Model Validation of the Thermospheric Neutral Mass Density

By

Martin Geary McCandless

A Thesis
Submitted to the Faculty of
Mississippi State University
in Partial Fulfillment of the Requirements
for the Degree of Bachelor of Science
in Aerospace Engineering with an Aeronautics Concentration
in the Department of Aerospace Engineering

Mississippi State, Mississippi

May 2019

Name: Martin Geary McCandless

Date of Degree: May 3, 2019

Institution: Mississippi State University

Major Field: Aerospace Engineering with an Aeronautics Concentration

Major Professor: Dr. Keith Koenig

Title of Study: Model Validation of the Thermospheric Neutral Mass Density

Pages in Study: 61

Candidate for Degree of Bachelor of Science

Accurately predicting the thermospheric neutral mass density is crucial for estimating the trajectory of low Earth orbit (LEO) spacecraft since satellite drag introduces errors in orbit determination solutions for the rapidly increasing number of man-made objects. The purpose of this study is to quantify the degree of variability in the thermospheric neutral mass density during the 2013 St. Patrick's Day geomagnetic storm utilizing the coupled Whole Atmosphere Model and Ionosphere-Plasmasphere-Electrodynamics model (WAM-IPE). The neutral mass density variations for various geophysical conditions from WAM-IPE are compared between the accelerometer satellite observations from Gravity Field and Steady-State Ocean Circulation Explorer (GOCE) and the Coupled Thermosphere, Ionosphere, Plasmasphere, and electrodynamics (CTIPe) model. The results comparing WAM-IPE with GOCE and CTIPe suggest that the WAM-IPE model can capture normal diurnal/latitude neutral density structure as well as the response and recovery to the geomagnetic storm. With appropriate parameters in place, the results agree remarkably well with a standard deviation = 0.0917, a bias = 1.04, and a correlation coefficient = 0.949.

DEDICATION

To my parents for their love and support and to all of the great minds who contributed to this work.

ACKNOWLEDGEMENTS

This work has been supported by the funds from the NOAA Ernest F. Hollings Scholarship Program and from the Judy and Bobby Shackouls Honors College. I would like to express my gratitude to many people from the NOAA Space Weather Prediction Center and the University of Colorado Boulder CIRES for providing their service in making this effort possible.

To Professor Keith Koenig, for agreeing to serve as the major professor and offering unselfish guidance and mentorship throughout my undergraduate studies.

To my primary NOAA mentor, Dr. Naomi Maruyama, for her outstanding mentorship carried throughout this project and access to NCAR resources.

To Dr. Timothy Fuller-Rowell, for providing reference material and his intuitive explanations of space plasma physics were critical for a successful project.

To Dr. Mariangel Fedrizzi, who provided all necessary GOCE satellite data and programming guides for IDL and NCL along with conference presentation strategies.

To Dr. Zhuxiao Li, who provided all model outputs and her efforts for formatting the data.

To Dr. Rodney Viereck, for establishing the initial correspondence to obtain this position at NOAA/SWPC.

To Dr. Jamie Dyer and Dr. David Hoffman, for serving as the internal and Honors committee members.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

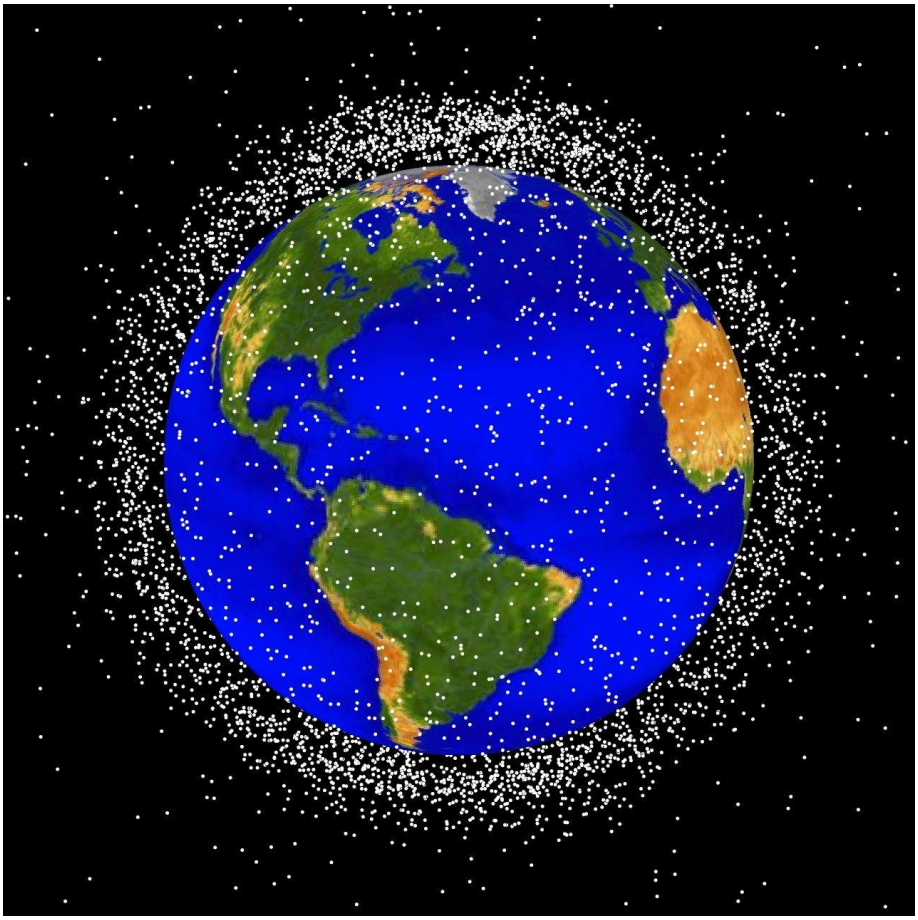
INTRODUCTION

1.1 Motivation

The growing importance of accurately tracking the orbits of artificial debris and satellites around Earth has been crucial in collision avoidance or re-entry predictions. More than 24,500 pieces of space debris have been cataloged since 1957 [1]. Many of these high Earth orbit (HEO) satellites have decayed into low Earth orbit (LEO). LEO satellites operate at an altitude less than 2,000 km above the Earth's surface. Currently, the U.S. Space Surveillance Network tracks and catalogs over 22,000 man-made objects that are greater than 10 cm in diameter. Debris sizes that are greater than 1 cm can be catastrophic to satellites because their total kinetic energy can severely compromise the structural integrity of satellites, leading to fragmentation of satellite components. Debris ranging from 1 mm to 1 cm can puncture fuel supply lines and radiative shielding surfaces that protect internal electronic components. Objects smaller than 1 mm risk performance degradation to satellite instrumentation such as optical lenses and gradual thermal surface erosion [2]. Figure 1.1 shows an orbit propagation model from the NASA Orbital Debris Program Office that tracks and catalogs LEO satellites.

Figure 1.1

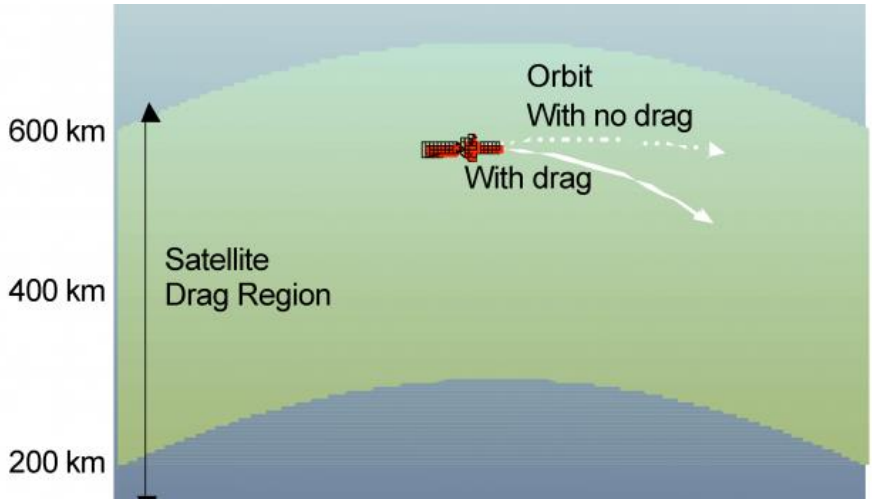
Computational model of LEO objects as the most highly populated area compared to GEO and HEO orbits (courtesy NASA Orbital Debris Program Office).



To track space debris for collision avoidance and re-entry predictions, orbit propagation models must consider natural forces that influence the orbit of LEO space objects. These perturbations include the irregular gravitational field acceleration, atmospheric drag, and electrodynamic forces. Of these natural forces, atmospheric drag is the most significant contributor to the uncertainty in orbit determination and satellite lifetime optimization. Drag is an aerodynamic force that is exerted on an object as it moves through a fluid. This force acts in the opposite direction of motion. Figure 1.2 shows orbit trajectories of a LEO satellite in the presents and absence of atmospheric drag. To reduce orbit trajectory uncertainty because of drag, atmospheric models are used to aid orbit propagation models by capturing various atmospheric processes that directly influence the space objects' orbit.

Figure 1.2

LEO satellite orbit trajectories in the region of Earth's atmosphere where drag is important [1].



The following general equation is used to express the acceleration due to the aerodynamic interactions of drag on an object.

$$\boldsymbol{a} = \frac{1}{2}\rho_n v^2 \frac{A}{m} C_D \boldsymbol{u_D} \tag{1.1}$$

Here, $\boldsymbol{a}$ is the aerodynamic acceleration component defined by the unit vector, $\boldsymbol{u_D}$, which is in the opposite direction of the relative velocity, ρ_n is the neutral density that is to be modeled, v is the velocity of the object relative to the surrounding atmosphere, $\frac{A}{m}C_D$ represents the inverse of the ballistic coefficient where C_D is the drag coefficient, A is the reference area perpendicular to the direction of the acceleration due to the aerodynamic interactions, and m is the mass of the object. The terms associated with the drag force can be acquired from direct measurements such as mass; however, terms such as neutral density are traditionally obtained using empirical neutral density models. The neutral mass density (kg/m^3) is defined as the mass (kilograms) of chemical molecules with a net charge of zero per cubic area (m^3). The difficulty of modeling drag stems from the complex neutral density variations, which are driven by external energy inputs from charged solar particles interacting with Earth's magnetosphere and internal energy inputs from meteorological phenomena in the lower atmosphere. Further discussion of these complex interactions and their effects on the neutral density variability is addressed in the following section. In this study, first principle (physics-based) modeling of the neutral mass density is used in the application of orbit determination. Unlike empirical models, the physics-based atmospheric models rely on the laws of physics to quantify the variability of the neutral mass density rather than governed exclusively by probability distributions. This approach provides improved spatial resolution of the neutral density and offers the benefit of

temporal evolution during long-term geomagnetic storm events [3]. The aim of this study seeks to validate a new physics-based atmospheric model by understanding and quantifying model inconsistencies. Although this validation effort does not directly model drag, it does address specific model improvements that can later be implemented into orbit propagation models to reduce atmospheric drag uncertainties. With these improved models to better estimate drag, the reduced resources needed to correct and optimize the orbits of modern space technologies including the International Space Station, global positioning systems, and communication satellites can, thereby, reduce costs on taxpayers.

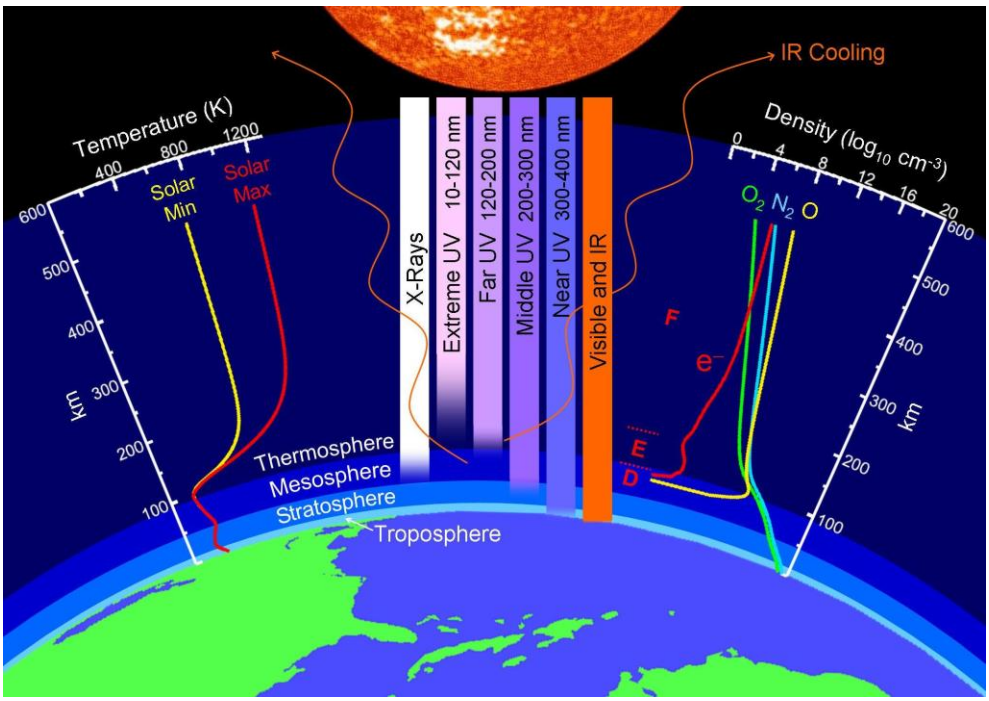
1.2 Physical Characteristics of the Thermospheric Neutral Mass Density

Modeling satellite orbits in collision avoidance and satellite lifetime predictions must consider natural forces such as irregular gravitational field perturbations from the Sun, Earth, and Moon, electrodynamic forces, and atmospheric drag. Lunar gravitational influences become less important when the orbit of a satellite is in LEO. Of these forces, drag poses the most challenging to model as the largest uncertainties are associated with this force due to the variation of the thermospheric neutral density [4]. This thermospheric property is characterized by the non-ionized chemical species whose variation is driven by solar radiative energy, magnetospheric energy, and lower atmospheric propagation from terrestrial weather. The physics of the thermosphere is introduced to better understand neutral density models and what governs their behavior.

The thermosphere is defined as the region of the upper atmosphere above the troposphere, stratosphere, and mesosphere. This layer ranges from 80 km to 600 km in altitude and comprises 99% of neutral chemical species [7]. Figure 1.3 depicts Earth's upper atmosphere including the temperature and neutral composition profiles as a function of altitude.

Figure 1.3

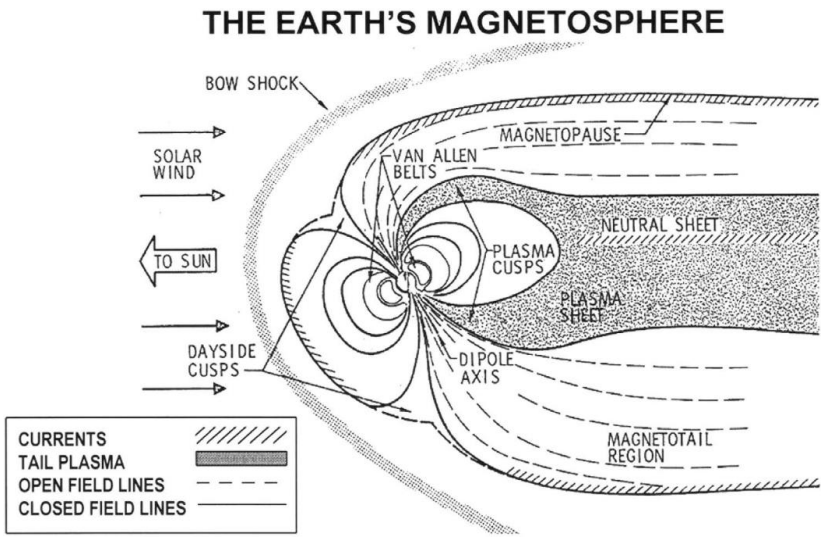
Graphic depiction of Earth's upper atmosphere with temperature, wavelength absorption, and neutral composition profiles (courtesy John Emmert/Naval Research Lab).



Here, the atmosphere is treated as a continuous medium where a majority of the highly energetic solar radiation is absorbed. This results in atmospheric temperatures as high as 2,500 °C (4,530 °F) during quiet geomagnetic activity and even higher temperatures during elevated geomagnetic disturbances. Geomagnetic disturbances are strongly associated with space weather events, which are interactions between the solar wind and the Earth's magnetosphere. If the solar wind has a net interplanetary magnetic field that is oriented southward, then the magnetospheric shield is peeled away. This results in fluctuations in the Earth's magnetic field and the penetration of high-energy particles into the lower atmosphere that can affect space or ground-based technologies and through these, human life. Figure 1.3 shows the dynamic interactions between the solar wind and Earth's magnetosphere.

Figure 1.4

Solar wind interactions on Earth's magnetosphere that develop into space weather [5].



As one increases in altitude, the observer will still experience cold conditions. This is due to the decreased frequency of molecular collisions on the object in the extremely low-density environment by an order of magnitude less than 10^{-11} compared to the density at sea-level. The high temperatures energize the particles to where the enhanced kinetic energy of colliding atomic nuclei is enough to strip off neighboring valence electrons, which results in ionic species that are electrically charged. The upper region of the thermosphere is predominantly composed of ions, which is collectively known as the ionosphere. Although this study focuses on the neutral portion of the thermosphere, ionic species play important roles in dissipating the neutral density during normal diurnal heating and enhanced geomagnetic activity.

