PyIRI: Whole-Globe Approach to the International Reference Ionosphere Modeling Implemented in Python

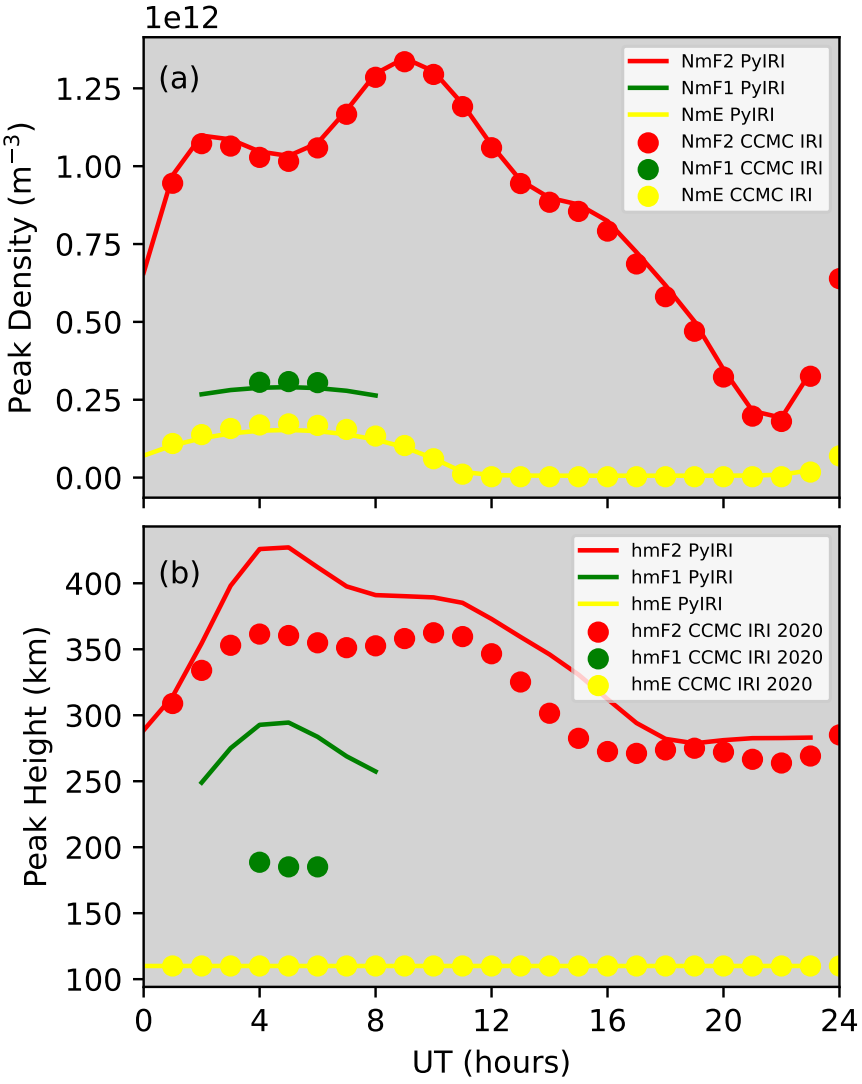
Victoriya Forsythe[1], Dieter Bilitza[2], Angeline Gail Burrell[3], Kenneth F. Dymond[4], Bruce Aaron Fritz[1], and Sarah E McDonald[4]

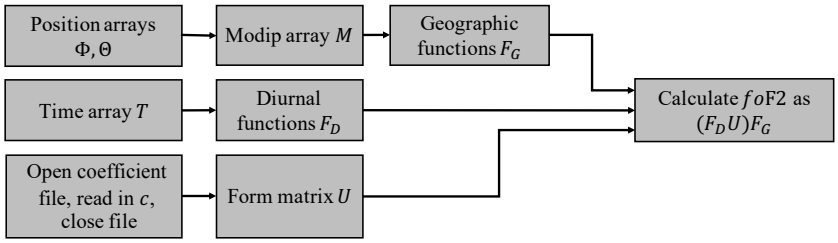
[1]U.S. Naval Research Laboratory
[2]George Mason University
[3]US Naval Research Laboratory
[4]Naval Research Laboratory

