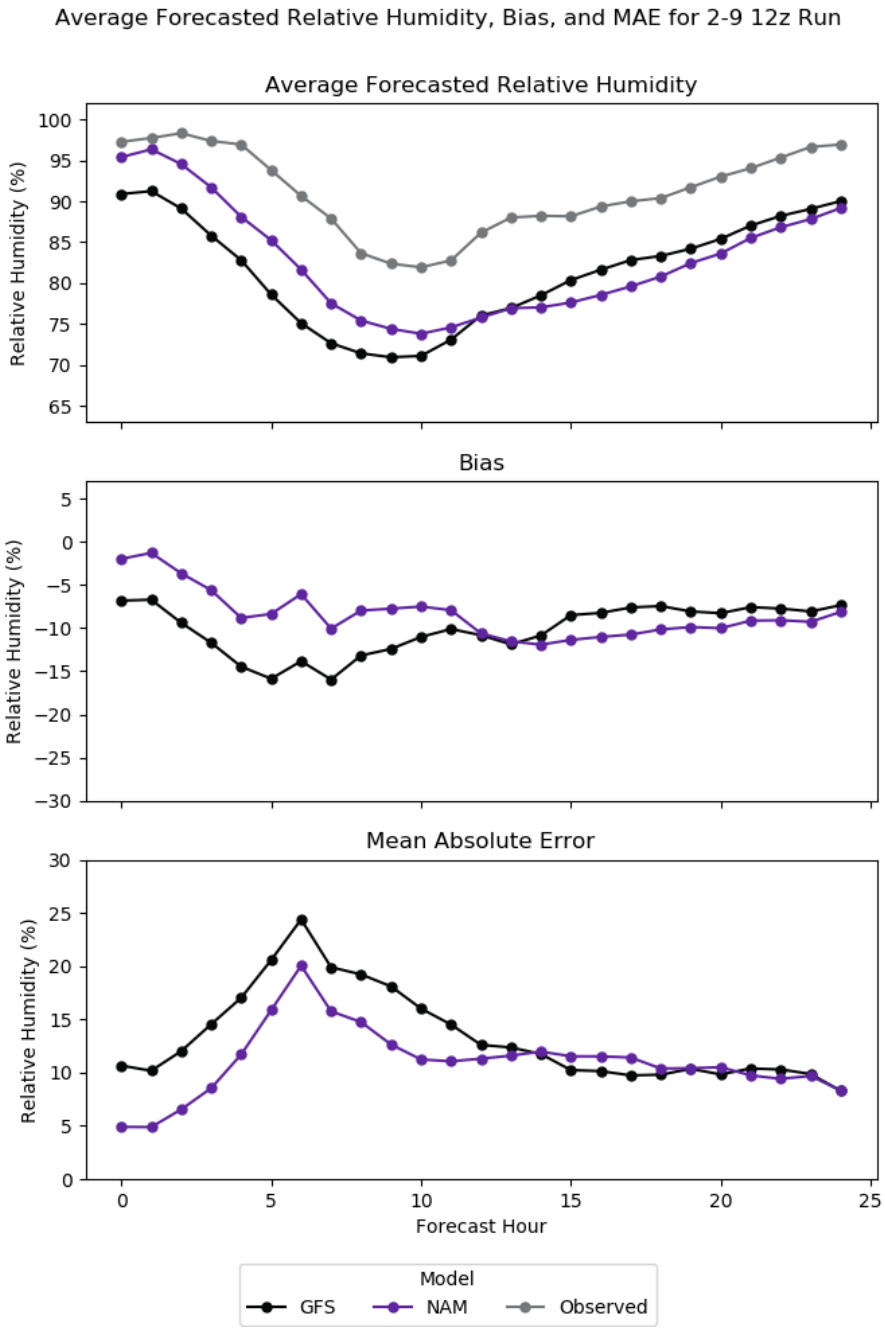

Figure 5.19: Averaged Forecasted Relative Humidity (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 2-9 12z run of the WRF.