Thermospheric neutral gas and thermal dynamics during elevated geomagnetic conditions are a consequence of magnetospheric energy input into the high-latitude regions. This Ohmic production of heat, called Joule heating (scalar product of current and electric field), is due to ionic species drifting through the resistive neutral medium in response to perturbations in the electric field. The weakly colliding ions induce similar convection patterns on the neutral particles. This similar convection pattern of ions dragging neutral particles is known as ion drag. The ion drag expression is

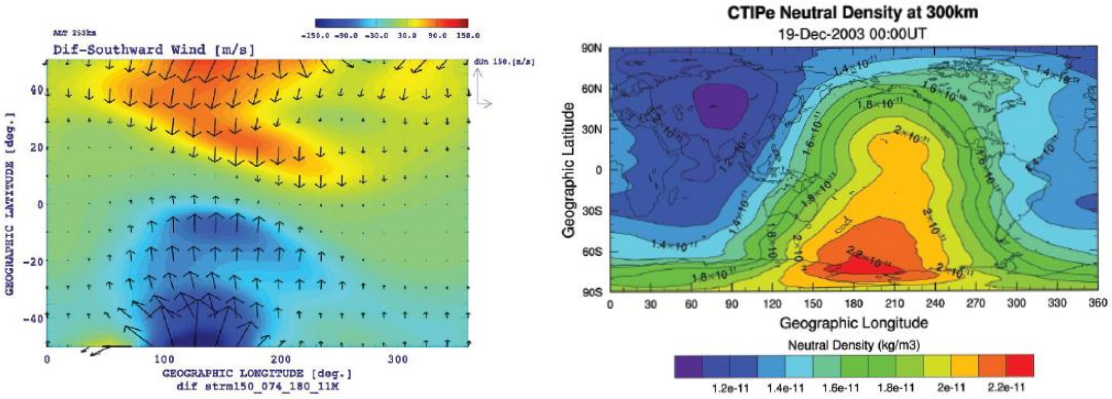
$$-\nu_{ni}(\boldsymbol{V}-\boldsymbol{U}) \tag{1.2}$$

where ν_{ni} is the frequency of neutral-ion collisions, $\boldsymbol{V}$ is the neutral velocity, and $\boldsymbol{U}$ is the ion velocity [6]. A negative sign is introduced because the ion drag acts opposite to the direction of the particle motion. If the neutral and ionic species move in the same convective motion, then Joule heating equals zero. However, other external forces, including viscosity, inertia, and the Coriolis effect, restrict the neutral and ion species to

exactly match the convective motion. As a result, a temperature gradient is formed where the high-latitude regions undergo thermospheric expansion of the neutral density more quickly than the equatorial regions. This gradient governs the global neutral density propagation in the form of nonlinear meridional and zonal neutral winds that flow horizontally from high-latitude regions to equatorial regions. Figure 1.5 shows this neutral wind propagation to the equatorial regions and global neutral density distribution. During enhanced geomagnetic storms, these winds significantly upwell or transport heavier neutral species to higher altitudes, which ultimately increases the mean neutral density at a fixed height. During the 20-21 November 2003 geomagnetic storm, Bruinsma, S. et. al. [7] found that neutral density increased between 300-800% when measured from Challenging Minisatellite Payload (CHAMP) and Gravity Recovery and Climate Experiment (GRACE) satellites at an altitude of 400-500 km.

Figure 1.5

Images of neutral wind at mid-latitudes at 250 km altitude after an increase of magnetospheric energy input (left) and model representation of global neutral density distribution contours at pressure level at 300 km (right) [6].



However, the Earth does have a cooling mechanism that recovers the temperature and transports the higher neutral densities back to lower altitudes via thermospheric contraction. While Joule heating and solar radiation increase thermospheric temperature, a thermodynamic energy sink called infrared (IR) cooling is responsible for absorbing and releasing energy from excited nitric oxide (NO) gas into space. This cooling reduces global temperatures after intense geomagnetic events via radiative emission at a wavelength of 5.3 μm at an altitude of 150-200 km [6]. This reduction of temperature also reduces the thermospheric expansion. The following momentum and energy equations in spherical polar coordinates, respectively, are meant to highlight the predominant components to model neutral density dynamics [6].

$$\frac{\partial}{\partial t}V_\theta = -\frac{V_\theta}{r}\frac{\partial}{\partial \theta}V_\theta - \frac{V_\phi}{rsin\theta}\frac{\partial}{\partial \phi}V_\theta - \omega\frac{\partial}{\partial p}V_\theta - \frac{g}{r}\frac{\partial}{\partial \theta}h + \left(2\Omega + \frac{V_\phi}{rsin\theta}\right)V_\phi cos\theta$$
$$+ g\frac{\partial}{\partial p}\left[(\mu_m + \mu_T)\frac{p}{H}\frac{\partial}{\partial p}V_\theta\right] - v_{ni}(V_\theta - U_\theta) \quad (1.3)$$

and

$$\frac{\partial}{\partial t}V_\phi = -\frac{V_\theta}{r}\frac{\partial}{\partial t}V_\phi - \frac{V_\phi}{rsin\theta}\frac{\partial}{\partial \phi}V_\phi - \omega\frac{\partial}{\partial p}V_\phi - \frac{g}{rsin\theta}\frac{\partial}{\partial \varphi}h - \left(2\Omega + \frac{V_\phi}{rsin\theta}\right)V_\theta cos\theta$$
$$+ g\frac{\partial}{\partial p}\left[(\mu_m + \mu_T)\frac{p}{H}\frac{\partial}{\partial p}V_\phi\right] - v_{ni}(V_\phi - U_\phi) \quad (1.4)$$

The energy equation can be expressed as

$$\frac{\partial}{\partial t}\varepsilon = -\frac{V_\theta}{r}\frac{\partial}{\partial \theta}(\varepsilon + gh) - \frac{V_\phi}{rsin\theta}\frac{\partial}{\partial \phi}(\varepsilon + gh) - \omega\frac{\partial}{\partial p}(gh) + Q_{euv} + Q_{ir} + Q_{vis}$$
$$+ g\frac{\partial}{\partial p}\left[(\kappa_m + \kappa_T)\frac{p}{H}\frac{\partial}{\partial p}T\right] - g\frac{\partial}{\partial p}\frac{g\kappa_T}{c_p} - \frac{J_\theta E_\theta + J_\phi E_\phi}{\rho} \quad (1.5)$$

where V_θ and V_ϕ are the meridional and zonal neutral winds, U_θ and U_ϕ are the meridional and zonal ion winds, r is the radius from point of origin to the gas parcel, μ_m and μ_T are viscous drag terms, ω is the vertical wind in the pressure coordinates, Ω is the planet's angular velocity of the Coriolis term, g is the gravitational acceleration, p is the pressure surface, and h is the height for the momentum equations. For the energy equation, ε is the sum of the specific enthalpy, Q_{euv} and Q_{vis} represent the energy sources extreme ultraviolet (EUV) and viscous heating, Q_{ir} represents an energy sink infrared (IR) cooling, κ_m and κ_T represent the vertical heat conduction terms, $\frac{g}{c_p}$ is the adiabatic lapse rate, J_θ and J_ϕ are the meridional and zonal currents, E_θ and E_ϕ are the meridional and zonal electric fields as part of the Joule heating term, and H is the scale height.

The terms of these equations such as horizontal and vertical advection, pressure, vertical winds and heat conduction, Coriolis, and viscosity are not discussed because their implications are out of the scope of this study. However, the components do offer an insight into modeling the complex dynamics of neutral gases in the thermosphere. This study does focus on external energy inputs of Joule heating, ion drag, and other energy sources and sinks when explaining the behavior of the studied model.

Not only do external energy drivers from galactic cosmic rays and the solar wind affect the thermosphere but so do internal factors from the troposphere that drive upper atmospheric characteristics. Energy from atmospheric gravity waves and atmospheric tides propagate upward into the upper atmosphere. Gravity waves occur when fluid boundaries are perturbed, and the atmosphere attempts to restore equilibrium in density. These disturbances manifest from wind flowing over mountains or from violent thunderstorms or

hurricanes [8]. Gravity waves with lower amplitudes can avoid wave-breaking in the lower atmosphere. Wave-breaking is analogous to waves breaking in oceanic turbulence because of instabilities associated with enhanced wave amplitudes. Once these waves progress into the mesopause, the amplitudes increase exponentially as a function of altitude. This is due to the logarithmic decrease of density as the waves propagate upward resulting in periodic advection and compression of plasma. Those that propagate further into the thermosphere and can cause traveling ionospheric disturbances, which are associated with affecting technologies that use high frequency (HF) radio waves for communication services.

CHAPTER II

PHYSICS-BASED MODELING AND OBSERVATIONS OF THE THEROSPHERIC NEUTRAL MASS DENSITY

2.1 Efforts in Modeling the Neutral Density

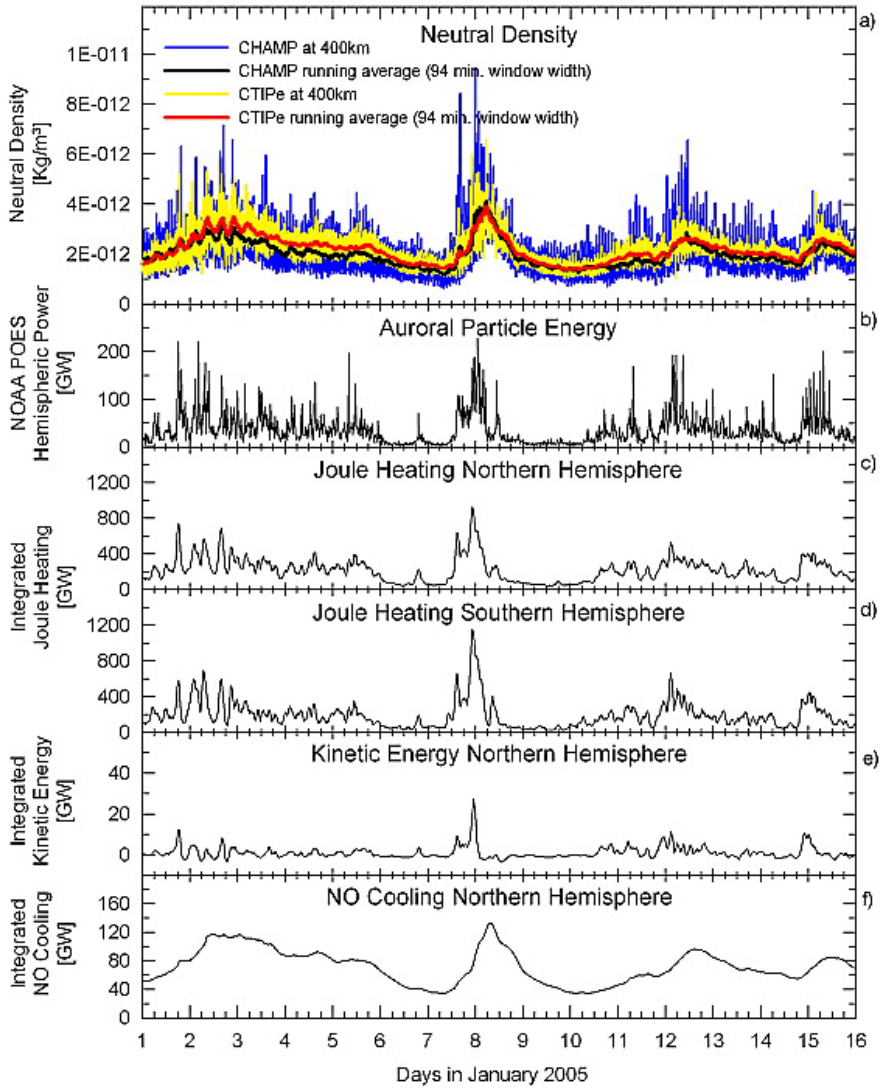
The combination of external and internal energy drivers introduces many complexities in modeling thermospheric physical processes. Initial neutral density models that are used for orbit prediction are empirically driven such as the MSIS models [9] and Jacchia-Bowman models [10]. Empirical data are obtained from historical observations from geomagnetic conditions at a given time and location. These data include Kp, solar radiative energy inputs, and the Disturbance Storm Time (Dst) index to drive neutral density variability within the model. However, describing the neutral density variation in terms of indices is insufficient as the thermosphere is predominantly driven by external sources of energy through time [11]. Empirical models cannot reproduce accurate magnitude and time-evolution of upper atmospheric variations [12]. Physics-based models offer a solution to temporal-dependent evolution of neutral density during quiet-time and geomagnetic storm events [13].

One of the latest physics-based models is the Coupled Thermosphere-Ionosphere-Plasmasphere electrodynamics (CTIPe) model. This model is well-validated regarding its neutral density component [13]. Therefore, understanding its physical processes is important in this validation study. CTIPe is currently in operations at the Space Weather

Prediction Center (SWPC) to develop nowcasting and forecasting of space weather events. This model runs about 30 minutes ahead of real-time to forecast total electron content that directly affects the accurate positioning of the Global Navigation Satellite System satellites. Three components are coupled together to form CTIPe. The first is a neutral thermosphere code developed by Fuller-Rowell, T. J. and Rees, D. [14]. This thermosphere code was then coupled to the second component that modeled the mid- and high-latitude ionosphere, which collectively became the Coupled Thermosphere-Ionosphere Model (CTIM) [15]. Improvements to the CTIM model were incorporated by coupling a low-latitude ionosphere, plasmasphere, and electrodynamics code, resulting in CTIPe [16, 17]. The resolution of CTIPe is 2º and 18º in geographic latitude and longitude, respectively, and into 15 logarithmic pressure levels in the vertical direction from 80 km to over 500 km [18]. This model uses the TIROS/NOAA auroral particle precipitation measurements and the Weimer electric field model for Joule heating and solar wind parameters [19]. Input parameters that drive the model include the solar wind characteristics such as density, velocity and the interplanetary magnetic field (IMF) [13]. This study utilizes CTIPe for verifying the physics of the next-generation physics-based model, Whole Atmosphere Model (WAM) coupled Ionosphere-Plasmasphere-Electrodynamics (IPE) model (WAM-IPE), during the 2013 St. Patrick's Day geomagnetic storm from 16 March through 19 March. Figure 2.1 provides a comparison of neutral density measurements from the CHAMP satellite and CTIPe simulation along with the primary energy contributors in neutral density variability.

Figure 2.1

Neutral density comparisons of CTIPe simulation results and CHAMP measurements at 400 km along with energy partitioning during the January 2005 geomagnetic storm (a). (b) is the estimated auroral particle energy contribution, (c) and (d) are Joule heating contribution in the Northern and Southern hemispheres, (e) is the kinetic energy deposition, and (f) is NO cooling rates at 5.3 μm infrared emission [13].

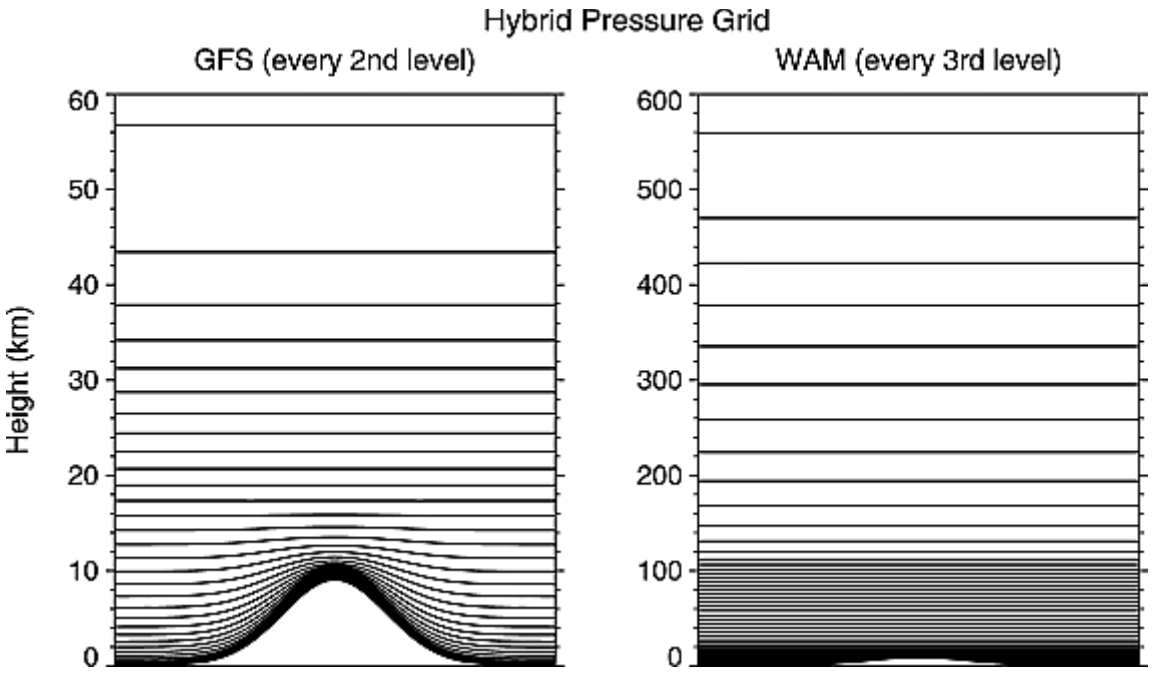


2.2 Physics-Based Model: WAM-IPE

The WAM-IPE model is a new comprehensive, time-dependent, three-dimensional spectral model of the Earth's ionosphere, thermosphere, and lower atmosphere. This model is in test-operational mode at the National Oceanic Atmospheric Administration (NOAA) SWPC to work in tandem with CTIPe on nowcasting and forecasting of space weather events [20]. The modeled thermosphere is solved on a discrete grid defined on a non-inertial frame, fixed with respect to Earth's rotation. The model has a resolution of 2º and 1.875º in geographic latitude and longitude, respectively, and into 150 logarithmic pressure levels in the vertical direction from the ground to over 600 km. This model captures lower and upper atmospheric forcings including gravity waves and magnetospheric electric field perturbations, which both contribute to the neutral density variability. The WAM portion is an extended model of the Global Forecast System (GFS) that is currently in operations at the National Weather Service. Figure 2.2 illustrates the grid extension applied to GFS. This lower atmospheric weather model includes cloud physics and radiation that contribute to the vertical propagation of meteorological perturbations on the ionosphere. The physics from WAM are currently in a one-way coupling configuration to IPE. In-depth details regarding the IPE portion of the WAM-IPE model are not discussed as these details are beyond the scope of this study. However, IPE can reproduce total electron content observations, which are associated with storm enhanced densities that impact satellite navigation and communications [22]. Neutral density simulations from WAM-IPE are verified with CTIPe simulations and validated with accelerometer-derived measurements from the Gravity field and steady-state Ocean Circulation Explorer (GOCE).