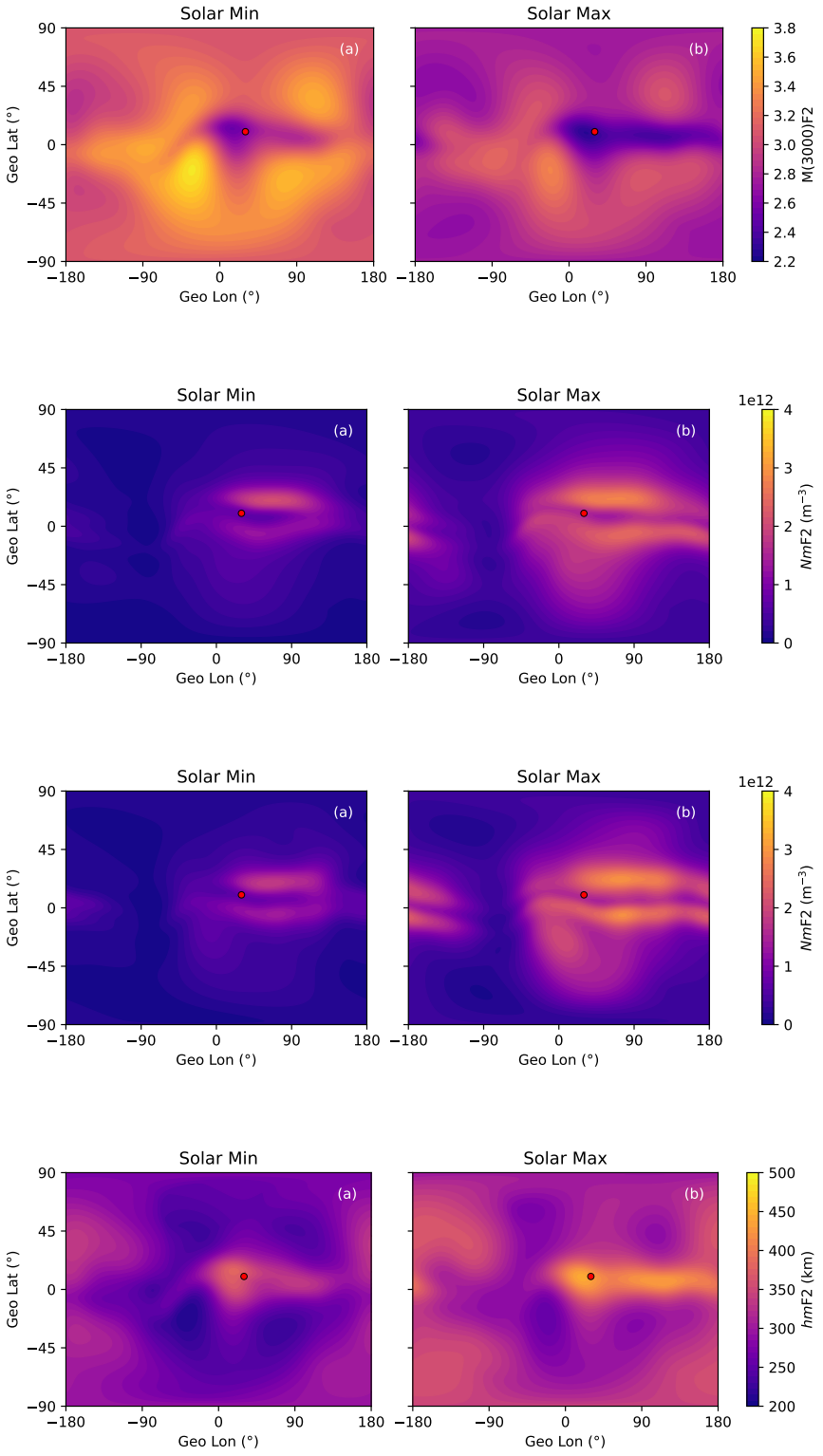
September 28, 2023

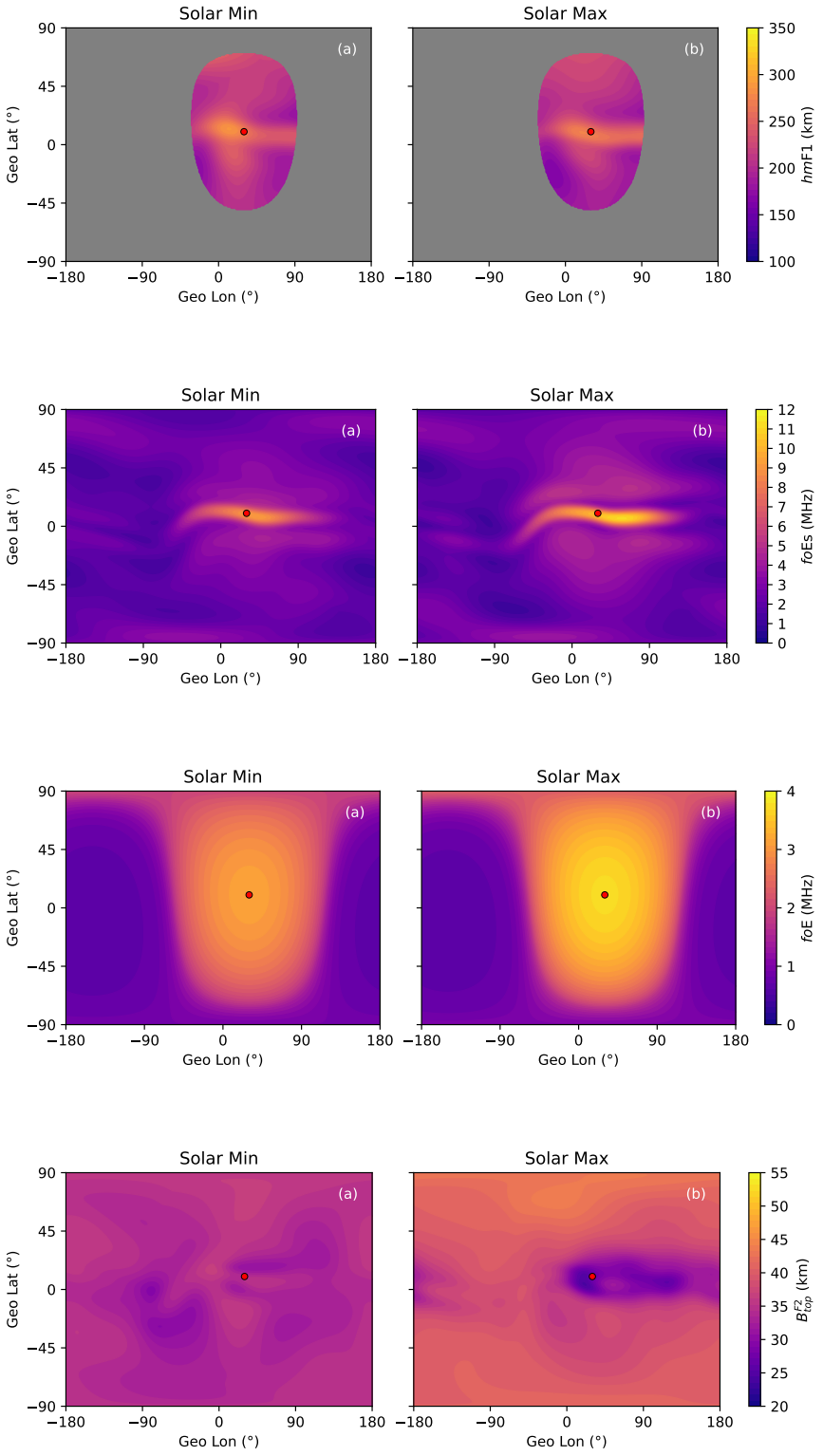
Abstract

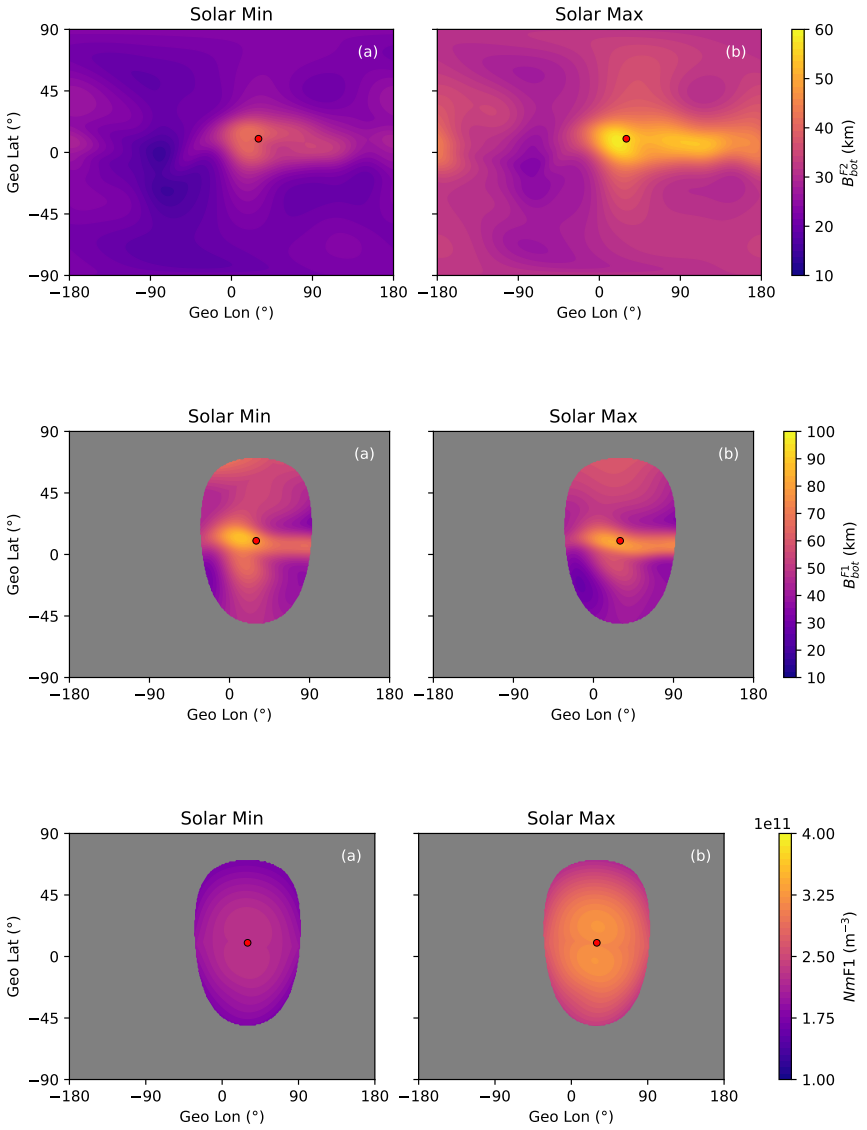
The International Reference Ionosphere (IRI) model is widely used in the ionospheric community and considered the gold standard for empirical ionospheric models. The development of this model was initiated in the late 1960s using the FORTRAN language; for its programming approach, the model outputs were calculated separately for each given geographic location and time stamp. The Consultative Committee on International Radio (CCIR) and International Union of Radio Science (URSI) coefficients provide the skeleton of the IRI model, as they define the global distribution of the maximum usable ionospheric frequency foF2 and the propagation factor M(3000)F2. At the U.S. Naval Research Laboratory (NRL), a novel Python tool was developed that enables global runs of the IRI model with significantly lower computational overhead. This was made possible through the Python rebuild of the core IRI component (which calculates ionospheric critical frequency using the CCIR or URSI coefficients), taking advantage of NumPy matrix multiplication instead of using cyclic addition. This paper explains in detail this new approach and introduces all components of the PyIRI package.