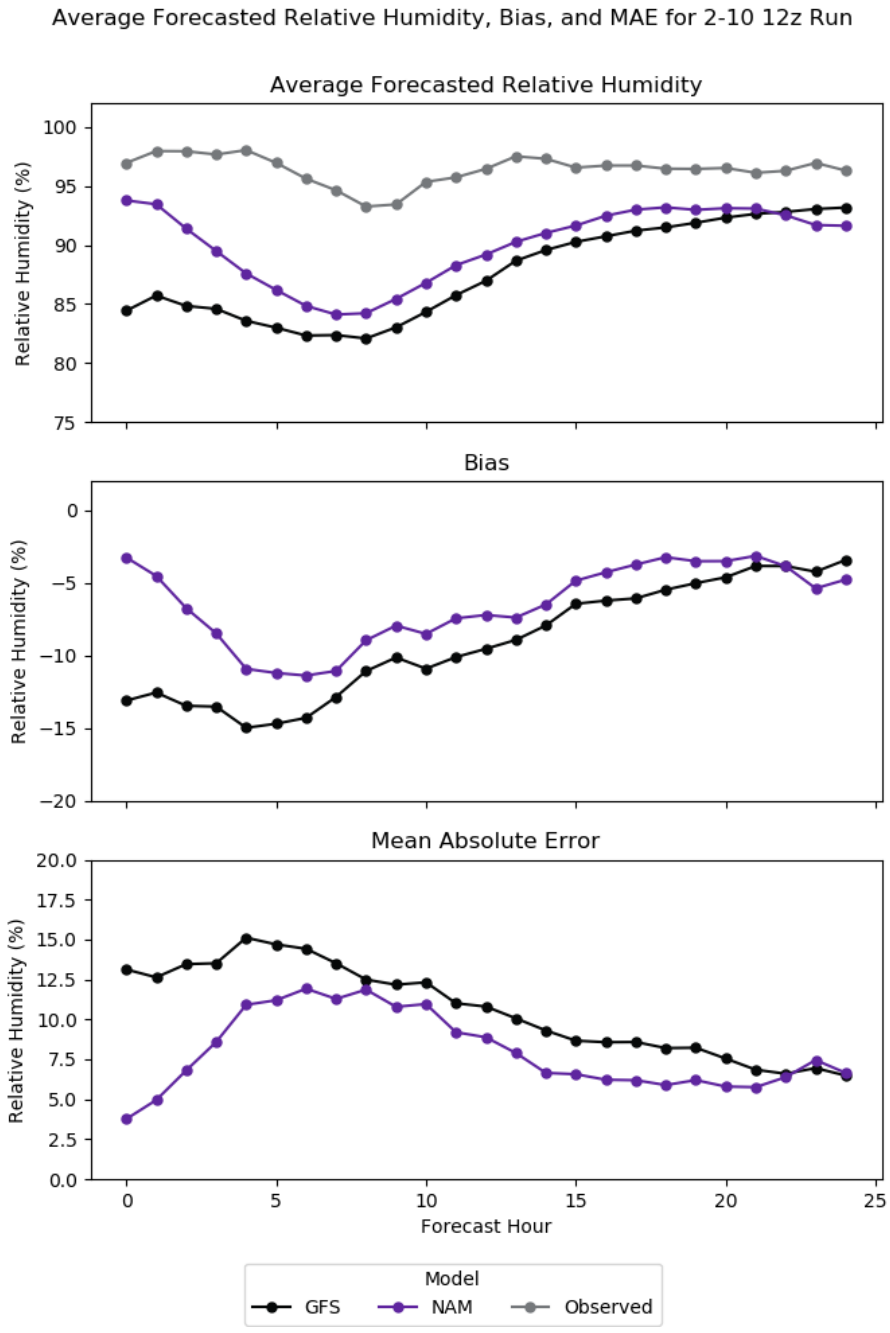

Figure 5.20: Averaged Forecasted Relative Humidity (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 2-10 12z run of the WRF.