Figure 2.2

Illustration of an idealized 9 km mountain in the GFS 64-layer grid along with the WAM 150-layer grid [21].



2.3 GOCE Satellite Neutral Density Measurements

The GOCE satellite was designed to map the Earth's gravitational field at LEO. GOCE was launched by the European Space Agency in March 2009 into a near Sun-synchronous orbit (96.7º inclination) at an altitude of 255 km. Its mission lifespan was expected to last only 21 months. However, solar wind pressure estimates were lower than predicted, which resulted in reduced fuel consumption. Therefore, its mission was extended to four years. With its additional time in LEO, GOCE was able to provide valuable gravitational measurements and thrust data that are used to estimate the local neutral density for a variety of geomagnetic events. This near-Sun-synchronous dawn-dusk orbit crossed the equator at 18:00 and 06:00 local solar time [23]. GOCE's orbital altitude gradually decreased until the end of its mission in November 2013 after depleting its propellant due to continuous orbit corrections in the denser region. Neutral density observations were derived from a highly sensitive triaxial accelerometer on-board GOCE [23]. Figure 2.4 illustrates the algorithm used in neutral density determination. The accelerometer was initially used to monitor the ion thruster's performance of GOCE rather than for scientific purposes. As a result, the temporal resolution is dependent on the sampling rate (0.1 Hz) of the thruster activation data. Therefore, the spatial resolution of the time-series is approximately 80 km. Neutral density measurements are not entirely absolute because the measurements are derived from accelerometer values rather than from dedicated instrumentation designed for direct neutral density measurements. Since the neutral density values are obtained through a geometry-based approach, a scalar modification applied to the modeled acceleration vectors to match the observed accelerometer values. This results in reference neutral density values that are dependent on

the scalar modification applied and a correction factor may be needed to correct the bias in GOCE. Another source of error is a lack of understanding the momentum transfer of the thermospheric particle bombardments on GOCE's exterior structure throughout its oblate orbit. The drag coefficient in equation 1.1 is therefore inadequate in modeling the density. The algorithm shown in figure 2.4 is based on the geometry of GOCE's orbit in figure 2.3 and the environmental conditions at the time of the accelerometer measurements using a modeled along-track acceleration vector that matches the accelerometer's vectoral measurements. The modeled density is then modified to match the length of the modeled and observed vectors. In this study, the magnitude of the provided neutral density values during the 2013 March 16-19 time-series are multiplied by a correction factor of 1.23 to account for bias errors associated with the algorithm [23].

Figure 2.3

Schematic of matching the magnitudes of the modeled acceleration ($\boldsymbol{a}_{\text{mod}}$) with the observation acceleration ($\boldsymbol{a}_{\text{obs}}$) by adjusting the density ($\boldsymbol{\rho}$) and relative velocity vector ($\boldsymbol{v_r}$) [23].

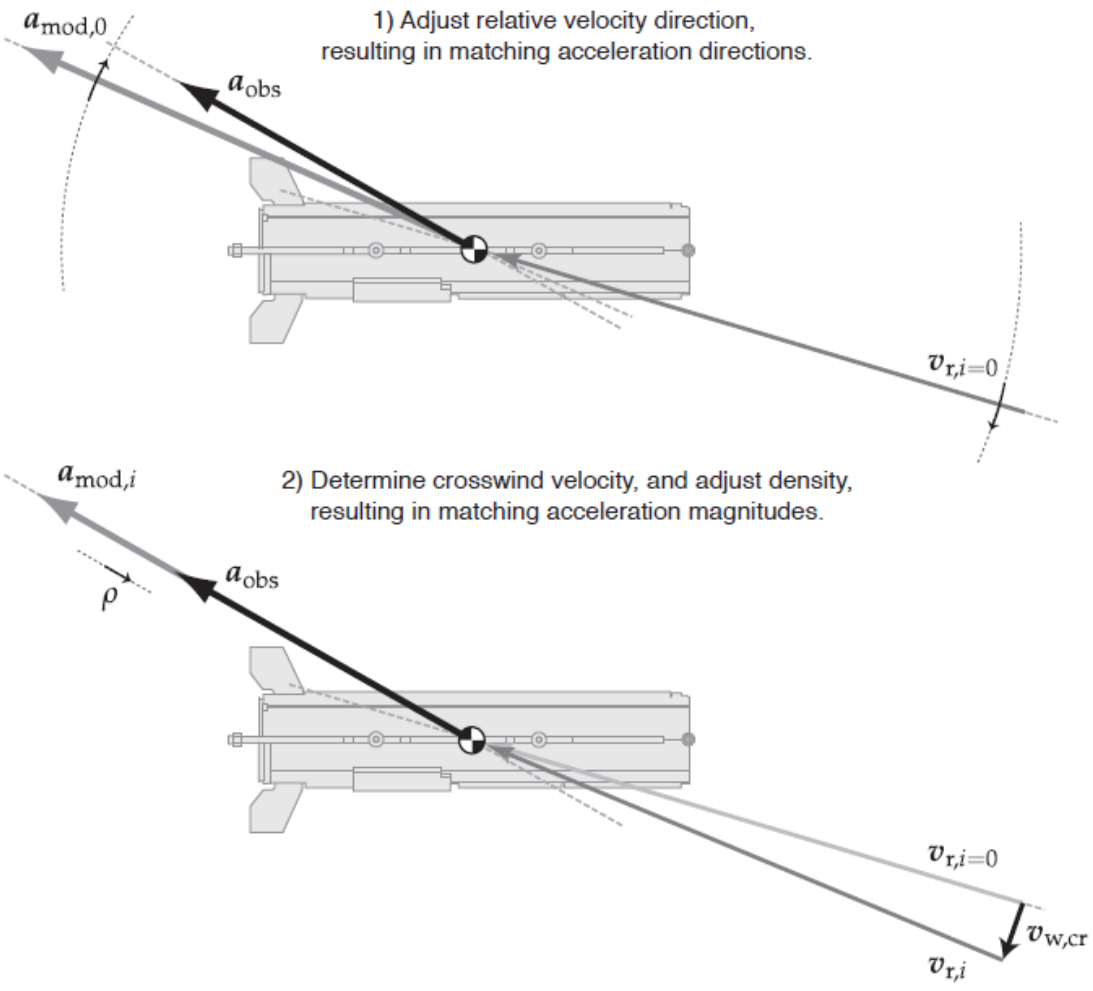
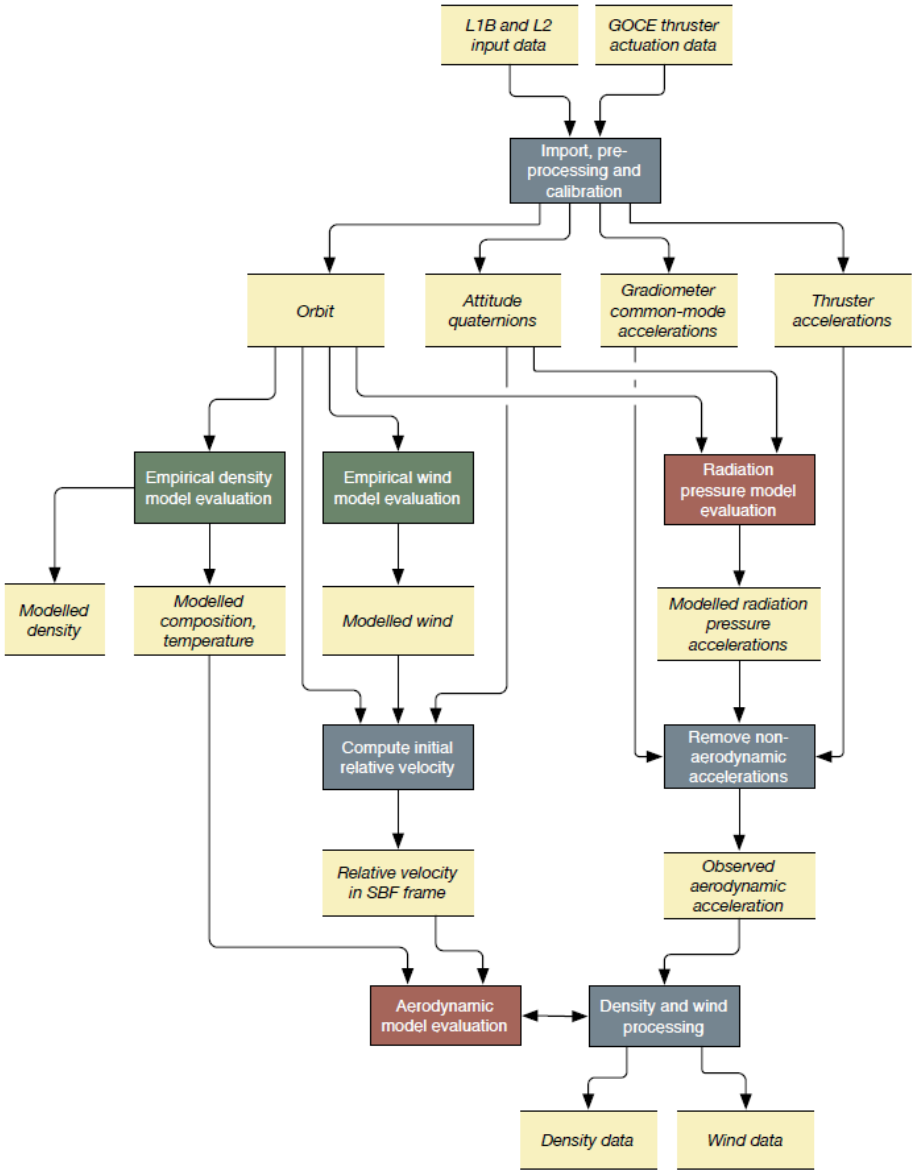


Figure 2.4

Flow chart illustrating the neutral density and wind determination [23].



CHAPTER III

VALIDATION OF WAM-IPE NEUTRAL MASS DENSITY

3.1 Methodology

WAM-IPE and CTIPe simulations are run on NOAA supercomputers and are compared with GOCE satellite observations during 2013 March 16-19. This time period was chosen because a geomagnetic storm called the 2013 St. Patrick's Storm impacted Earth at the same time as GOCE was in orbit. With these neutral density measurements from GOCE, WAM-IPE simulation capabilities can be analyzed and compared with observations during a geomagnetic disturbance. To plot the comparisons, GOCE satellite location and time are used to define specific interpolated model outputs. GOCE neutral density data are taken from a .dat file and imported into a .xlsx file for data organization. Table 3.1 is the organized GOCE data displaying its location and time at the beginning of March 16. The time dimension is converted from the day/hour/minute/second format into the decimal form based on the hour for proper treatment of defining indices while accounting for randomly missed observations. Each geographic latitude, longitude, altitude, and time columns are inserted into .txt files for reading into the Interactive Data Language (IDL) software. IDL is used for programming for its default statistical and smoothing functions. The statistics for quantifying model validation and smoothing functions for enhanced data visualization are discussed in the later section. The program outputs were written to .txt files for further postprocessing and plotting in Microsoft Excel.

Table 3.1

GOCE Spatial and Temporal Resolution Corresponding with Neutral Density

Day	Hour	Minute	Second	Time [hr]	Lat.	Lon.	Alt. [m]	ρ_n [kg/m^3]
16	00	00	00	0.000000	-58.918	300.676	264850.830	3.61387e-11
16	00	00	10	0.002778	-58.267	300.355	264638.459	3.65595e-11
16	00	00	20	0.005556	-57.614	300.043	264422.335	3.69878e-11
…	…	…	…	…	…	…	…	…

Since both CTIPe and WAM-IPE simulation solutions are generated at discrete points in a grid at each timestep, GOCE's geographic location at a defined time may not lie directly at a known solution. Therefore, intermediate points within a grid cube must be interpolated at GOCE's location. Because the neutral density varies logarithmically between height levels, logarithmic interpolation is applied between the pressure level outputs to solve for the corresponding scale heights. Once the heights are determined, the density is then interpolated. Zonal and meridional neutral density variations can be assumed to vary linearly because the high WAM-IPE spatial resolution between any two adjacent latitude and longitude points corresponds to minimal density fluctuations at the same height level. Bilinear interpolation is then applied to match GOCE's latitude and longitude location at a defined time and altitude. The following equations are used for logarithmic and bilinear interpolation

$$h(z) = z_2^f z_1^{1-f} \tag{3.1}$$

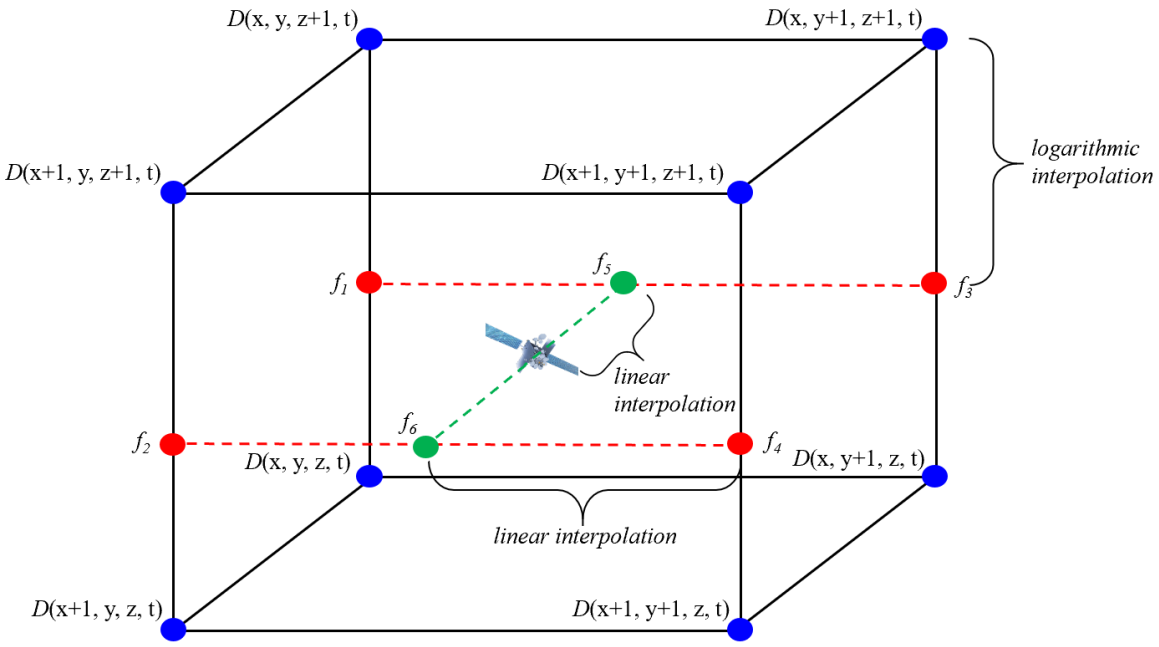
$$f(x) = (1-f)x_1 + fx_2 \tag{3.2}$$

$$f = \frac{a}{a+b} \tag{3.3}$$

where *h* is the logarithmic interpolated scale height corresponding to GOCE's altitude, *z* is the model pressure levels, *x* is the model latitude or longitude grid points, *f* is the fractional division of the point of interest between consecutive (a and b) pressure levels, latitudes, or longitudes. Density is then determined by using the interpolated height. Figure 3.1 illustrates an idealized grid pattern of the eight WAM-IPE or CTIPe grid solutions surrounding the along-orbit location of GOCE. These points are used to resolve the 3-dimensional interpolated neutral density value at GOCE's location within the model.

Figure 3.1

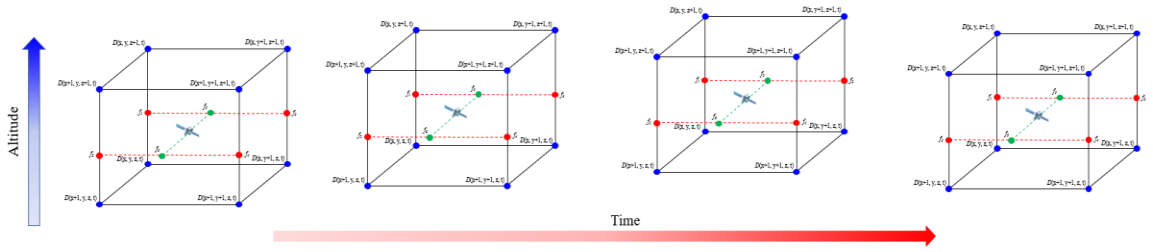
3D depiction of logarithmic and bilinear interpolation at GOCE geographic location and time.