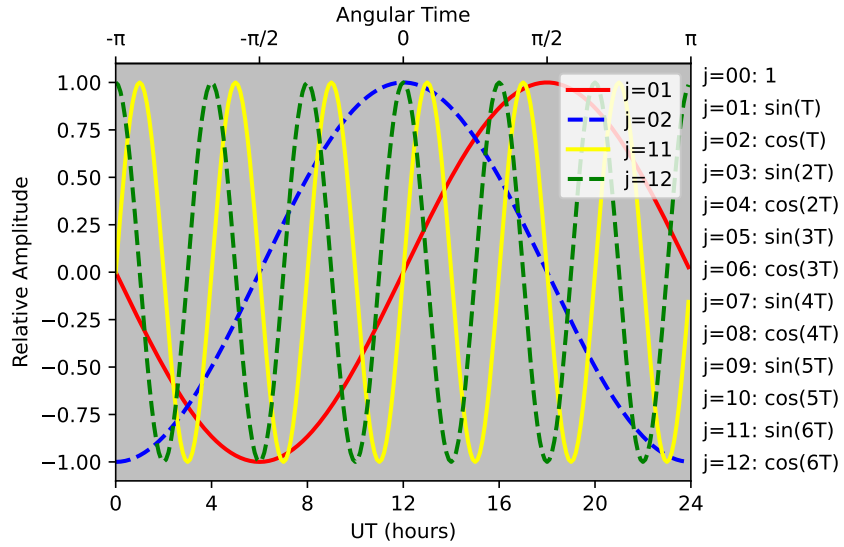


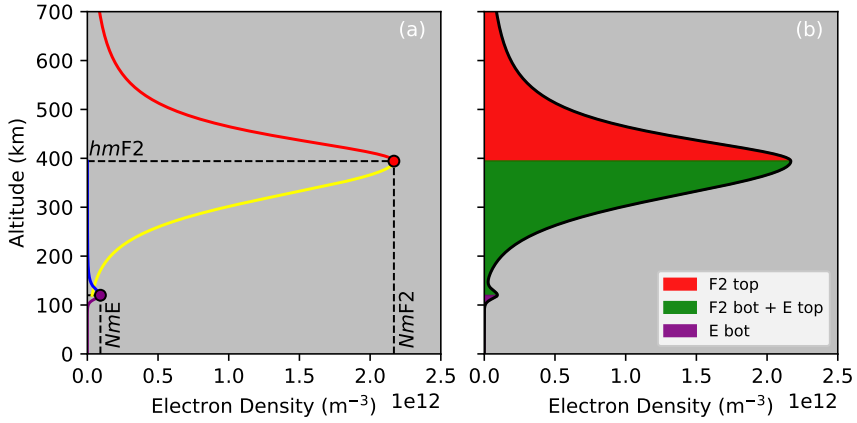


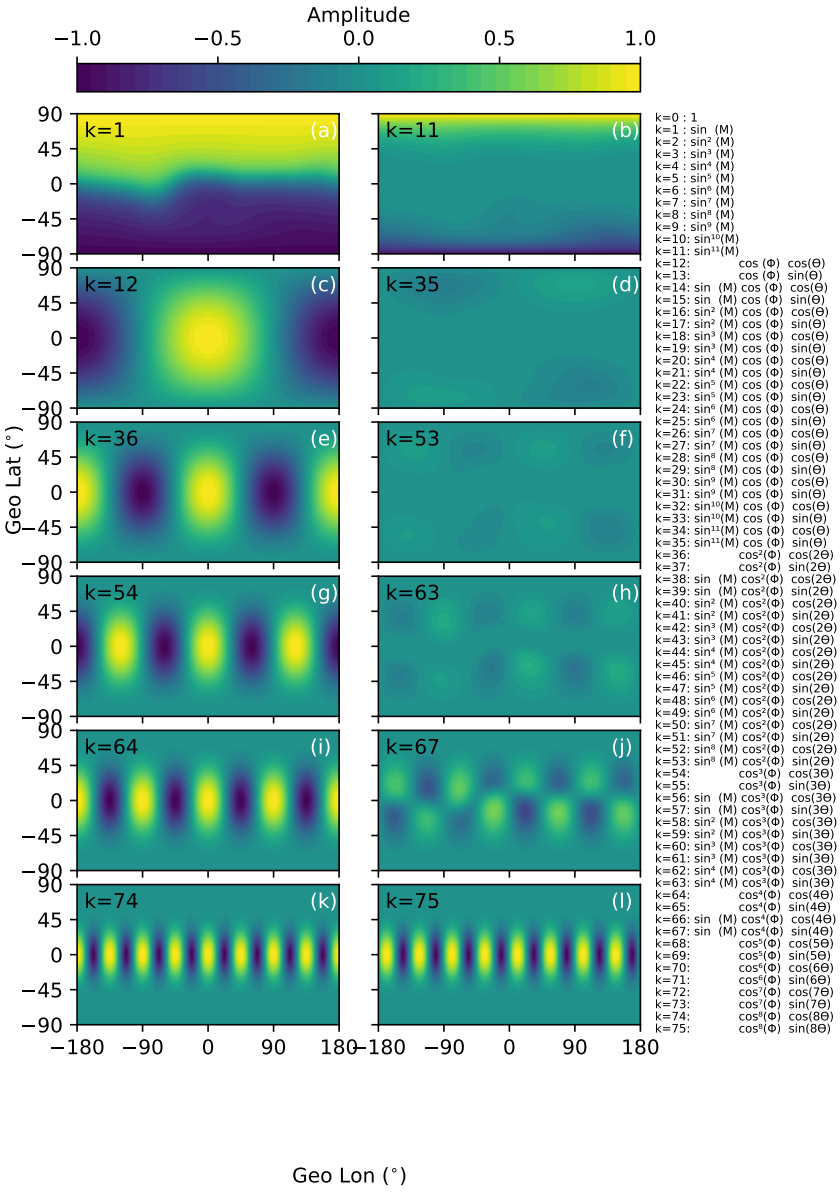


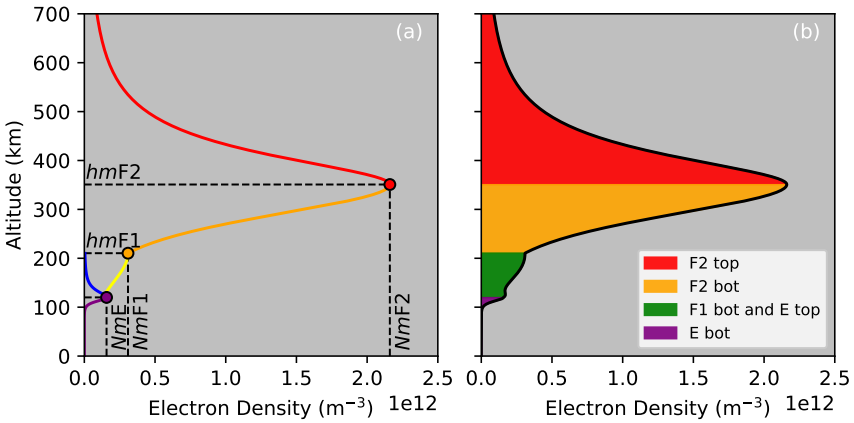


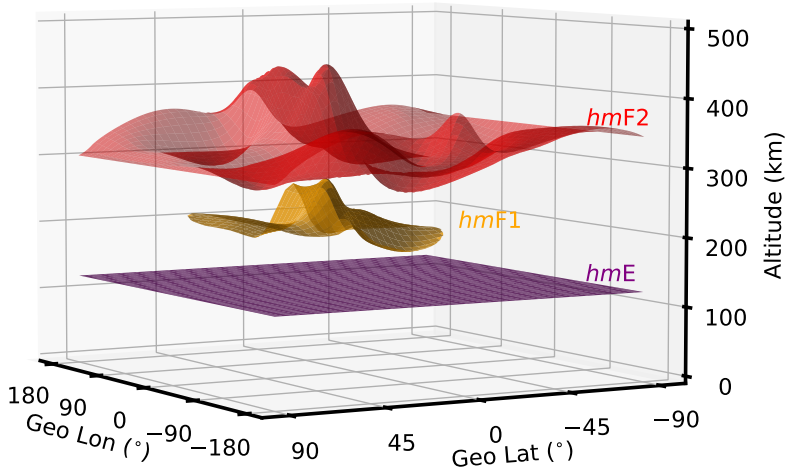


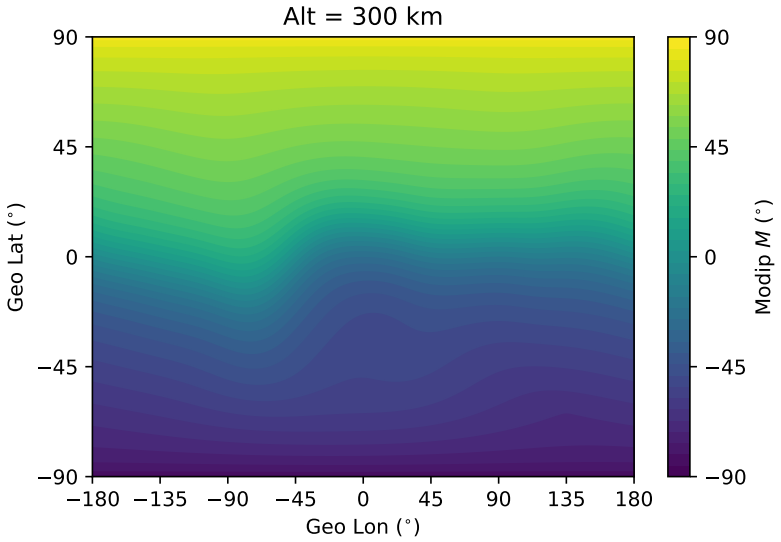


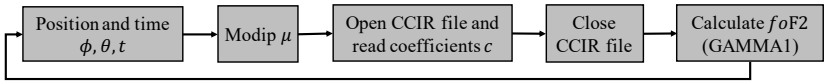


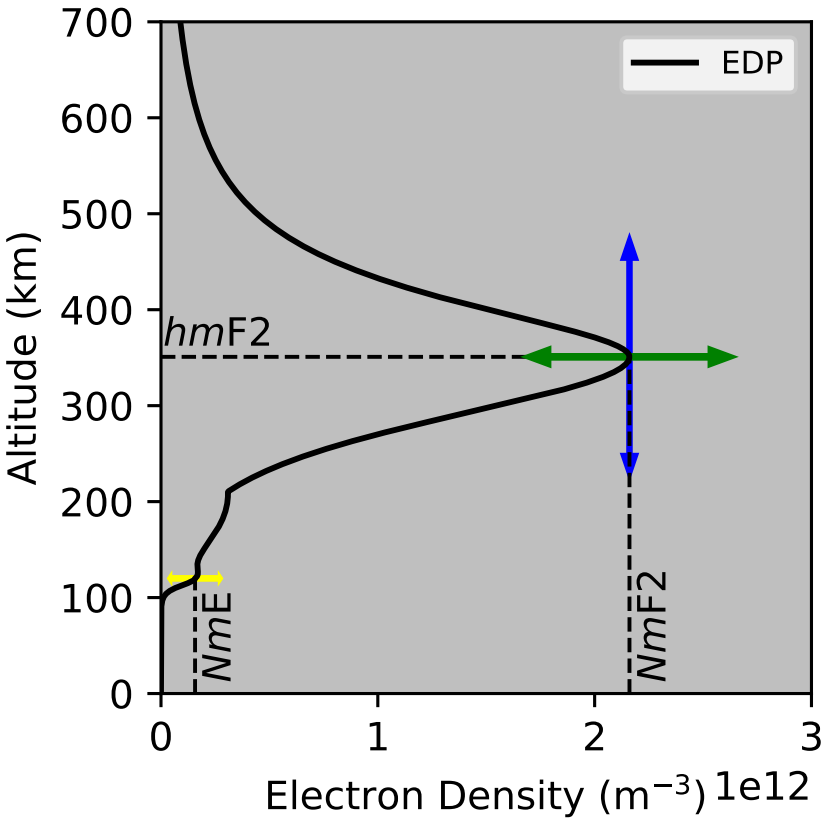












PyIRI: Whole-Globe Approach to the International Reference Ionosphere Modeling Implemented in Python

Victoriya V. Forsythe[1], Dieter Bilitza[2,3], Angeline G. Burrell[1], Kenneth F. Dymond[1], Bruce A. Fritz[1], Sarah E. McDonald[1]

[1]U.S. Naval Research Laboratory, Washington, DC, USA
[2]Department of Physics and Astronomy, George Mason University, Fairfax, VA, USA
[3]Heliospheric Laboratory, NASA Goddard Space Flight Center, Greenbelt, MD, USA

Distribution Statement

Key Points:

- Python tool for rapid global ionospheric electron density estimates
- Novel approach to running the core of the IRI model
- 24-hour global electron density in a few seconds

Corresponding author: Victoriya V. Forsythe, victoriya.makarevich@nrl.navy.mil

Abstract

The International Reference Ionosphere (IRI) model is widely used in the ionospheric community and considered the gold standard for empirical ionospheric models. The development of this model was initiated in the late 1960s using the FORTRAN language; for its programming approach, the model outputs were calculated separately for each given geographic location and time stamp. The Consultative Committee on International Radio (CCIR) and International Union of Radio Science (URSI) coefficients provide the skeleton of the IRI model, as they define the global distribution of the maximum usable ionospheric frequency foF2 and the propagation factor $M(3000)$F2. At the U.S. Naval Research Laboratory (NRL), a novel Python tool was developed that enables global runs of the IRI model with significantly lower computational overhead. This was made possible through the Python rebuild of the core IRI component (which calculates ionospheric critical frequency using the CCIR or URSI coefficients), taking advantage of NumPy matrix multiplication instead of using cyclic addition. This paper explains in detail this new approach and introduces all components of the PyIRI package.