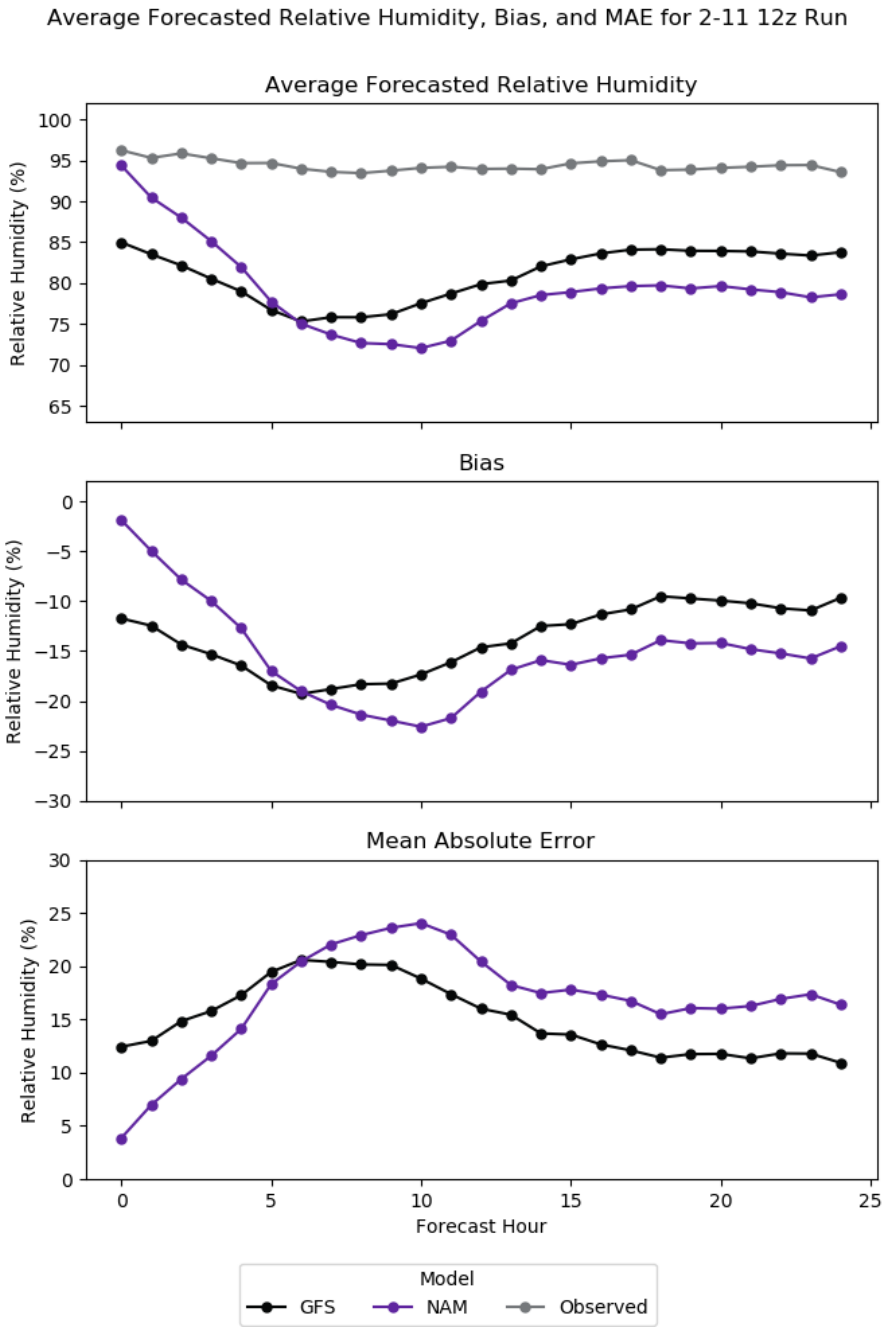

Figure 5.21: Averaged Forecasted Relative Humidity (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 2-11 12z run of the WRF.