Since GOCE along-orbit track is changing with time, the code must be able to account for the spatial distance traveled and atmospheric temporal evolution between each 10-second interval. Therefore, the IDL code must perform an active "self-check" VALUE_LOCATE function for the nearest latitude, longitude, height, and time indices that correspond to GOCE's position at each interval. Figure 3.2 is a 4-dimensional representation of the code's ability to produce the time-dependent neutral density values throughout GOCE's orbit.

Figure 3.2

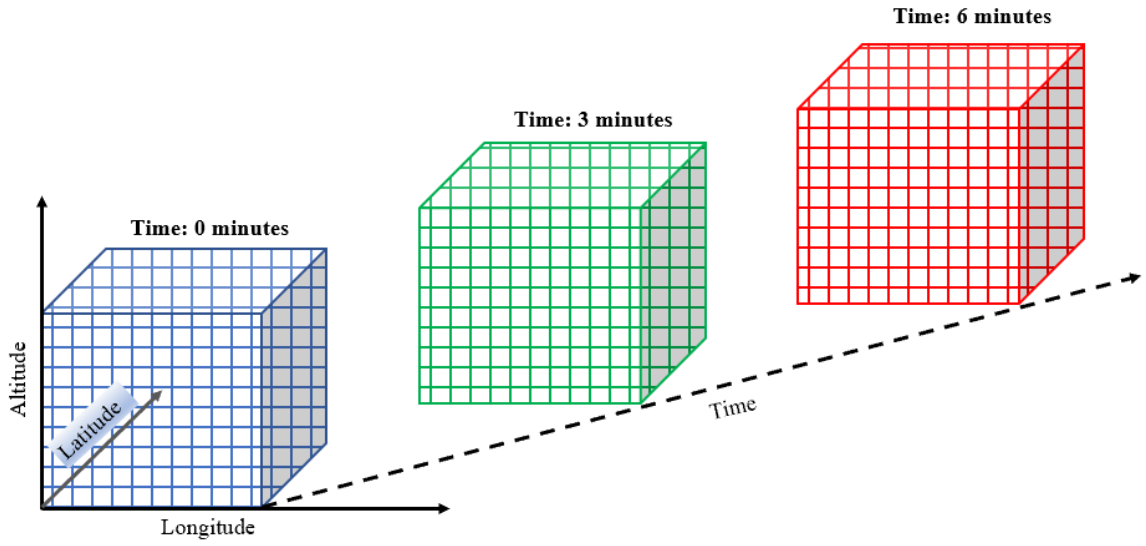
4D depiction of density determination in WAM-IPE and CTIPe using the location and time from GOCE data.



WAM-IPE and CTIPe simulations are output to netCDF file formats, which are “self-describing” files where a header defines the multidimensional array associated with attributes or variables defined by the user. These file formats are commonplace in developing array-oriented scientific data in geosciences such as atmospheric modeling. Figure 3.3 illustrates the layout of a hypothetical netCDF neutral density variable at 3-minute time steps with each resolved grid space corresponding to the following timestep. For example, an hour-long WAM-IPE netCDF file would have 20 time steps for the temporal resolution and 150 pressure levels, 94 latitudes, and 192 longitudes for the spatial resolution. This means that the total neutral density variable solutions per hour long file are 20 x 150 x 94 x 192 = 54,144,000 solutions and 2,707,200 solutions per timestep. In addition to the neutral density variable, the file also contains other output variables for mean molecular mass and temperature.

Figure 3.3

Schematic of model output netCDF multidimensional neutral density array example with 3-minute timesteps.



3.2 Model/Data Comparisons

In this section, four days of neutral density simulations from WAM-IPE are compared with CTIPe and GOCE satellite observations during the 2013 St. Patrick's Day Storm. As mentioned in section 2.1, the four-day window was chosen to assess the WAM-IPE model capability during the coronal mass ejection (CME) on 17 March. These comparisons also identify areas that require improvements during CME response and recovery. The model and data comparisons are performed through a time series of along-orbit and orbit-averaged values. The orbit-averaged values assist in data visualization by shape filtering density peaks and troughs using a boxcar average provided by the IDL SMOOTH function. The box car average is expressed as

$$R_i = \begin{cases} \frac{1}{w}\sum_{j=0}^{w-1} A_{i+j-\frac{w}{2}}, & if\ \frac{(w-1)}{2} \le i \le N - \frac{(w+1)}{2} \\ A_i, & otherwise \end{cases} \tag{3.4}$$

where *w* is the smoothing width or the number of elements in one orbit, *N* is the number of elements in the array *A*. The along-orbit and orbit-averaged time series statistical analysis is conducted by the evaluation of correlation coefficients (R), biases, standard deviations (SD), and root-mean square errors (RMSE). The correlation coefficient is the measure of the degree of the linear relationship between the model and observations; bias measures the consistent model offset with respect to a given reference value; standard deviation is the measure of dispersion in the data from its mean value; and the root-mean square error is a quadratic scoring function, which measures the average magnitude of the error [9]. These statistics are taken in logarithmic space rather than linear space. The log space is chosen because a normal distribution is skewed in linear space because the neutral density

varies based on a logarithmic best-fit relationship between two scale heights. The log space accounts for this density variation, and the model to observation ratio interval scale is more consistent with a standard normal distribution function. The statistical metrics can be expressed as the following equations

$$\mu(m/o) = exp\left(\frac{1}{N}\sum_{n=1}^{N} ln\frac{\rho_{m,i}}{\rho_{o,i}}\right) \tag{3.5}$$

$$\sigma(m/o) = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(ln\frac{\rho_{m,i}}{\rho_{o,i}} - \ln\mu(m/o)\right)^2} \tag{3.6}$$

$$RMSe^2 = ln\big(\mu(m/o)\big)^2 + \sigma(m/o)^2 \tag{3.7}$$

where $\mu(m/o)$ is the mean bias of model-to-observation, N is the total number of data points (34,546), $\rho_{m,i}$ and $\rho_{o,i}$ are the model and observation neutral density values at a specified index, respectively, and $\sigma(m/o)$ is the standard deviation of model-to-observation. In this study, the ratios of the modeled and observed neutral densities are used for statistical analysis using a similar technique conducted by Mariangel et. al [9].

The predominant effects of the CME that impacted Earth during the St. Patrick's Day storm are increased magnitudes of solar wind pressure on the magnetosphere, magnetic fields, particle precipitation, and Dst ring current as shown in figure 3.4. The quantified disturbances to the Earth's horizontal magnetic field component are indicated by the K-index (Kp), which is derived from high-latitude ground-based magnetometers. This storm had a peak Kp of 6 based on the NOAA/SWPC nowcasting services. The following figure 3.4 is publicly available 1-min-averaged field and plasma data sets from ACE, Wind, IMP 8, and Geotail satellites during the 2013 St. Patrick's Day storm.

Figure 3.4

Magnetic field, plasma, energetic particle data (OMNI) provided by NASA Goddard's Space Physics Data Facility for 16-19 March 2013. Shown are Kp, interplanetary magnetic field (IMF) in B_y and B_z orientations, solar wind velocity (SWV), auroral electrojet (AE) index, and ring current (Dst). These values serve as inputs into WAM-IPE and CTIPe simulations.

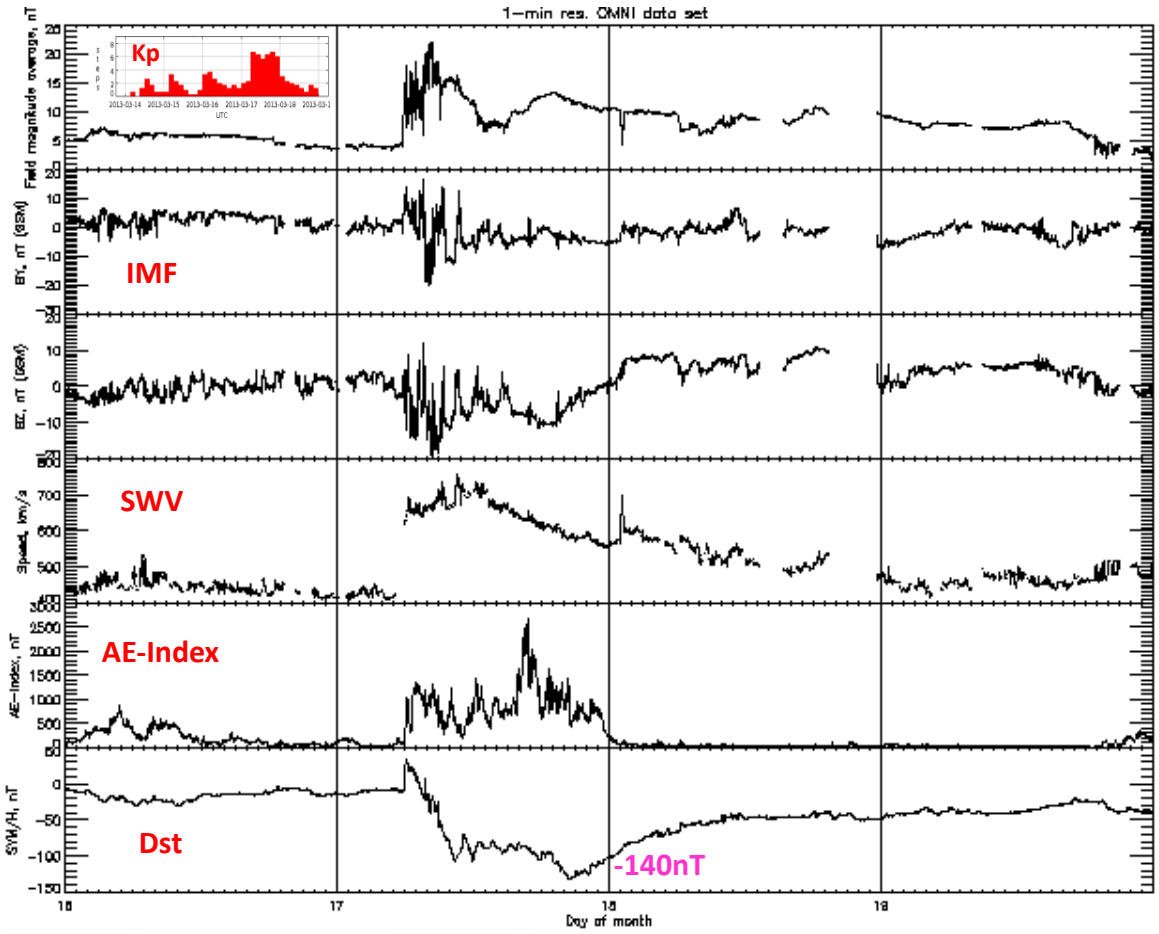


Figure 3.5 shows the first comparisons between the 4th order numerical scheme of WAM-IPE model neutral density simulations and along-orbit/orbit-averaged GOCE observations. The agreement between the model and observations are illustrated by the scatterplot and linear regression where a reference (i.e. y = x black dashed line) line is provided. It should be noted that a bias of 1 describes no mean offset between the model and the observation. The thin blue, black, and red lines represent the along-orbit WAM-IPE simulation results, GOCE observations, and CTIPe simulation results, respectively. The thick blue, black, and red lines represent the respective orbit-averaged density values. Results from the statistical analysis show R = 0.917, RMSE = 0.328, bias = 1.36, and SD = 0.120 for the along-orbit GOCE and WAM-IPE comparisons and R = 0.949, RMSE = 0.466, bias = 1.57, and SD 0.111 for the along-orbit CTIPe and WAM-IPE comparisons. WAM-IPE is able to follow GOCE observations during quiet-time, but its density values underestimate storm-time response and cooling recovery when compared to GOCE and CTIPe. This comparison is better visualized in figure 3.6 when the orbit-averaged density values of the WAM-IPE model are normalized to GOCE on March 16. WAM-IPE reasonably follows the increase in neutral density during the onset of the storm on day 17 and the cooling response when compared to CTIPe and GOCE. However, the rate of heating during storm-time and the rate of cooling post-storm-time is not as pronounced as CTIPe and GOCE. This may be due to inconsistencies within the model's Joule heating and neutral composition parameters. During the storm-time period, Joule heating is the predominant driver of increased temperatures in the upper latitudes. Due to this temperature gradient, the heavier neutral molecular species are expected to be transported higher in altitude in the lower latitude regions.

Figure 3.5

Time series (a) of the original WAM-IPE neutral density simulations compared with CTIPe and GOCE along with linear regression and correlations (b) (c). GOCE's geographical location is used as inputs for WAM-IPE and CTIPe. WAM-IPE output demonstrates statistical bias when compared to GOCE and CTIPe and underestimates storm-time and cooling recovery responses.

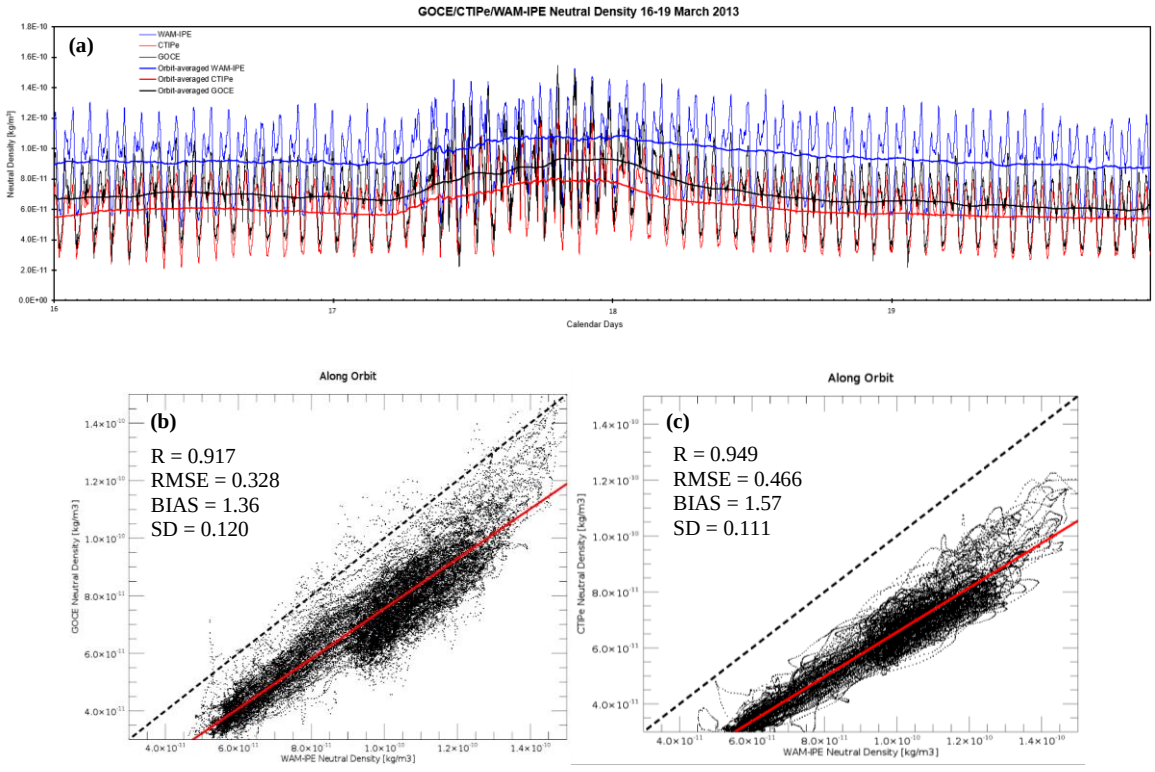
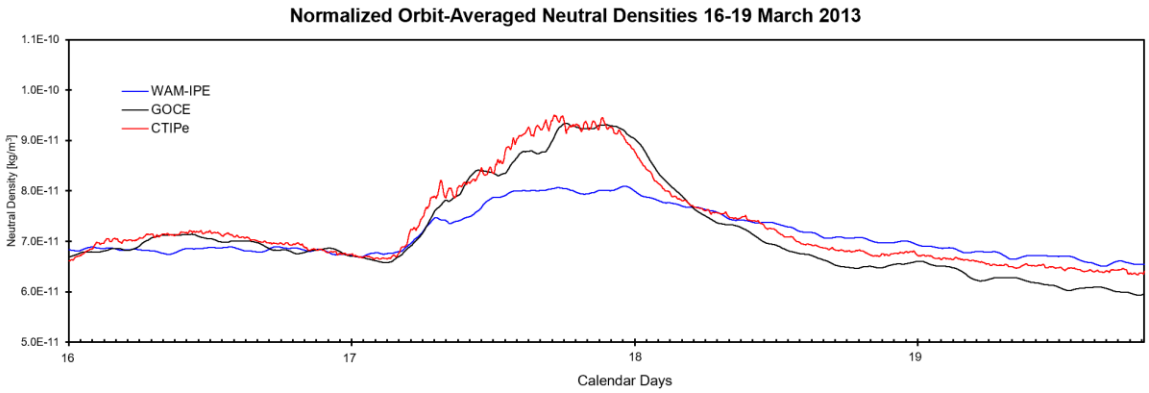


Figure 3.6

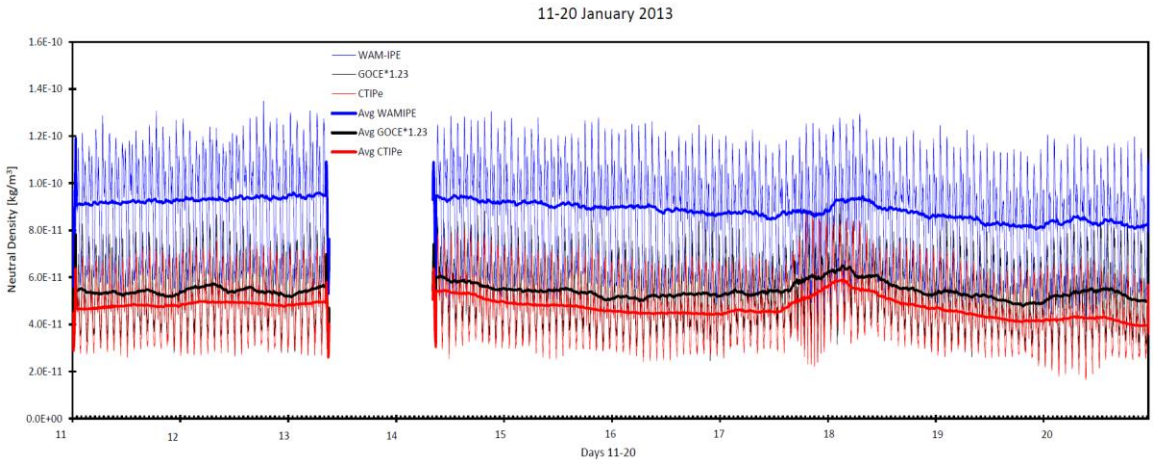
Time series of the orbit-averaged WAM-IPE, CTIPe, and GOCE neutral densities normalized to GOCE on March 16 to better capture WAM-IPE underestimation of storm-time and cooling recovery responses.