Plain Abstract

The International Reference Ionosphere (IRI) estimates the number of electrons in the upper atmosphere, which is important to know for the ground- and space-based communication and investigation. Scientists and communication specialists often use IRI to plan future and ongoing operations. The core software of the IRI model was written in the late 1960s, when arrays and matrices were not practical to implement given computational limitations. This means that IRI evaluates the electron density at each geographic location and desired time separately. This causes a long processing time for high-resolution, global IRI runs, even with modern computers. We have introduced modern programming approaches to the IRI code by building a Python tool, PyIRI, that enables estimation of the electron density at all desired geographic locations simultaneously. With PyIRI, it takes just a few seconds to obtain the electron density for a day over the entire globe.

1 Introduction

The ionosphere is a region within the upper atmosphere that surrounds the Earth starting from ∼80 km altitude and extending all the way into the exosphere thousands

of km above the Earth's surface. Unlike the neutral atmosphere, the ionosphere has free electrons that refract electromagnetic waves, especially the waves that have frequencies below 300 MHz. Therefore, in order to establish the high frequency (HF) communication link between any two positions it is crucial to know the amount of the electrons along the signal path. The International Reference Ionosphere (IRI) empirical model estimates the electron density in the ionosphere based on a statistical analysis of ionospheric climatology over four years.

The International Standardization Organization (ISO), the International Union of Radio Science (URSI), the Committee on Space Research, and the European Cooperation for Space Standardization have all recognized IRI as the official standard for the Earth's ionosphere (ISO 16457: https://www.iso.org/standard/61556.html). A recent review paper by Bilitza et al. (2022) describes the current state of the IRI model, its history, and recent developments.

It is important to mention that there are several Python IRI wrappers and interfaces, e.g. `iri2016` (Ilma, 2017). However, they merely wrap the original FORTRAN IRI code to make its execution more convenient for Python users. This work, for the first time, introduces a novel software package that redefines the core of the IRI fully in the Python language.

Despite being the gold standard of ionospheric modeling, not all parts of the IRI software are efficient. The CCIR models were developed in the early 1960s and use a punch card FORTRAN language, as was suitable for that period of time. Unfortunately, resources for updating IRI to modern programming standards were not provided. As a result, the current execution of the IRI code is based on the sequential calculation of the electron density for each desired time and geographic location separately. As an example, to obtain the global density distribution over 24-hours, one needs to execute the IRI model $N_G \times N_T$ times, where N_G is the number of horizontal locations, and N_T is the number of diurnal time frames. Considering a typical global regular grid of $1^\circ \times 1^\circ$ and a 15-min temporal resolution, the number of executions is equal to 6,272,736. Even given the number crunching power of FORTRAN on modern computers, this number of serial calculations will take a significant amount of time. Further consider different typical use cases for an ionospheric model, including the need to analyze different seasonal dependencies, solar conditions, or (a nightmare) to construct an ensemble of the global

density distributions. This last example was the motivation to rethink and rebuild the current IRI code base, with the goal of developing a software implementation that allows for the simultaneous calculation of the electron density on the entire globe over a given day. It involves the total rebuild of the IRI core, namely the calculation of the $NmF2$ and $hmF2$ parameters from the CCIR (or URSI for $NmF2$) coefficients. In other words, what previously required 6,272,736 executions is now possible to obtain in only one step. To be more precise, the FORTRAN IRI code requires 934.15 hours to obtain 6,272,736 executions, but takes only 8 min using PyIRI.

The focus of this paper is on the global and rapid construction of the main core of the IRI model, which is the climatology of the ionospheric peak density $NmF2$ and its height $hmF2$. These parameters represent the skeleton of the model, because the electron density profile is normalized to these parameters. The full version of the IRI model contains many options for the other parameters. For example, the parameters that describe the top side of the electron density profile (EDP) can be derived using 4 different approaches (Bilitza et al., 2022). This work will further present the construction of the EDP choosing the easiest approach. In some cases, the formalism from NeQuick model (Nava et al., 2008) was used instead of the IRI approach. For example, the bottom side of the F2 layer was chosen to depend on a single thickness parameter instead of two parameters. Additionally, the EDP construction further simplifies some traditional IRI methods to keep the focus of this paper on the core output of the IRI program. At the current state of development, it is fair to consider the PyIRI to be a mixture of the traditional IRI with some elements of the NeQuick model.

This paper starts by discussing the core of the IRI model and its use of the CCIR coefficients to obtain the main ionospheric parameters. It then introduces the Pythonic approach to construct the global maps of $NmF2$ and $hmF2$. Finally, it describes the derivation of other IRI components and introduces a novel Python IRI software package, PyIRI (Forsythe & Burrell, 2023), that has been made available to the community as free and open source software (FOSS), and can be found at `https://github.com/victoriyaforsythe/PyIRI`.

2 The core parameters of the IRI model

Despite a wide range of the different IRI options and internal sub-models, there are three main parameters that work as independent anchor points for the construction of the ionospheric EDP: the density of the F2 peak ($NmF2$), the height of the F2 peak ($hmF2$), and the density of the E region peak (NmE). An example EDP is shown in Figure 1, where the independent variation of the three anchor points are visualized with arrows. The $NmF2$ and $hmF2$ determine the position of the F2 region, while NmE controls the shape of the E region (the height of the E region peak is considered to be constant at 110 km). Other parameters that determine the shape of the EDP are the peak density of the F1 region and the thicknesses of the F2, F1, and E regions.

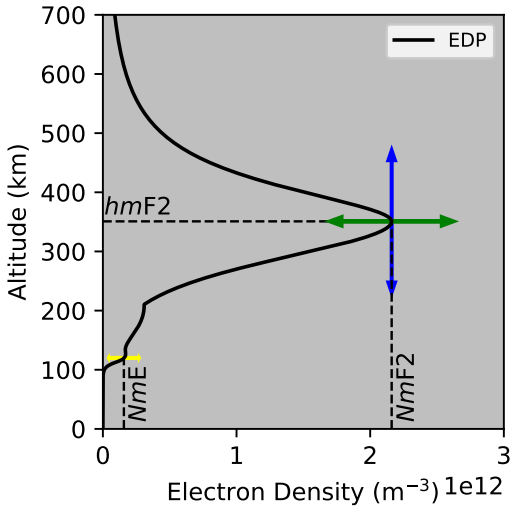


Figure 1. Three anchor points control the shape of EDP by adjusting their values as illustrated by the colored arrows. The peak of the F2 region is determined by the $NmF2$ (green arrows), the height of the peak is determined by the $hmF2$ (blue arrows), and the peak of the E region is described by NmE (yellow arrows) with fixed height of 110 km.

2.1 Calculating $NmF2$: FORTRAN approach

The IRI model and NeQuick model employ CCIR coefficients to obtain the diurnal and geographic variations of the ionospheric critical frequency, $foF2$. The $NmF2$ can be derived from $foF2$ using the plasma physics formula:

$$NmF2/\mathrm{m}^{-3} = 0.124 \times 10^{11} (foF2/\mathrm{MHz})^2. \quad (1)$$

The CCIR coefficients were obtained in the pioneering studies conducted by Jones and Gallet (1962, 1965) and Jones et al. (1966). They analyzed the monthly medians of the $foF2$ for minimum and maximum levels of solar activity. First they found the coefficients for the Fourier time series to represent the diurnal trends of $foF2$ at about 150 ionosonde stations using a Least Squares minimization. They then found the coefficients for a special set of geographic functions (similar to surface waves) to describe the variation of the diurnal Fourier coefficients with geographic location. As a result of their work, the diurnal and geographic variations of the monthly median $foF2$ measurements are described for two levels of solar activity using sets of monthly coefficients.