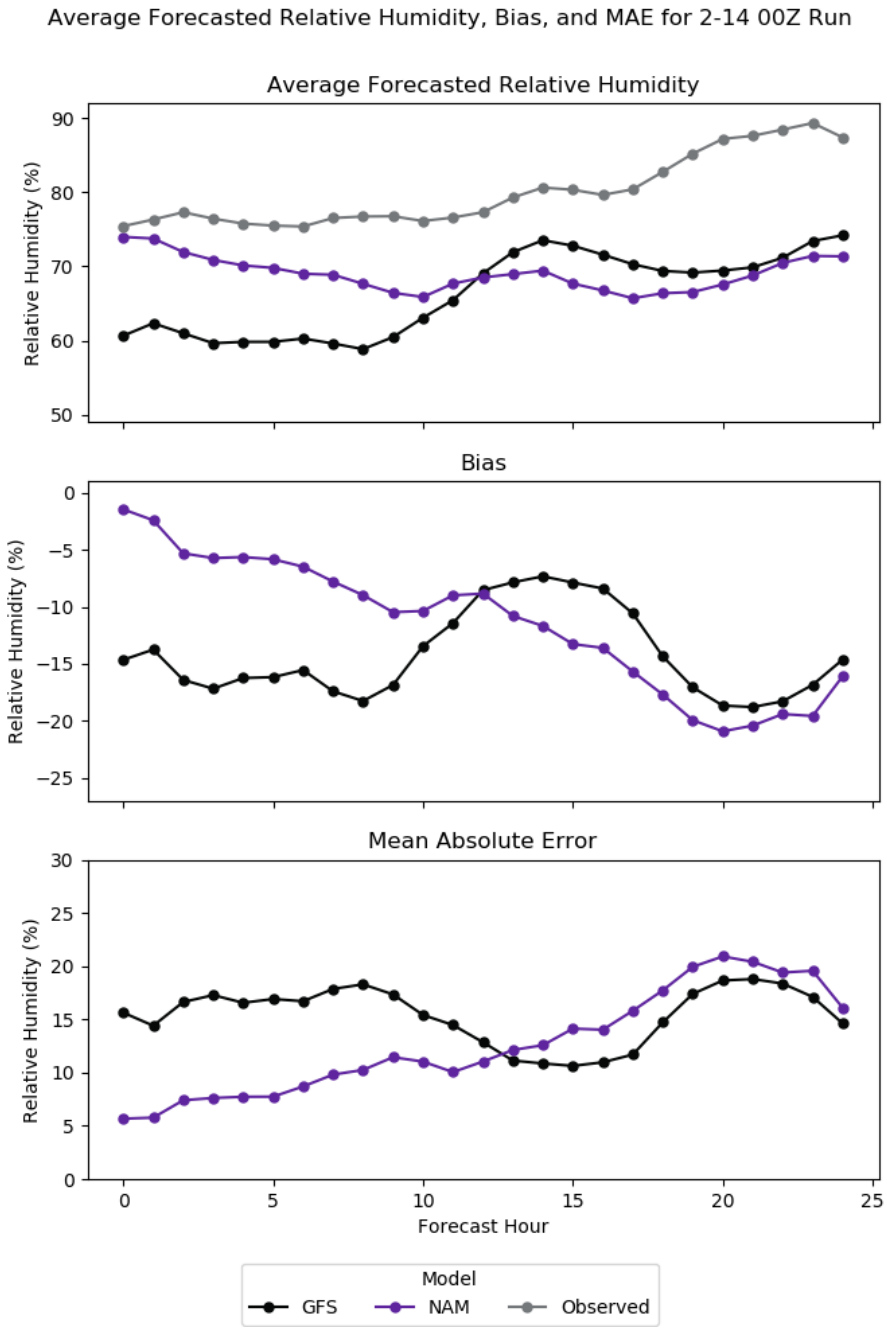

Figure 5.22: Averaged Forecasted Relative Humidity (First row), Bias (Second row), and Mean Algebraic Error (Third row) from the 2-14 00z run of the WRF.

6 CONCLUSION

The purpose of this project was to evaluate how the WRF model behaved when given different inputs. Through the ten model runs, we were able to see the differences in output when using inputs from the NAM model and the GFS model within a forecast run of 24 hours. In each run, the GFS and NAM inputs were fairly accurate in the runs for the 2 meter temperature, with one being slightly more accurate than the other, and often saw both models either under or over perform of what was actually observed. When it came to overall performance, the inputs from the NAM model had lower error compared to the GFS model for eight of the ten runs, with the GFS input being lower in a single run, and both tied in another instance as well.

However, when it came to forecasting the 2 meter relative humidity, we noticed that since this was not a product from the direct output of the system of equations like the temperature, the accuracy of the WRF was not good at all. Some runs were able to have errors less than 6%, but there were also several runs that saw errors over 20%.

For future work, since we now understand how the model works for a forecast of 24 hours, is to observe how the WRF behaves when given a longer forecast period such as 48 or even 72 hours. Another angle to research is other variables that are a direct product of the output, such as wind speed and direction, and other upper-air variables. This will help to better decide on which model input may be more accurate in determining forecasts over the given area.

Bibliography

[1] Cleveland Abbe, *The physical basis of long-range weather forecasts*, Monthly Weather Review **29** (01 Dec. 1901), no. 12, 551 – 561.