Another noticeable trend is that WAM-IPE displays consistently higher global neutral density values when compared to GOCE. Various mechanisms are responsible for the consistently high average neutral density values. The global parameter that is responsible for the model's mean energy is the solar heating factor (SHF). An improved SHF may then change the mean global neutral density with respect to its reference scale height. Prior to modifying the SHF, another study was conducted to determine if there was a seasonal dependence of the Joule heating and SHF. Figure 3.7 is a time series plot of 11-20 January 2013 to assess WAM-IPE's response during quiet to moderate geomagnetic activity.

Figure 3.7

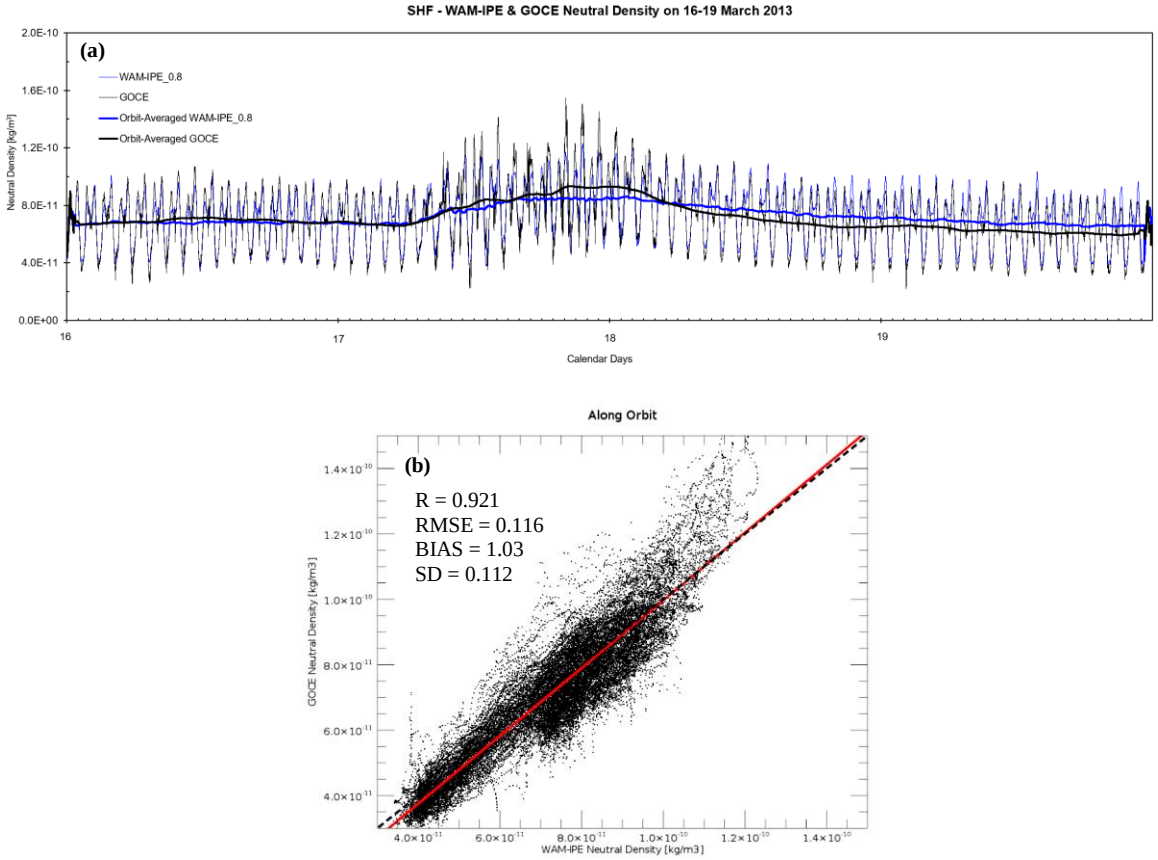
Time series of the original WAM-IPE neutral density simulations compared with CTIPe and GOCE during 11-20 January 2013 exhibits consistently higher mean global densities.



As shown in figure 3.7, the neutral density of WAM-IPE is consistently higher than GOCE and CTIPe during a different season; therefore, this supplied enough evidence to adjust the SHF. The SHF was adjusted to 0.8 to determine if this variable is responsible for the combined offset and underestimation of storm-time response and recovery. This coefficient was obtained from referencing U.S. Standard Atmosphere pressure level ratios at the respective altitudes. Based on how the solar factor is implemented in WAM-IPE, this would correct the pressure level at which the model is reading. A lower SHF would correct WAM-IPE to read at a higher apparent altitude, which is expected to lower the global neutral density output. Figure 3.8 is the time series plot with the adjusted SHF. Results from the statistical analysis show R = 0.921, RMSE = 0.116, bias = 1.03, and SD = 0.112, which is a significant improvement from the previous statistics by decreasing the bias associated with the global heating parameter. Prior to the geomagnetic storm, WAM-IPE agrees well with GOCE data with a reduction in the RMSE and bias by 40.1% and 14.5%, respectively, when using the correct SHF instead of the original factor. However, WAM-IPE continues to underestimate the storm-time and cooling recovery responses. This suggests that Joule heating and the neutral composition model parameters at this altitude may be responsible for these discrepancies.

Figure 3.8

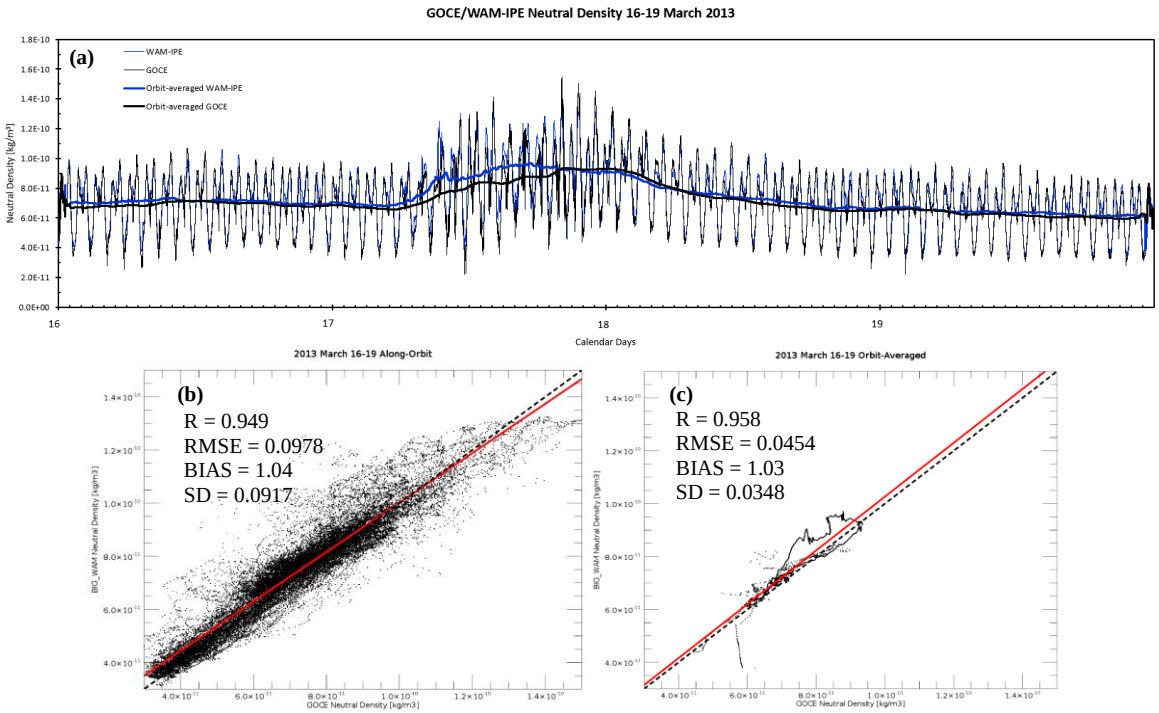
Time series of the SHF corrected WAM-IPE neutral density simulations compared with GOCE during 16-19 March 2013 with reduced bias. Continues to underestimate storm-time response and cooling recovery.



Further investigation in the dynamical core of the WAM-IPE model revealed that a numerical inconsistency in the advection terms of the governing equations within the WAM portion may be responsible for the inconsistent mean offset, response, and recovery. Recent GFS adjustments to the advection terms within the operational model were made by transitioning from an Eulerian transport scheme to a semi-Lagrangian scheme, which improves model efficiency particularly at high resolutions [24]. Figure 3.9 shows the resultant time series plot and statistics that incorporated this modification into WAM.

Figure 3.9

Time series of improved WAM dynamical core regarding the switch from Eulerian transport to semi-Lagrangian transport. WAM-IPE simulations demonstrates *reduced statistical bias*.



This new version of WAM-IPE also included an 8th order numerical scheme for improved precision and model stability. Results from the statistical analysis show R = 0.949, RMSE = 0.0.978, bias = 1.035, and SD = 0.0917, demonstrating the best results thus far. Although the new model densities overestimate during the onset of storm-time observed by GOCE, they do follow satellite measurements remarkably well during quiet-time prior to the onset of the geomagnetic storm and cooling recovery. The global structure during select maxima on March 16 is shown in figure 3.10, and the WAM-IPE/GOCE agreement is better captured visually for daily neutral density variations in figure 3.11.

Figure 3.10

Global structure WAM-IPE neutral density maxima at defined GOCE locations on 2013 March 16.

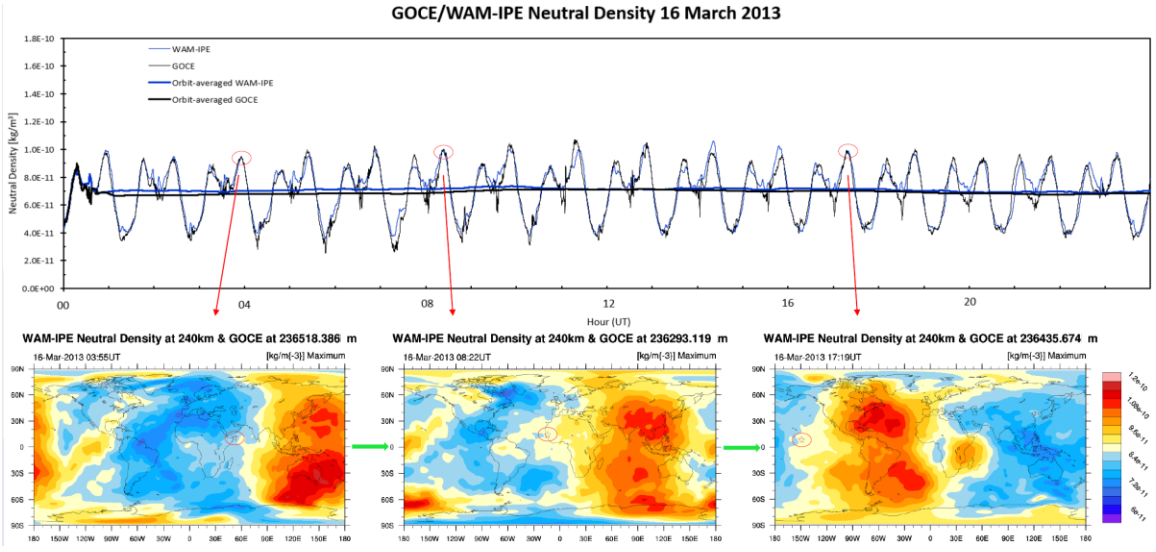
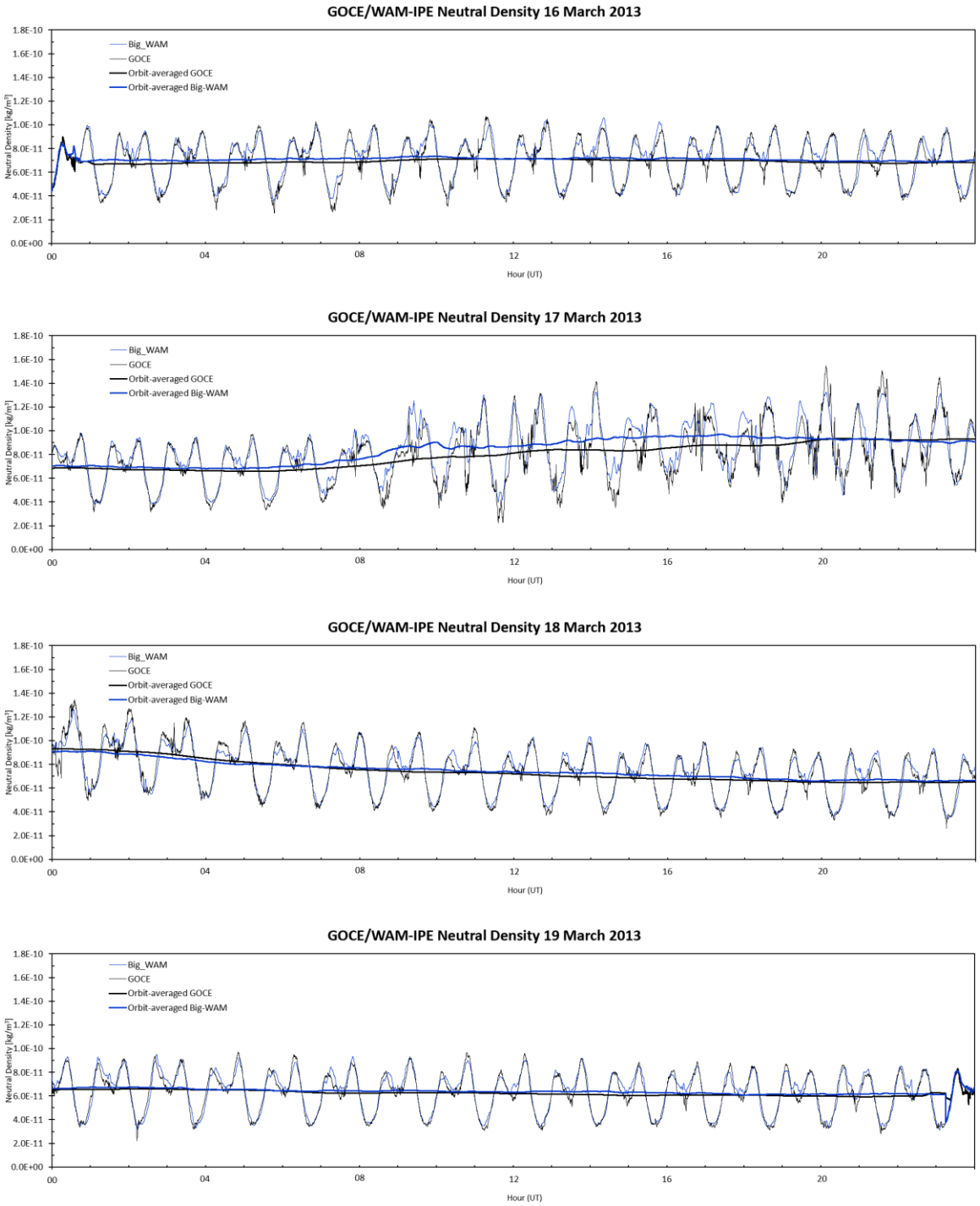


Figure 3.11

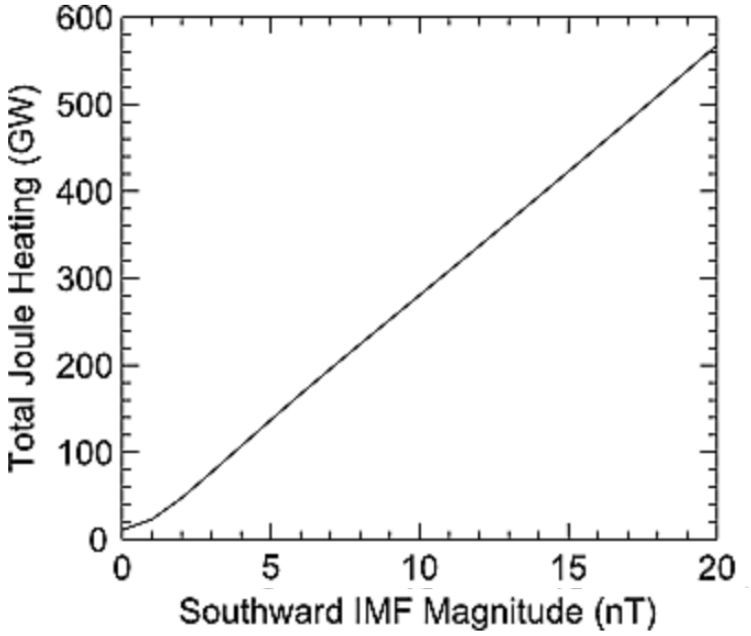
Individual-day time series of improved WAM dynamical core for graphical visualization.