In mathematical terms, the diurnal variations at a geographic North latitude ϕ and East longitude θ, and at a particular time of the day in universal time (UT) t expressed as angle time from π to $-\pi$, the critical frequency can be expressed as:

$$foF2(\phi, \theta, t) = a_0(\phi, \theta) + \sum_{i=1}^{M} [a_{2i-1}(\phi, \theta) \cos(it) + a_{2i}(\phi, \theta) \sin(it)]. \quad (2)$$

In Equation 2 the maximum number of the harmonics is $M = 6$, and the geographic functions a_i are defined as

$$a_i(\phi, \theta) = \sum_{j=0}^{J(0)} c_{i,j,0} P_{j,0}(\phi, \theta) + \sum_{k=1}^{9} \sum_{j=0}^{J(k)} (c_{i,j,2k-1} \cos(k\phi) + c_{i,j,2k} \sin(k\phi)) \sin^j(\mu(\phi, \theta)) \cos^k(\phi), \quad (3)$$

where μ is the modified dip angle that can be calculated from the Earth's magnetic inclination, $I(\phi, \theta)$, as:

$$\mu = \arctan\left(\frac{I(\phi, \theta)}{\cos(\phi)}\right). \quad (4)$$

The summation cutoffs in Equation 3 limits j to $J(k)$ terms. The cutoff truncates the higher degrees of the latitudinal expansion and was introduced by Jones and Gallet (1962) to reduce the noise of the median data points. Specifically, $J(k) = [11, 11, 8, 4, 1, 0, 0, 0, 0]$ is employed for $foF2$. In cases where $J(k)$ has fewer elements, k would change to the lower number to be equal to the number of elements in $J(k)$.

Coefficients c in Equation 3 are provided for two levels of solar activity as the first 1,976 numbers in 12 CCIR files (one for each month) that accompany the IRI model.

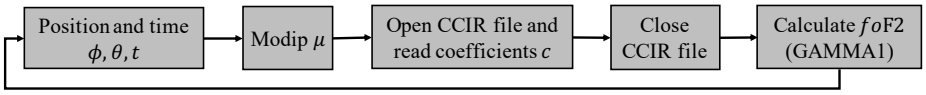


Figure 2. A simplified flow diagram of IRI model to obtain foF2 from CCIR coefficients.

In the IRI source code, a function called GAMMA1 calculates foF2 using Equations 2 and 3. The summation is obtained by sequential multiplication and addition of the coefficients with the functions inside of FORTRAN `do` loops. A simplified version of the flow diagram for this process is shown in Figure 2. For a particular time t and geographic position (ϕ, θ), the modip μ is obtained, then the coefficients c are read from the CCIR file and the function GAMMA1 is called to calculate foF2. For the next UT time frame or for the next location of interest, this same process is repeated until outputs for all times and locations are calculated. This scheme is simplified from the actual model, removing the interpolations performed for solar activity and day of month for each set of coefficients, that will be discussed in Section 5.

2.2 Calculating NmF2: Pythonic approach

This section describes a novel approach to calculate NmF2, which takes advantage of the fact that one set of CCIR coefficients contains all the necessary information to obtain the foF2 for the entire globe at all local times. This novel approach utilizes the efficient matrix handling of NumPy to reduce input/output (I/O) calls. To streamline the computational calculation of Equations 2 and 3, the order of operations shown in Figure 2 must be revised.

The first step is the formation of 1-D positional arrays, Φ and Θ that specify the desired paired latitude and longitude locations (with the size of each array being N_G). Capital Greek letters are used to emphasize that these are the arrays and not the single numbers. These arrays are coordinate pairs, they can describe regular, irregular, global, or regional grids. Similarly, the time of interest is specified as a 1-D array, T. The Python 3.7 package for International Geomagnetic Reference Field (IGRF-13) (Alken et al., 2021) is employed to calculate the magnetic inclination, I, for the Φ and Θ arrays. The modip array, M, is then calculated using Equation 4. The global distribution of M at 300 km

altitude is shown in Figure 3 for 1 Apr 2020. The M distribution specifies how much a compass needle will deviate from the horizontal plane and can be used to easily identify the magnetic equator and poles.

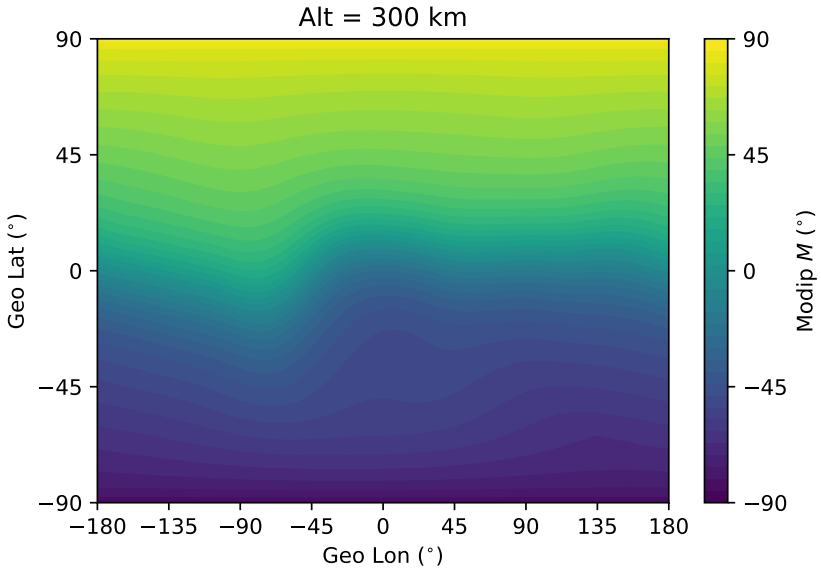


Figure 3. Example of the modified dip angle global distribution for 1 Apr, 2020.

Next, the glTobal function components from Equations 2 and 3 are evaluated using the positional arrays Φ, Θ, and M. Figure 4 shows several examples of the global function components. The main difference between these functions and regular spherical harmonics is the M dependency, which is clearly visible from the first function shown in Figure 4a. The smallest ionospheric structures that can be revealed by the highest expansion component is shown in Figure 4l. Additionally, the list of all 76 global functions is shown on the right side of Figure 4. As a result of this step, a matrix of global function components, F_G, has $[76, N_G]$ elements (where N_G is the number of geographic locations).