[2] Vilhelm Bjerknes, *The problem of weather forecasting as a problem in mechanics and physics*, American Meteorological Society, Boston, MA, 1999.

[3] Naira Chaouch, Marouane Temimi, Michael Weston, and Hosni Ghedira, *Sensitivity of the meteorological model wrf-arw to planetary boundary layer schemes during fog conditions in a coastal arid region*, Atmospheric Research **187** (2017), 106–127.

[4] NCAR Computational and Information Systems Lab, *Raspberry Pi Weather Forecast*, `https://www2.cisl.ucar.edu/docs/raspberry-pi-weather-forecast-0`, Accessed: 2021-01-22.

[5] Samantha Durbin, *What are weather models, exactly, and how do they work?*, Washington Post (2018).

[6] Stanley J. Farlow, *Partial Differential Equations for Scientists and Egineers*, dover Publications Inc., 1993.

[7] Mike Farrar, *Upgrade NCEP global forecast systems (GFS) to v16 effective march 17, 2021*, `https://www.weather.gov/media/notification/scn_21-20_gfsv16.0_aaa_update.pdf`, 2 2021.

[8] Peter Lynch, *The origins of computer weather prediction and climate modeling*, Journal of Computational Physics **227** (2008), no. 7, 3431–3444, Predicting weather, climate and extreme events.

[9] Ryan M. May, Sean C. Arms, Patrick Marsh, Eric Bruning, John R. Leeman, Kevin Goebbert, Jonathan E. Thielen, and Zachary S. Bruick, *Metpy: A Python package for meteorological data*, 2020.

[10] National Weather Service Science and Training Resource Center, *STRC UEMS Home Page*, `http://strc.comet.ucar.edu/software/uems/`, Accessed: 2020-10-21.

[11] Jordan G. Powers, Joseph B. Klemp, William C. Skamarock, Christopher A. Davis, Jimy Dudhia, David O. Gill, Janice L. Coen, David J. Gochis, Ravan Ahmadov, Steven E. Peckham, Georg A. Grell, John Michalakes, Samuel Trahan, Stanley G. Benjamin, Curtis R. Alexander, Geoffrey J. Dimego, Wei Wang, Craig S. Schwartz, Glen S. Romine, Zhiquan Liu, Chris Snyder, Fei Chen, Michael J. Barlage, Wei Yu, and Michael G. Duda, *The weather research and forecasting model: Overview, system efforts, and future directions*, Bulletin of the American Meteorological Society **98** (01 Aug. 2017), no. 8, 1717 – 1737.

[12] William C. Skamarock, Joseph B. Klemp, Jimy Dudhia, David O. Gill, Zhiquan Liu, Judith Berner, Wei Wang, Jordan G. Powers, Michael G. Duda, Dale M. Barker, and Xiang-Yu Huang, *A Description of the Advanced Research WRF Model Version 4. ncar technical note -556+str*, National Center for Atmospheric Research, Boulder, Colorado, 2019.

[13] John P. Snyder and Philip M. Voxland, *An album of map projections: Professional paper 1453*, 1989.

[14] Henry R. Stanski, Laurence J. Wilson, and William R. Burrows, *Survey of common verification methods in meteorology*, Atmospheric Environment Service Forecast Research Division, Downsview, Ontario, 2 ed., July 1989.

[15] Edmund P. Willis and William H. Hooke, *Cleveland abbe and american meteorology, 1871–1901*, Bulletin of the American Meteorological Society **87** (01 Mar. 2006), no. 3, 315 – 326.

[16] Hailing Zhang, Zhaoxia Pu, and Xuebo Zhang, *Examination of errors in near-surface temperature and wind from wrf numerical simulations in regions of complex terrain*, Weather and Forecasting **28** (01 Jun. 2013), no. 3, 893 – 914.

VITA

Aaron Baker attended Stephen F. Austin State University in August 2012 and received a Bachelor of Science degree in Computer Science in December 2016 followed by a Bachelor of Science degree in Mathematics in May 2018. Aaron then started full-time graduate work in August 2018 and is expected to graduate with a Master of Science degree in Mathematics in May 2021.

The style manual used in this thesis is A Manual For Authors of Mathematical Papers published by the American Mathematical Society.

This thesis was prepared by Aaron Baker using LaTeX.