The agreement between the satellite and WAM-IPE is dependent on the accuracy of the spatial distribution and the magnitude of the Joule heating as a function of altitude. This overestimation response during the onset of the geomagnetic storm may be due to the Weimer statistically linear increase in Joule heating energy that drives the WAM-IPE model during geomagnetic disturbances [25]. Figure 3.12 shows the linear relationship of the total Joule heating as a function of the southward IMF Magnitude. This linear relationship may need to introduce saturation as the IMF increases in magnitude throughout the storm. This saturation modification promises a potential reduction of the overestimation exhibited during the onset of the storm in figure 3.9. This is still an open question of whether the Weimer model is exclusively driving this particular physical process. Further studies are required to understand the Joule heating relationship for this increase in neutral density variation. The neutral composition parameter is also another mechanism that may improve model accuracy. WAM-IPE and CTIPe do not account for dominant molecular species in the upper altitudes. In this region of the atmosphere, the molecular oxygen component dramatically reduces, and helium becomes the predominant molecule. Neither of these models includes helium transport processes during geomagnetic disturbances. Again, incorporating helium transport may improve model responses to geophysical processes, and further studies are required to better understand upper atmospheric characteristics.

Figure 3.12

Total Joule heating (GW) as a function of southward IMF magnitude (nT) at 450 km/s SWV from the Weimer model that drives the Joule heating component of WAM-IPE [25].



CHAPTER IV

CONCLUSION

4.1 Summary

The neutral density during quiet and geomagnetically disturbed times is an important and complex variable to model when predicting satellite and debris orbits. Many thermodynamic and electrodynamic factors contribute to the energy transfer processes in the magnetosphere-ionosphere-thermosphere system. During geomagnetic storms, Joule heating energy contributions can be greater than the combined solar extreme ultraviolet and ultraviolet radiation absorption. The energy is then radiated out into space in the form of IR from the NO cooling effect, which returns the neutral density and temperature to homeostatic levels. The work presents a validation effort on the neutral density relationship between the WAM-IPE model and GOCE satellite observations. When the model results and observation are in good agreement, the model can be used to estimate neutral density variations. The results of the study suggest that WAM-IPE can capture the storm-time response and recovery remarkably well when compared to GOCE with a reduction of statistical bias from the initial study to the improved dynamical core by 88.9%. With appropriate inputs and parameters implemented, small variations in the neutral density can be captured with high precision when compared to previous physics-based models [22]. There is more structure in the WAM-IPE model compared to the CTIPe model, but this is expected given that the WAM portion incorporates meteorological processes to the

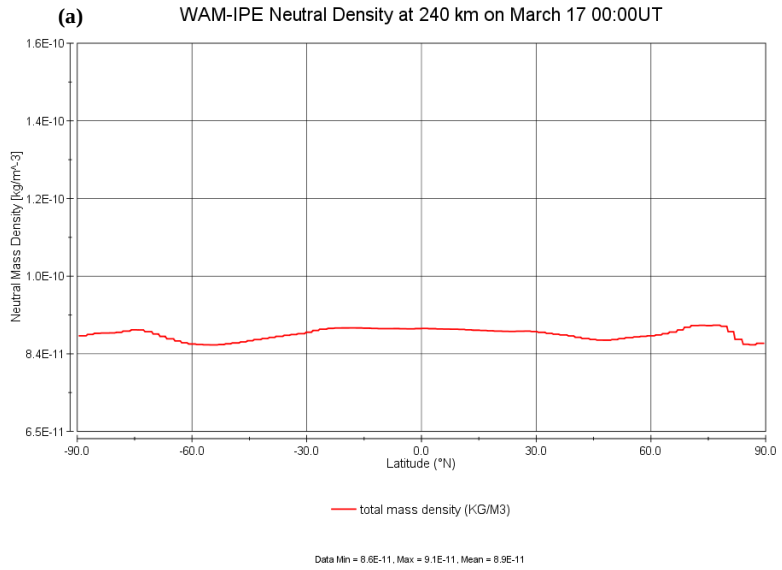
thermospheric and ionospheric system. This study also demonstrated that small modifications in the model dynamical core can cause large changes in the simulation, confirming that continuous improvements to parameterizations are needed to resolve sub-grid processes and characteristics for obtaining meaningful results. Currently, WAM-IPE runs 20-30 minutes ahead of real-time by taking SWV and IMF measurements from the Advanced Composition Explorer spacecraft. With more refined magnetospheric and solar wind models, WAM-IPE can forecast thermospheric processes including neutral density variations from a few hours to potentially days ahead of real-time.

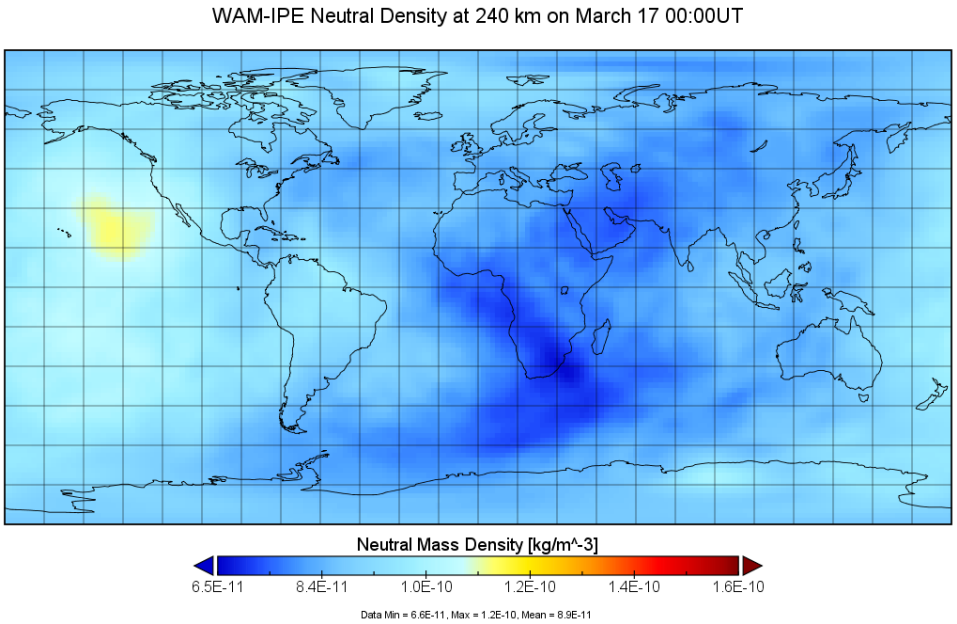
4.2 Future Work

The immediate validation efforts seek to improve neutral density responses during geomagnetic storm-times to further understand and quantify model inconsistencies when compared to observations. Validation is needed for seasonal dependence of heating and cooling of WAM-IPE by analyzing global structure rather than the local characteristics with respect to a satellite's along-orbit track. Figure 4.1 shows the zonal neutral density averages with the corresponding equirectangular plots toward global validation efforts. These plots are the latest global density structure before, during, and after the St. Patrick's Day storm, respectively. Efforts are continually being made for increased resolution and inclusion of geophysical processes to better model responses to geomagnetic activity throughout the whole atmosphere. With gradual improvements to the WAM-IPE model, orbit propagation models may be able to utilize WAM-IPE's innate atmospheric modeling functions. This coupling will better predict satellite orbits in collision avoidance and lifetime optimization schemes.

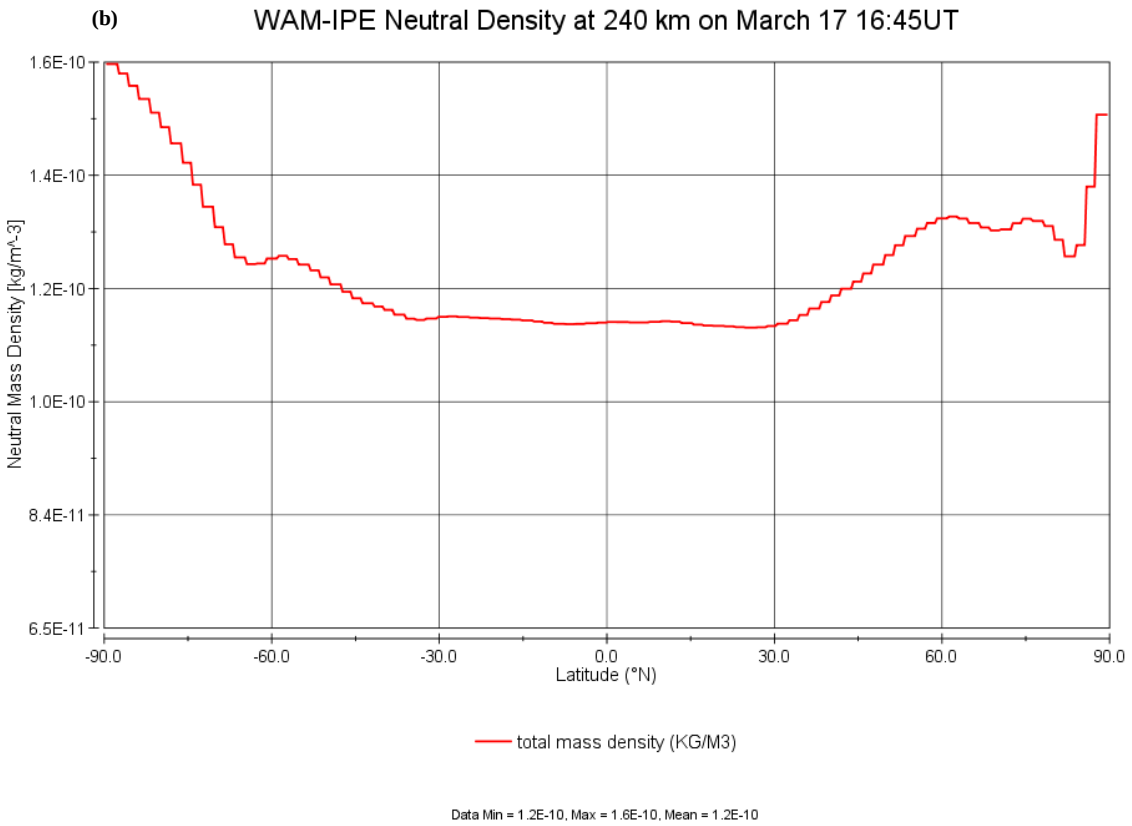
Figure 4.1

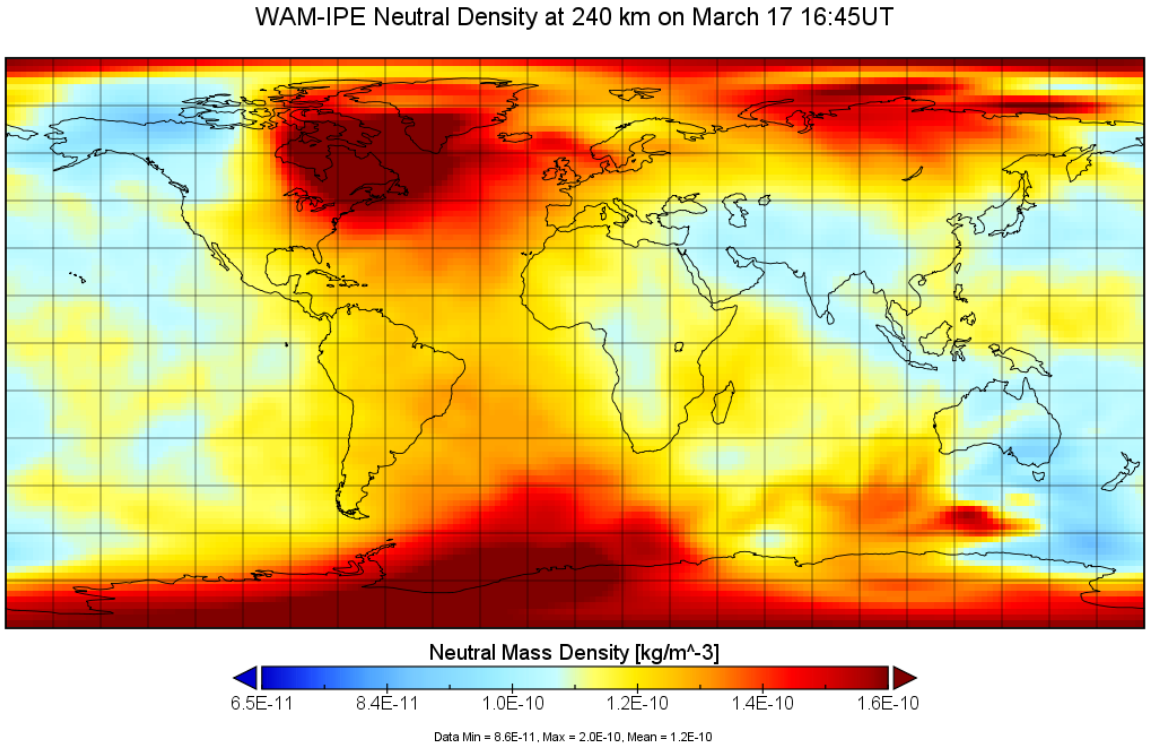
Zonal averages and global plots of the neutral density. (a) shows the neutral density during quiet geomagnetic activity. (b) shows the thermospheric expansion during the peak of the St. Patrick's Day storm. (c) shows the thermospheric cooling due to 5.3 μm IR emission.