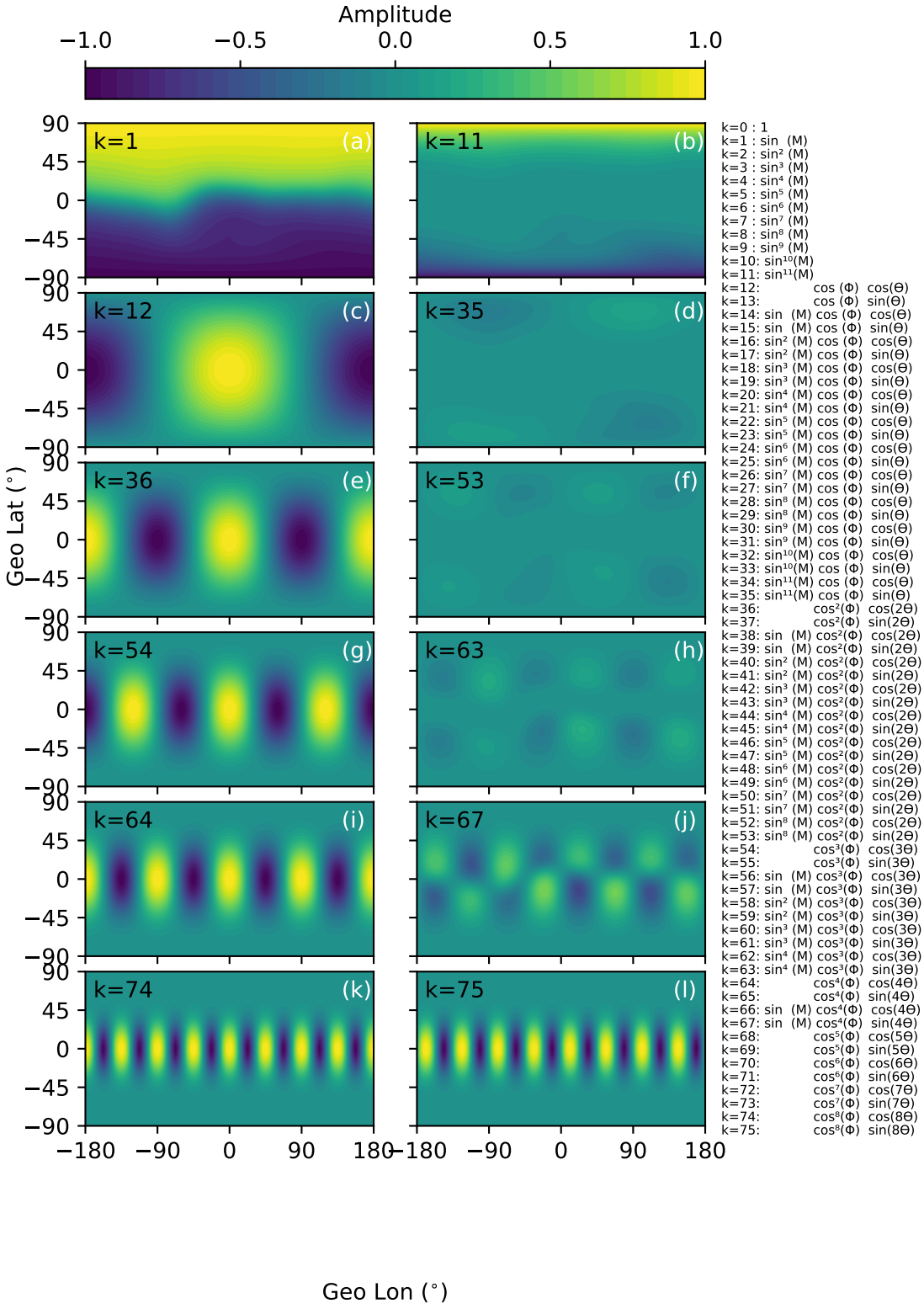


Figure 4. Few examples of the global functions calculated for the array of grid points. The list of all functions is shown on the right.

The diurnal variations are evaluated separately from the location variations using the Fourier function components shown in Equation 2. These components are evaluated for the given time array, T. Figure 5 shows the first two low-order and the last two high-order components, visualizing the highest level of the temporal resolution that can be achieved. A full list of components is shown on the right side of Figure 5. This calculation step produces a matrix of diurnal function components, F_D, with dimensions $[N_T, 13]$ (where N_T is number of time steps).

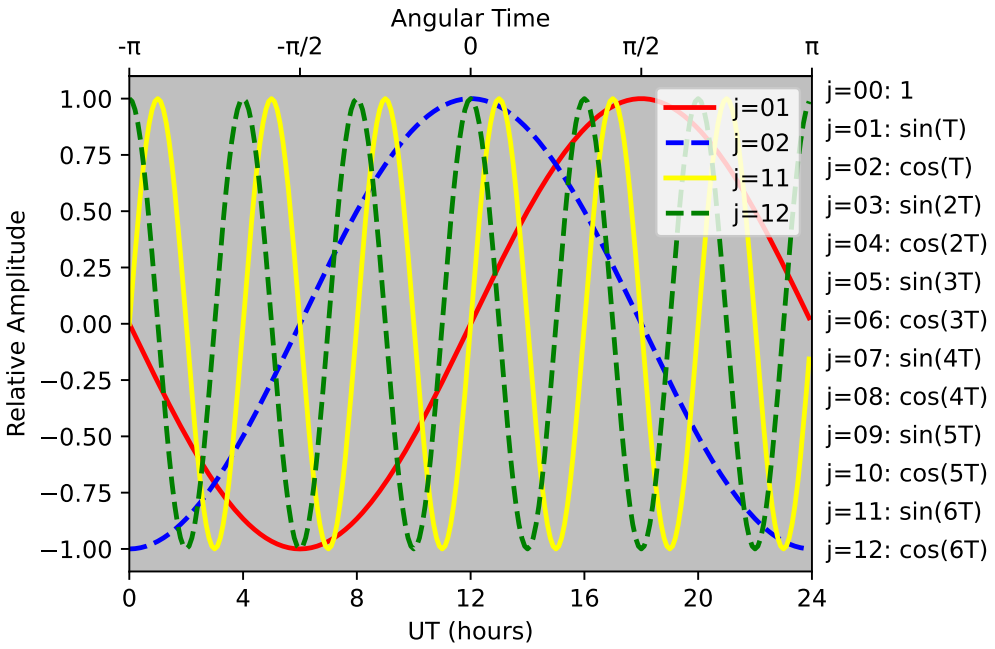


Figure 5. Few examples of the Fourier diurnal functions calculated for array of time spanned 24 hours of UT.

The final step before calculating foF2 is to create a CCIR (or URSI, if chosen) coefficient matrix. The 1-D array of coefficients, c, that has 1,976 elements can be reformed into a matrix U of size $[13, 76, 2]$, where the first dimension corresponds to 13 diurnal variations, the second dimension represents the 76 location variations, and the third dimension represents the two levels of solar activity. IRI uses linear interpolation to establish all other levels of solar activity. For one level of solar activity, or after the interpolation for a particular solar activity level, matrix U is reduced to size $[13, 76]$.

195 Finally, Equation 2 can be solved using a matrix multiplication operation:

$$foF2 = (F_D U) F_G, \tag{5}$$

196 which yields a $foF2$ matrix with size $[N_T, N_G]$. This may then be converted to $NmF2$
197 using Equation 1.

198 An example of the $NmF2$ output for two levels of solar activity (defined by CCIR
199 coefficients as solar minimum and solar maximum) is shown in Figure 6 for 10 UT of 15
200 April 2020. The location of the subsolar point is marked by a red circle. For solar min-
201 imum, the CCIR coefficients were derived using data from 1954-1955, and for solar max-
202 imum the years 1956-1958 were considered (Jones & Gallet, 1962, 1965; Jones et al., 1966).