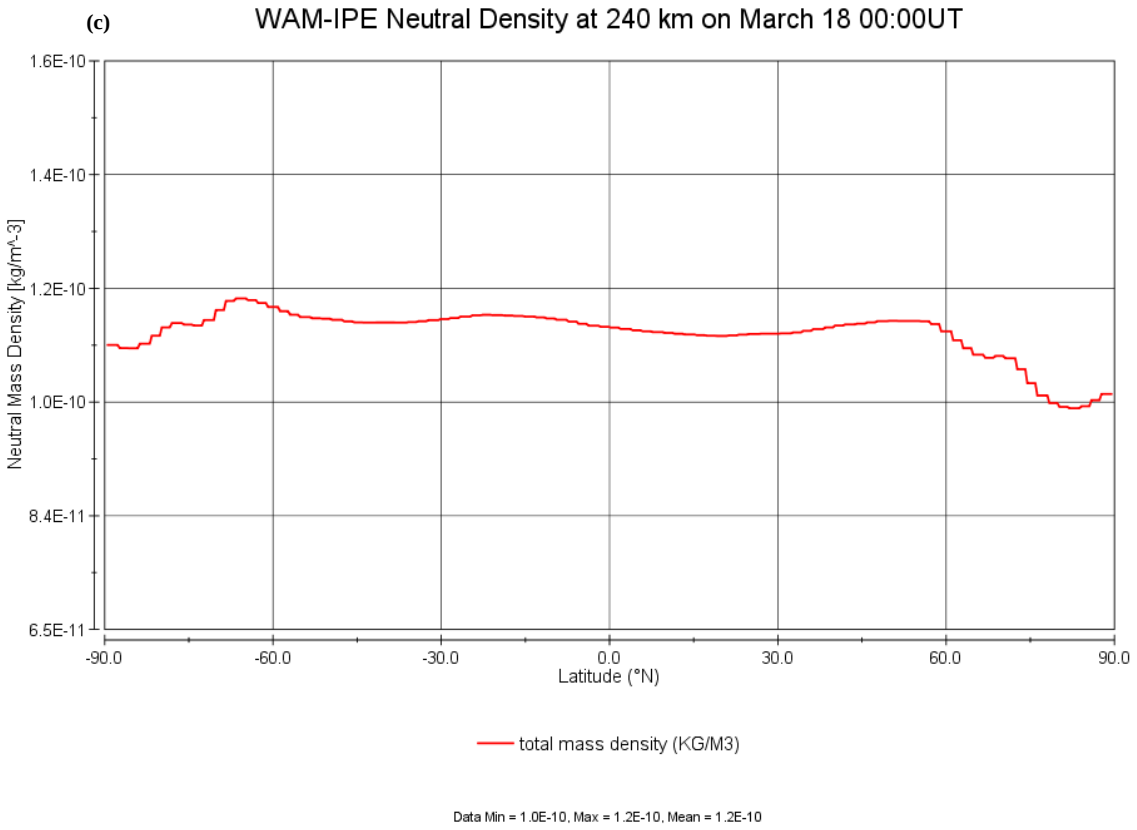


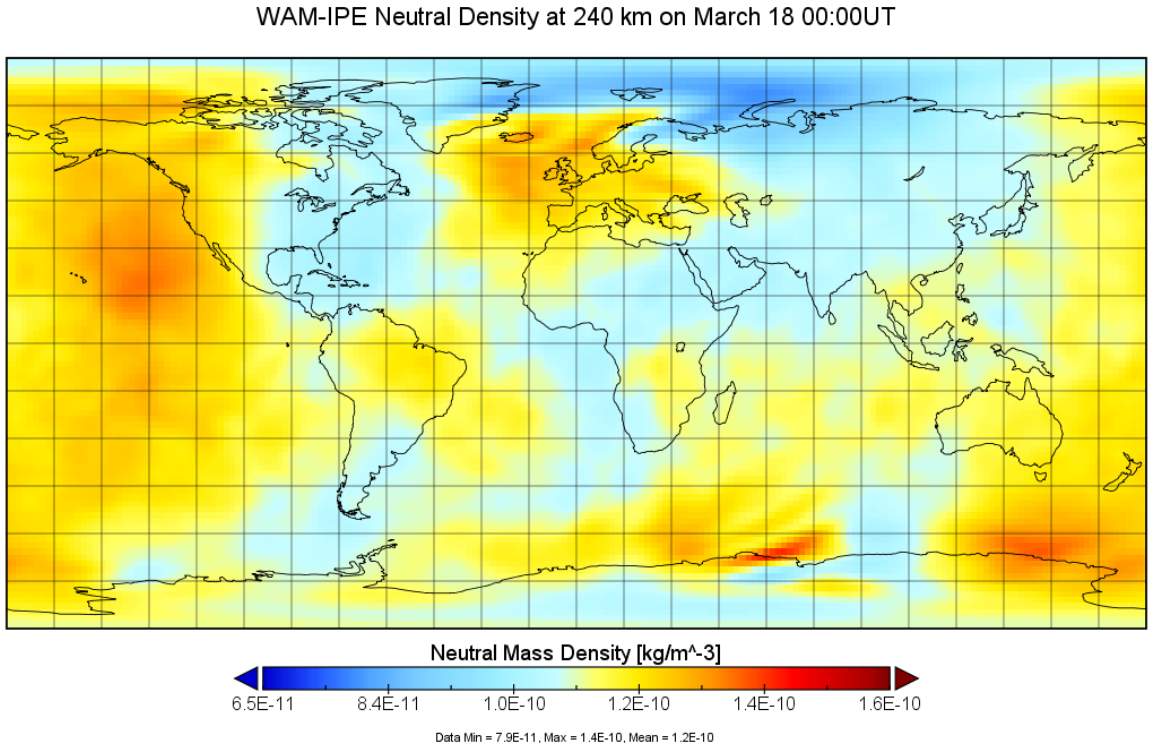
(b)


WAM-IPE Neutral Density at 240 km on March 17 16:45UT
1.6E-10
1.4E-10
1.2E-10
1.0E-10
8.4E-11
6.5E-11
Neutral Mass Density [kg/m^-3]
-90.0
-60.0
-30.0
0.0
30.0
60.0
90.0
Latitude (°N)
total mass density (KG/M3)
Data Min = 1.2E-10, Max = 1.6E-10, Mean = 1.2E-10


WAM-IPE Neutral Density at 240 km on March 17 16:45UT
Neutral Mass Density [kg/m^-3]
6.5E-11
8.4E-11
1.0E-10
1.2E-10
1.4E-10
1.6E-10
Data Min = 8.6E-11, Max = 2.0E-10, Mean = 1.2E-10

(c)


WAM-IPE Neutral Density at 240 km on March 18 00:00UT
1.6E-10
1.4E-10
1.2E-10
1.0E-10
8.4E-11
6.5E-11
Neutral Mass Density [kg/m^-3]
-90.0
-60.0
-30.0
0.0
30.0
60.0
90.0
Latitude (°N)
total mass density (KG/M3)
Data Min = 1.0E-10, Max = 1.2E-10, Mean = 1.2E-10


WAM-IPE Neutral Density at 240 km on March 18 00:00UT
Neutral Mass Density [kg/m^-3]
6.5E-11
8.4E-11
1.0E-10
1.2E-10
1.4E-10
1.6E-10
Data Min = 7.9E-11, Max = 1.4E-10, Mean = 1.2E-10

REFERENCES

[1] Koskinen, H., Tankanen, E., Pirjola, R., Pulkkinen, A., Dyer, C., Rodger, D., Cannon, P., Mandeville, J.-C., and Boscher, D. (2001). "Space weather effects catalogue: ESA space weather study (ESWS)". Finnish Meteorological Institute (2): 1-41

[2] National Research Council (NRC). (1995). "Orbital Debris: A Technical Assessment". National Academy of Science. Washington, D.C.

[3] Fuller-Rowell, T. J., and Solomon, S. C. (2010). "Flares, coronal mass ejections, and atmospheric responses", *Heliophysics: Space Storms and Radiation: Causes and Effects*. edited by C. J. Schrijver and G. L. Siscoe, 321-57

[4] Doornbos, E. (2007). "Thermosphere density model calibration". In: Lilensten J. (eds) Space Weather: Research Towards Application in 2nd European Space Weather Week (ESWW2). *Astrophysics and Space Science Library* 344: 107-14

[5] Olson, W. P., and Moe, K. (1974). "Influence of precipitating charged particles on the high-latitude thermosphere". *Journal of Atmospheric and Terrestrial Physics* 36: 1715-16

[6] Fuller-Rowell, T. (2013). "Physical characteristics and modeling of earth's thermosphere". *Geophysical Monograph Series* 201:13-27

[7] Bruinsma, S., Forbes, J. M., Nerem, R. S., and Zhang, X. (2006), "Thermosphere density response to the 20-21 November 2003 solar and geomagnetic storm from CHAMP and GRACE accelerometer data". *Journal of Geophysical Research: Space Physics* 111(A6): 1-14

[8] Vadas, S., and Liu, H. (2009). "Generation of large-scale gravity waves and neutral winds in the thermosphere from the dissipation of convectively generated gravity waves". *Journal of Geophysical Research: Space Physics* 114(A10): 1-25

[9] Picone, J. M., Hedin, A. E., Drob, D.P., and Aikin, A. C. (2002). "NRLMSISE-00 empirical model of the atmosphere: Statistical comparisons and scientific issues". *Journal Geophysical Research: Space Physics* 107(A12): 1-16

[10] Bowman, B. R., Tobiska, W. K., Marcos, F. A., Huang, C. Y., Lin, C. S., and Burke, W. J. (2008). "A new empirical thermospheric density model JB2008 using new solar and geomagnetic indices". *AIAA/AAS Astrodynamic Specialist Conference, American Institute of Aeronautics and Astronautics* 1-19

[11] Fuller-Rowell, T., Minter, C., Codrescu, M., and Fedrizzi, M. (2009). "Estimating neutral atmosphere drivers using a physical model". Report. *Air Force Office of Scientific Research* 1-9

[12] Moe, K., and Moe, M. M. (2011). "Operational models and drag-derived density trends in the thermosphere". *Space Weather: The International Journal of Research and Applications* 9(5): 1-6

[13] Fedrizzi, M., T. Fuller-Rowell, M. Codrescu (2012). "Global Joule heating index derived from thermospheric density physics-based modeling and observations". *Space Weather* 10(3): 1-13

[14] Fuller-Rowell, T. J., and Rees, D. (1980). "A three-dimensional, time-dependent, global model of the thermosphere". *Journal of Atmospheric Sciences* 37: 2545-67.

[15] Fuller-Rowell, T. J., Rees, D., Quegan, S., Moffett, R. J., Codrescu, M. V., and Millward, G. H. (1996). "A coupled thermosphere-ionosphere model (CTIM)". *Handbook of Ionospheric Models* edited by R. W. Schunk. 217-38

[16] Millward, G. H., Moffett, R. J., Quegan, S., and Fuller-Rowell, T. J. (1996). "A coupled thermosphere-ionosphere-plasmasphere model (CTIP)". In: Schunk, R. W. (eds), *Solar-Terrestrial Energy Program: Handbook of Ionospheric Models*, Center for Atmospheric and Space Sciences, Utah State University, Logan, Utah 239-279

[17] Richmond, A. D., and Roble, R. G. (1987). "Electrodynamic effects of thermospheric winds from NCAR thermospheric general circulation model". *Journal of Geophysical Research: Space Physics* 92(A11): 12365-76

[18] Fuller-Rowell, T. J., Millward, G. H., Richmond, A. D., and Codrescu, M. V. (2002). "Storm-time changes in the upper atmosphere at low latitudes". *Journal of Atmospheric and Solar-Terrestrial Physics* 64(12-14): 1383-91

[19] Foster, J. C., Holt, J.M., Musgrove, R. G., and Evans, D. S. (1986). "Ionospheric convection associated with discrete levels of particle precipitation". Geophysical Research Letters 13(7): 656-9

[20] Negrea C., Codrescu, M., Fuller-Rowell, T. J. (2012). "On the Validation Effort of the Coupled Thermosphere Ionosphere Plasmasphere Electrodynamics Model". *Space Weather* 10(8): 1-9

[21] Akmaev, R. A. (2011). “Whole atmosphere modeling: Connecting terrestrial and space weather”. *Reviews of Geophysics* 49(4): 1-30

[22] Maruyama N., et al. (2017). “Ionospheric Weather from the Coupled Whole Atmosphere Model (WAM)-Ionosphere-Plasmasphere-Electrodynamics (IPE) Model toward NOAA Operational Space Weather Forecasting”. *Major Scientific Challenges in Space Weather* 4C-2

[23] Doornbos, E., Bruinsma, S., Fritsche, B., Visser, P., Van Den IJssel, J., Teixeira Encarnaçõa, J., and Kern, M. (2013). “Air density and wind retrieval using GOCE data”. *ESA Living Planet Symposium 2013* ESA SP-722: 1-8

[24] Sela, J. (2010). “The derivation of the sigma pressure hybrid coordinate semi-Lagrangian model equations for the GFS”. *NOAA/NCEP Office Note* 462: 1-31

[25] Weimer, D. R. (2005). “Improved ionospheric electrodynamic models and application to calculating Joule heating rate”. *Journal of Geophysical Research: Space Physics* 110(A5): 1-21

APPENDIX A

IDL CODES FOR VALIDATION AND STATISTICAL ANALYSIS

yearprog.pro

```
; Martin McCandless Code
; NOAA SWPC 7-5-2018
; *NOTE ONLY A TEST*

PRO YearProg, channel

rhoWAM1 = 'C:\Users\Martin McCandless\Desktop\NOAA\Mar16-19\density_output.txt'
OPENW,1,rhoWAM1

obstime = DBLARR(5,34546)
OPENR,lun,'C:\Users\Martin McCandless\Desktop\NOAA\Mar16-19\16-19time.txt',/get_lun
READF,lun,obstime
FREE_LUN, lun
close,/all
help,obstime
print,'obstime',obstime(21)

obsheight = DBLARR(1,34546)
OPENR,lun,'C:\Users\Martin McCandless\Desktop\NOAA\Mar16-19\height.txt',/get_lun
READF,lun,obsheight
FREE_LUN, lun
close,/all
help,obsheight
print, 'obsheight', obsheight(0)

obslong = DBLARR(1,34546)
OPENR,lun,'C:\Users\Martin McCandless\Desktop\NOAA\Mar16-19\longitude.txt',/get_lun
READF,lun,obslong
FREE_LUN,lun
close,/all
help,obslong
print, 'obslong',obslong(0)

obslat = DBLARR(1,34546)
OPENR,lun,'C:\Users\Martin McCandless\Desktop\NOAA\Mar16-19\latitude.txt',/get_lun
READF,lun,obslat
FREE_LUN, lun
close,/all
help,obslat
print, 'obslat', obslat(0)

FOR j = 700,800 DO BEGIN ;8635
```

```
43 tic
44
45 ; Open WAM file
46 i = obstime(1,j)
47 k = obstime(0,j)
48 filename = 'C:\Users\Martin McCandless\Desktop\WAM_Output\Mar' +
   STRING(k,FORMAT='(I2.2)') + '\2013Mar' + STRING(k, FORMAT='(I2.2)')
   + 'h' + STRING(i, FORMAT='(I2.2)') + '.nc'
49 file = NCDF_OPEN(filename)
50
51 ; Defining and Calling Variables in WAM
52 ; Latitude(a), Longitude(b), Pressure Level(c)
53 a = NCDF_VARID(file, 'lat')
54 b = NCDF_VARID(file, 'lon')
55 c = NCDF_VARID(file, 'lev')
56 d = NCDF_VARID(file, 'den')
57 e = NCDF_VARID(file, 'z')
58 f = NCDF_VARID(file, 'time')
59 g = NCDF_VARID(file, 't')
60
61 ; Assigning names to a, b, c, d, e, f
62 NCDF_VARGET, file, a, latitude
63 NCDF_VARGET, file, b, longitude
64 NCDF_VARGET, file, c, level
65 NCDF_VARGET, file, d, density
66 NCDF_VARGET, file, e, height
67 NCDF_VARGET, file, f, time
68 NCDF_VARGET, file, g, temp
69
70 ; Pring Dimenstions of each variable
71 print, 'DimensionsLat = ',size(latitude)
72 print, 'DimensionsLon = ',size(longitude)
73 print, 'DimensionsLev = ',size(level)
74 print, 'DimensionsDen = ',size(density)
75 print, 'DimensionsHt  = ',size(height)
76 print, 'DimensionsTime= ',size(time)
77 print, 'DimensionsTemp= ',size(temp)
78
79 ; Print Initial Latitude, Longitude, and Pressure Level Values
80 print, 'initialLat = ',latitude(0)
81 print, 'initialLon = ',longitude(0)
82 print, 'initialLev = ',level(0)
83 print, 'initialDen = ',density(0,0,0,0)
84 print, 'initialHt = ',height(0,0,0,0)
85 print, 'initialTime = ',time(0)
86 ;-----------------------------------------------------------------
   ----------------------
```

```
; Quick Search Routine for Latitude
numberLat = obslat(j)
print, 'observed Latitude', numberLat
binLat = VALUE_LOCATE(latitude, numberLat)
print, 'Lat & Lat+1', latitude[binLat], latitude[binLat+1]

; Check with two values that bracket the chosen number.
CASE 1 OF
  binLat EQ -1: closestLat = latitude[0]
  binLat EQ (N_ELEMENTS(latitude)-1): closestLat =
  latitude[N_ELEMENTS(latitude)-1]
  ELSE: IF ABS(latitude[binLat] - numberLat) GT
  ABS(latitude[binLat+1] - numberLat) THEN $
    closestLat = binLat+1 ELSE closestLat = binLat
ENDCASE
CASE 2 OF
  binLat EQ -1: closestLatvalue = latitude[0]
  binLat EQ (N_ELEMENTS(latitude)-1): closestLatvalue =
  latitude[N_ELEMENTS(latitude)-1]
  ELSE: IF ABS(latitude[binLat] - numberLat) GT
  ABS(latitude[binLat+1] - numberLat) THEN $
    closestLatvalue = latitude[binLat+1] ELSE closestLatvalue =
  latitude[binLat]
ENDCASE
differenceLat = MIN(ABS(latitude - numberLat), indexLat)
print, 'closest Lat index in data=', closestLat
print, 'closest Lat value in data=', closestLatvalue
print, 'difference between closestLatvalue & numberLat=',
  differenceLat
;-------------------------------------------------------------------
  ----------------------
;Quick Search Routine for Longitude
numberLon = obslong(j)
print, 'observed Longitude', numberLon
binLon = VALUE_LOCATE(longitude, numberLon)
print, 'LON & LON+1', longitude[binLon], longitude[binLon+1]

; Check with two values that bracket the chosen number.
CASE 3 OF
  binLon EQ -1: closestLon = longitude[0]
  binLon EQ (N_ELEMENTS(longitude)-1): closestLon =
  longitude[N_ELEMENTS(longitude)-1]
  ELSE: IF numberLon GE 358. THEN closestLon = binLon ELSE IF
  ABS(longitude[binLon] - numberLon) GT ABS(longitude[binLon+1] -
  numberLon) THEN $
    closestLon = binLon+1 ELSE closestLon = binLon
ENDCASE
```

```
124 CASE 4 OF
125   binLon EQ -1: closestLonvalue = longitude[0]
126   binLon EQ (N_ELEMENTS(longitude)-1): closestLonvalue =
    longitude[N_ELEMENTS(longitude)-1]
127   ELSE: IF numberLon GE 358. THEN closestLon = binLon ELSE IF
    ABS(longitude[binLon] - numberLon) GT ABS(longitude[binLon+1] -
    numberLon) THEN $
128     closestLonvalue = longitude(binLon+1) ELSE closestLonvalue =
    longitude(binLon)
129 ENDCASE
130 differenceLon = MIN(ABS(longitude - numberLon), indexLon)
131 print, 'closest Lon index in data=', closestLon
132 print, 'closest Lon value in data=', closestLonvalue
133 print, 'difference between closestLonvalue & numberLon=',
    differenceLon
134 ;-------------------------------------------------------------------
    ------------------------
135 ;Quick Search Routine for Time
136 numberTime = obstime(4,j)
137 print, 'observed time', numberTime
138 binTime = VALUE_LOCATE(time, numberTime)
139 print, 'TIME & TIME+1', time[binTime], time[binTime+1]
140
141 ; Check with two values that bracket the chosen number.
142 CASE 4 OF
143   binTime EQ -1: closestTime = time[0]
144   binTime EQ (N_ELEMENTS(time)-1): closestTime =
    time[N_ELEMENTS(time)-1]
145   ELSE: IF numberTime GE 57. THEN closestTime = binTime ELSE IF
    ABS(time[binTime] - numberTime) GT ABS(time[binTime+1] - numberTime)
    THEN $
146     closestTime = binTime+1 ELSE closestTime = binTime
147 ENDCASE
148 CASE 5 OF
149   binTime EQ -1: closestTimevalue = time[0]
150   binTime EQ (N_ELEMENTS(time)-1): closestTimevalue =
    time[N_ELEMENTS(time)-1]
151   ELSE: IF numberTime GE 57. THEN closestTime = binTime ELSE IF
    ABS(time[binTime] - numberTime) GT ABS(time[binTime+1] - numberTime)
    THEN $
152     closestTimevalue = time(binTime+1) ELSE closestTimevalue =
    time(binTime)
153 ENDCASE
154 differenceTime = MIN(ABS(time - numberTime), indexTime)
155 print, 'closest Time index in data=', closestTime
156 print, 'closest Time value in data=', closestTimevalue
157 print, 'difference between closestTimevalue & numberTime=',
```

```
   differenceTime
158;-------------------------------------------------------------------
   ----------------------
159; Quick Search Routine for Height to determine the Pressure level
   index
160numberHt = (obsheight(j)/1000.)
161print, 'observed Height', numberHt
162heightlock = height(closestLon,closestLat,*,closestTime)
163binHt = VALUE_LOCATE(heightlock, numberHt)
164print, 'HT & HT+1', heightlock[binHt], heightlock[binHt+1]
165CASE 6 OF
166  binHt EQ -1: closestHt =
   height[closestLon,closestLat,0,closestTime]
167  binHt EQ (N_ELEMENTS(heightlock)-1): closestHt =
   heightlock[N_ELEMENTS(heightlock)-1]
168  ELSE: IF ABS(heightlock[binHt] - numberHt) GT
   ABS(heightlock[binHt+1] - numberHt) THEN $
169    closestHt = binHt+1 ELSE closestHt = binHt
170ENDCASE
171CASE 7 OF
172  binHt EQ -1: closestHtvalue =
   height[closestLon,closestLat,0,closestTime]
173  binHt EQ (N_ELEMENTS(heightlock)-1): closestHtvalue =
   heightlock[N_ELEMENTS(heightlock)-1]
174  ELSE: IF ABS(heightlock[binHt] - numberHt) GT
   ABS(heightlock[binHt+1] - numberHt) THEN $
175    closestHtvalue = heightlock[binHt+1] ELSE closestHtvalue =
   heightlock[binHt]
176ENDCASE
177differenceHt = MIN(ABS(heightlock - numberHt), indexHt)
178print, 'closest Ht index in data=', closestHt
179print, 'closest Ht value in data=', closestHtvalue
180print, 'difference between closestHtvalue & numberHt=', differenceHt
181;-------------------------------------------------------------------
   ----------------------
182;inital parameters for density
183height0 = height(binLon, binLat, binHt, closestTime)
184IF binLon GE 191 THEN height1 = height0 ELSE IF binLon LT 191 THEN
   height1 = height((binLon+1), binLat, binHt, closestTime)
185height2 = height(binLon, (binLat+1), binHt, closestTime)
186height3 = height(binLon, binLat, (binHt+1), closestTime)
187IF binLon GE 191 THEN height4 = height2 ELSE IF binLon LT 191 THEN
   height4 = height((binLon+1), (binLat+1), binHt, closestTime)
188IF binLon GE 191 THEN height5 = height3 ELSE IF binLon LT 191 THEN
   height5 = height((binLon+1), binLat, (binHt+1), closestTime)
189height6 = height(binLon, (binLat+1), (binHt+1), closestTime)
190IF binLon GE 191 THEN height7 = height6 ELSE IF binLon LT 191 THEN
```

```
   height7 = height((binLon+1), (binLat+1), (binHt+1), closestTime)
191
192 density0 = density(binLon, binLat, binHt, closestTime)
193 IF binLon GE 191 THEN density1 = density0 ELSE IF binLon LT 191 THEN
   density1 = density((binLon+1), binLat, binHt, closestTime)
194 density2 = density(binLon, (binLat+1), binHt, closestTime)
195 density3 = density(binLon, binLat, (binHt+1), closestTime)
196 IF binLon GE 191 THEN density4 = density2 ELSE IF binLon LT 191 THEN
   density4 = density((binLon+1), (binLat+1), binHt, closestTime)
197 IF binLon GE 191 THEN density5 = density3 ELSE IF binLon LT 191 THEN
   density5 = density((binLon+1), binLat, (binHt+1), closestTime)
198 density6 = density(binLon, (binLat+1), (binHt+1), closestTime)
199 IF binLon GE 191 THEN density7 = density6 ELSE IF binLon LT 191 THEN
   density7 = density((binLon+1), (binLat+1), (binHt+1), closestTime)
200
201 ;Log Interpolation
202 f1 = 1/(((height3-numberHt)/(numberHt-height0))+1)
203 rho1 = ((density3)^f1)*((density0)^(1-f1))
204 f2 = 1/(((height6-numberHt)/(numberHt-height2))+1)
205 rho2 = ((density6)^f2)*((density2)^(1-f2))
206 f3 = 1/(((height7-numberHt)/(numberHt-height4))+1)
207 rho3 = ((density7)^f3)*((density4)^(1-f3))
208 f4 = 1/(((height5-numberHt)/(numberHt-height1))+1)
209 rho4 = ((density5)^f4)*((density1)^(1-f4))
210
211 ;Bilinear Interpolation
212 Lon0 = longitude(binLon)
213 IF binLon GE 191 THEN Lon1=Lon0 ELSE IF binLon LT 191 THEN Lon1 =
   longitude(binLon+1)
214 IF Lon1 EQ Lon0 THEN f5=0 ELSE IF binLon LT 191 THEN f5 = 1/
   (((Lon1-numberLon)/(numberLon-Lon0))+1)
215 rho5 = f5*rho3 + (1-f5)*rho2
216 rho6 = f5*rho4 + (1-f5)*rho1
217
218 Lat0 = latitude(binLat)
219 Lat1 = latitude(binLat+1)
220 f6 = 1/(((Lat1-numberLat)/(numberLat-Lat0))+1)
221 rhoWAM = f6*rho5 + (1-f6)*rho6
222 print, rhoWAM
223 PRINTF,1, rhoWAM
224 toc
225 ENDFOR
226 CLOSE,/all
227 ;-------------------------------------------------------------------
   ----------------------
228 RETURN
229 END
```

bigfastyearprog.pro

```
; Martin McCandless Code
; NOAA SWPC 7-5-2018
; *NOTE ONLY A TEST*

PRO BigFastYearProg, channel

rhoWAM1 = 'C:\Users\Martin
 McCandless\Desktop\NOAA\Mar16-19_big\temperature_output.txt'
OPENW,1,rhoWAM1
obsheight = DBLARR(1,34546)
OPENR,lun,'C:\Users\Martin
 McCandless\Desktop\NOAA\Mar16-19\height.txt',/get_lun
READF,lun,obsheight
FREE_LUN, lun
obslong = DBLARR(1,34546)
OPENR,lun,'C:\Users\Martin
 McCandless\Desktop\NOAA\Mar16-19\longitude.txt',/get_lun
READF,lun,obslong
FREE_LUN,lun
obslat = DBLARR(1,34546)
OPENR,lun,'C:\Users\Martin
 McCandless\Desktop\NOAA\Mar16-19\latitude.txt',/get_lun
READF,lun,obslat
FREE_LUN, lun
obstime = DBLARR(5,34546)
OPENR,lun,'C:\Users\Martin
 McCandless\Desktop\NOAA\Mar16-19\16-19time_new.txt',/get_lun
READF,lun,obstime
FREE_LUN, lun

filename = 'C:\Users\Martin
 McCandless\Desktop\NOAA\Mar16-19_del8\zz_3min_20130319_nsphys1_del8_b
 ig.nc
file = NCDF_OPEN(filename)
a = NCDF_VARID(file, 'lat')
b = NCDF_VARID(file, 'lon')
c = NCDF_VARID(file, 'DEN')
d = NCDF_VARID(file, 'lev')
e = NCDF_VARID(file, 'time')
f = NCDF_VARID(file, 't')
NCDF_VARGET, file, a, latitude
NCDF_VARGET, file, b, longitude
NCDF_VARGET, file, c, density
NCDF_VARGET, file, d, height
NCDF_VARGET, file, e, time

FOR j = 0,8634 DO BEGIN ;8634,17269,25907
```

```
41
42 numberLat = obslat(j)
43 binLat = VALUE_LOCATE(latitude, numberLat)
44 CASE 1 OF
45   binLat EQ -1: closestLat = latitude[0]
46   binLat EQ (N_ELEMENTS(latitude)-1): closestLat =
   latitude[N_ELEMENTS(latitude)-1]
47   ELSE: IF ABS(latitude[binLat] - numberLat) GT
   ABS(latitude[binLat+1] - numberLat) THEN $
48     closestLat = binLat+1 ELSE closestLat = binLat
49 ENDCASE
50 numberLon = obslong(j)
51 binLon = VALUE_LOCATE(longitude, numberLon)
52 CASE 2 OF
53   binLon EQ -1: closestLon = longitude[0]
54   binLon EQ (N_ELEMENTS(longitude)-1): closestLon =
   longitude[N_ELEMENTS(longitude)-1]
55   ELSE: IF numberLon GE 358. THEN closestLon = binLon ELSE IF
   ABS(longitude[binLon] - numberLon) GT ABS(longitude[binLon+1] -
   numberLon) THEN $
56     closestLon = binLon+1 ELSE closestLon = binLon
57 ENDCASE
58 numberTime = obstime(4,j)
59 binTime = VALUE_LOCATE(time, numberTime)
60 CASE 3 OF
61   binTime EQ -1: closestTime = time[0]
62   binTime EQ (N_ELEMENTS(time)-1): closestTime =
   time[N_ELEMENTS(time)-1]
63   ELSE: IF numberTime GE 57. THEN closestTime = binTime ELSE IF
   ABS(time[binTime] - numberTime) GT ABS(time[binTime+1] - numberTime)
   THEN $
64     closestTime = binTime+1 ELSE closestTime = binTime
65 ENDCASE
66 numberHt = (obsheight(j)/1000.)
67 binHt = VALUE_LOCATE(height, numberHt)
68 CASE 4 OF
69   binHt EQ -1: closestHt = height[0]
70   binHt EQ (N_ELEMENTS(height)-1): closestHt =
   height[N_ELEMENTS(height)-1]
71   ELSE: IF ABS(height[binHt] - numberHt) GT ABS(height[binHt+1] -
   numberHt) THEN $
72     closestHt = binHt+1 ELSE closestHt = binHt
73 ENDCASE
74
75 density0 = density(binLon, binLat, binHt, closestTime)
76 IF binLon GE 191 THEN density1 = density0 ELSE IF binLon LT 191 THEN
   density1 = density((binLon+1), binLat, binHt, closestTime)
```

```
density2 = density(binLon, (binLat+1), binHt, closestTime)
density3 = density(binLon, binLat, (binHt+1), closestTime)
IF binLon GE 191 THEN density4 = density2 ELSE IF binLon LT 191 THEN
  density4 = density((binLon+1), (binLat+1), binHt, closestTime)
IF binLon GE 191 THEN density5 = density3 ELSE IF binLon LT 191 THEN
  density5 = density((binLon+1), binLat, (binHt+1), closestTime)
density6 = density(binLon, (binLat+1), (binHt+1), closestTime)
IF binLon GE 191 THEN density7 = density6 ELSE IF binLon LT 191 THEN
  density7 = density((binLon+1), (binLat+1), (binHt+1), closestTime)

;log Interpolation
h1=height(binHt)
h2=height(binHt+1)
f1 = 1/(((h2-numberHt)/(numberHt-h1))+1)
rho1 = ((density3)^f1)*((density0)^(1-f1))
rho2 = ((density6)^f1)*((density2)^(1-f1))
rho3 = ((density7)^f1)*((density4)^(1-f1))
rho4 = ((density5)^f1)*((density1)^(1-f1))

;Bilinear Interpolation
Lon0 = longitude(binLon)
IF binLon GE 191 THEN Lon1=Lon0 ELSE IF binLon LT 191 THEN Lon1 =
  longitude(binLon+1)
IF Lon1 EQ Lon0 THEN f5=0 ELSE IF binLon LT 191 THEN f5 = 1/
  (((Lon1-numberLon)/(numberLon-Lon0))+1)
rho5 = f5*rho3 + (1-f5)*rho2
rho6 = f5*rho4 + (1-f5)*rho1

Lat0 = latitude(binLat)
Lat1 = latitude(binLat+1)
f6 = 1/(((Lat1-numberLat)/(numberLat-Lat0))+1)
rhoWAM = f6*rho5 + (1-f6)*rho6
print, rhoWAM
PRINTF,1, rhoWAM
ENDFOR
NCDF_CLOSE,file
  CLOSE,/all

  ;-------------------------------------------------------------------
  ----------------------
  RETURN
END
```

correlationmar16to19prog.pro

```
; Martin McCandless Code
; NOAA SWPC 6-8-2018
; *NOTE ONLY A TEST*

PRO CorrelationMar16to19Prog, channel

; Call arrays of models and GOCE data for Mar16-19th----------------------------------------
WAMIPE = dblarr(1,34546);8635,8638
openr,lun,'C:\Users\Martin McCandless\Desktop\NOAA\Mar16-19_big\OG_TMP_BigWAM.txt',/get_lun
readf,lun,WAMIPE
close,/all
help,WAMIPE
print,'WAMIPE',WAMIPE(15921)

GOCE = dblarr(1,34546)
openr,lun,'C:\Users\Martin McCandless\Desktop\NOAA\Mar16-19\SMOOTH_GOCE.txt',/get_lun
readf,lun,GOCE
close,/all
help,GOCE
print,'GOCEun',GOCE(15921)

CTIPe = dblarr(1,34546)
openr,lun,'C:\Users\Martin McCandless\Desktop\NOAA\Smooth\OG_CTIPe.txt',/get_lun
readf,lun,CTIPe
close,/all
help,CTIPe
print,'CTIPe',CTIPe(15921)

; n = number of data points in the time series
n0=34546 ;34546 total number of points
print, '|***************16-19th WAMvsCTIPe***************|'
R0=correlate(WAMIPE,GOCE)
rmse_ln0 = sqrt((total((alog(WAMIPE/GOCE))^2))/n0)
bias_ln0 = exp((total(alog(WAMIPE/GOCE)))/n0)
sd_ln0 = sqrt((total(((alog(WAMIPE/GOCE))-(alog(bias_ln0)))^2))/n0)
rmse_ln_check0 = (alog(bias_ln0)^2)+(sd_ln0)^2
print, 'R', R0
print, 'rmse_ln', rmse_ln0
print, 'bias_ln', bias_ln0
print, 'sd_ln', sd_ln0
print, 'rmse_ln_check', rmse_ln_check0
print, '|*********************************************|'

```

```
;boxcar averaged smoothing functions

sCTIPe = SMOOTH(CTIPe,538)
sCTIPe1 = '/gpfs/u/home/martinm/smooth_CTIPe_Temp.txt'
OPENW,1,sCTIPe1
PRINTF,1, sCTIPe
CLOSE,1

sGOCE = SMOOTH(GOCE,538)
sGOCE1 =
  'C:\Users\martin.mccandless\Desktop\NOAA\Jan14-20\smooth_GOCE_den.txt'
OPENW,1,sGOCE1
PRINTF,1, sGOCE
CLOSE,1

sWAMIPE = SMOOTH(WAMIPE,538)
sWAMIPE1 = 'C:\Users\Martin
  McCandless\Desktop\NOAA\Mar16-19_big\SMOOTH_TMP_Big_WAM.txt'
OPENW,1,sWAMIPE1
PRINTF,1, sWAMIPE
CLOSE,1

y = WAMIPE
x = GOCE

myPlot = SCATTERPLOT(x, y, SYMBOL='.',XRANGE=
  [0.00000000003,0.00000000015], YRANGE=[0.00000000003,0.00000000015],
  TITLE='2013 March 16-19 Orbit-Averaged', YTITLE='BIG_WAM Neutral
  Density [kg/m3]',SYM_SIZE=1.0,XTITLE='GOCE Neutral Density [kg/m3]')

myLine = LINFIT(x, y, CHISQR=chisqr, COVAR=covar,
  MEASURE_ERRORS=measures, PROB=prob, SIGMA=sigma,YFIT=yfit)
print, myLine
myX = FINDGEN(25000,INCREMENT=0.000000000000015)
myY = ((1.0176222)*(myX) + (0.00000000000090150199))
myLineEq = plot(myX, myY, /OVERPLOT, COLOR='red', THICK=4,
  LINESTYLE=0, XRANGE=[0.00000000003,0.00000000015], YRANGE=
  [0.00000000003,0.00000000015])
myY1 = myX
StdLine = plot(myX,myY1, /OVERPLOT, COLOR='black', THICK=4,
  LINESTYLE=5, XRANGE=[0.00000000003,0.00000000015], YRANGE=
  [0.00000000003,0.00000000015])

return
end
```