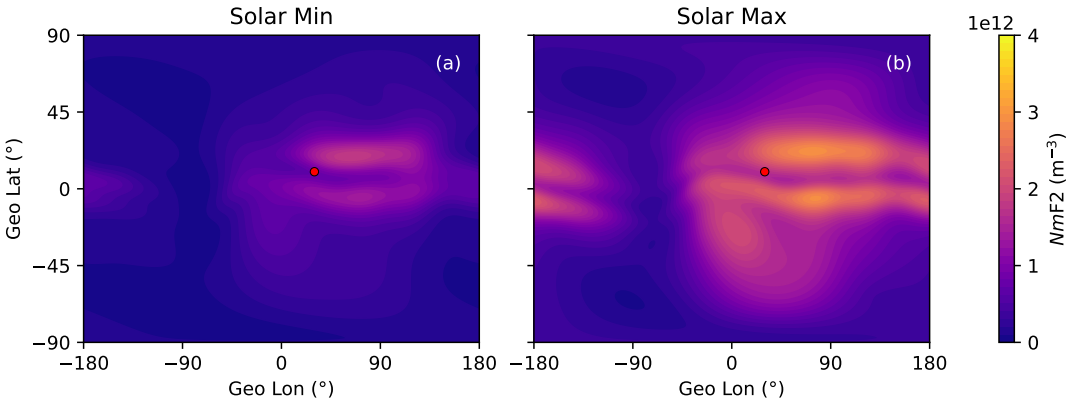


Figure 6. $NmF2$ for solar minimum (a) and solar maximum (b) for 10 UT in April 2020. Red circle shows the location of subsolar point.

203 Figure 7 shows same outputs as in Figure 6 obtained with URSI coefficients. The
204 main difference supposed to be over the oceans, as was demonstrated by Rush et al. (1989).

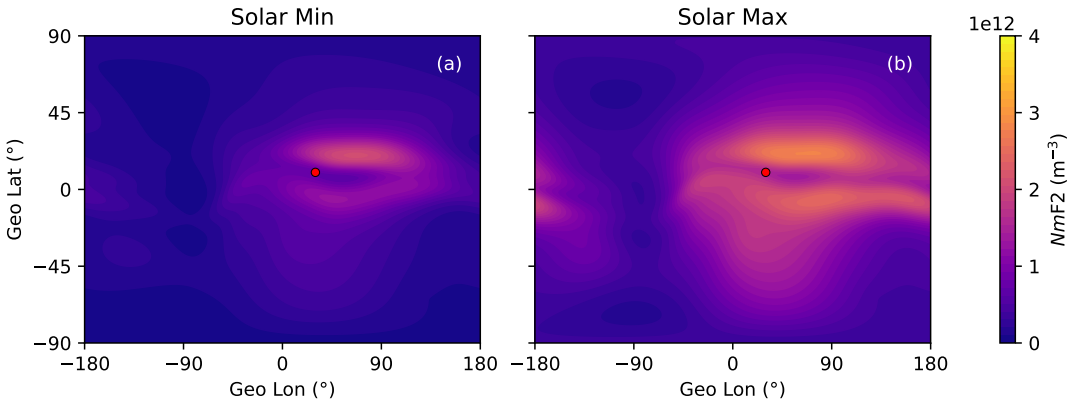


Figure 7. NmF2 for solar minimum (a) and solar maximum (b) for 10 UT in April 2020 obtained with URSI coefficients. Red circle shows the location of subsolar point.

A simplified version of the flow diagram for the PyIRI code is shown in Figure 8. In summary, the global function components F_G are calculated using the three arrays Φ, Θ, and M, and the Fourier function components F_T are calculated for a time array T. The CCIR coefficients are stored in matrix U, and the multiplication between F_D, U, and F_G gives foF2 for all the desired times and locations. Importantly, this single short operation substitutes the series of $6,272,736$ executions of the IRI FORTRAN code for the example case of a global regular grid of $1^\circ \times 1^\circ$ and 15-min resolution.

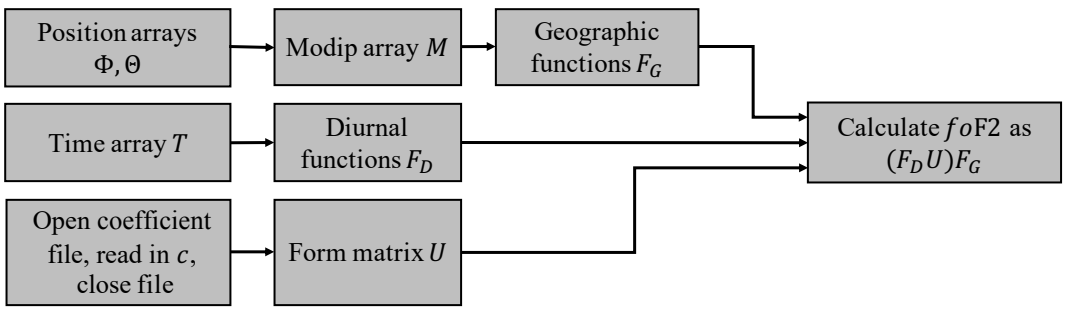


Figure 8. A simplified flow diagram for the novel method to obtain foF2 from coefficients.

2.3 Calculating hmF2

The CCIR and URSI files also contain 882 coefficients (following first 1,976 coefficients for foF2) that correspond to the propagation factor M(3000)F2, which is equal to MUF(3000)F2/foF2, where MUF(3000)F2 is the highest frequency refracted in the ionosphere that can be received at a distance of 3,000 km. The only difference between the calculation of foF2 and M(3000)F2 is in the number of the global and diurnal functions. In the case of M(3000)F2, the truncation is determined by $J = [6, 7, 5, 2, 1, 0, 0]$ and gives 49 geographic functions, whereas nine Fourier functions are used for the foF2. Therefore, the coefficient matrix has a size of $[9, 49, 2]$, unlike the previously discussed coefficient matrix. Figure 9 shows M(3000)F2 for minimum and maximum solar activity calculated from CCIR coefficients for 10 UT at April 2020.

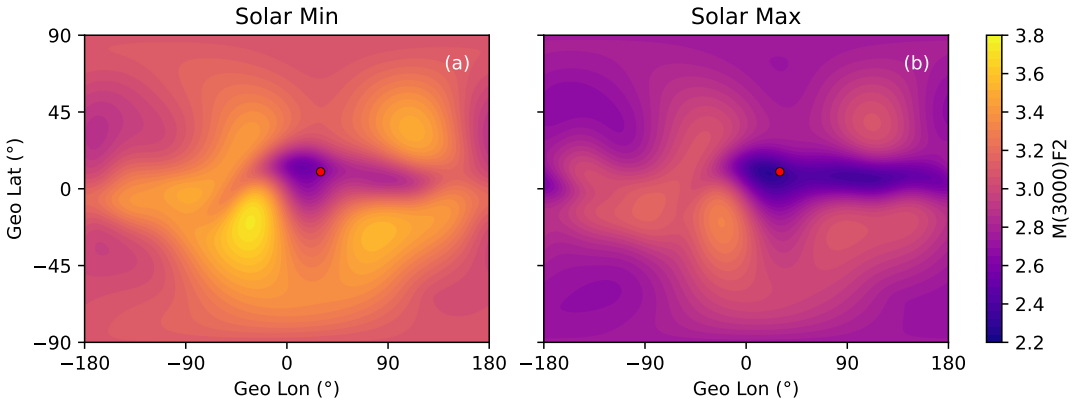


Figure 9. CCIR M(3000)F2 for solar minimum (a) and solar maximum (b) for 10 UT of April 2020. The red circle shows the location of the subsolar point.

Further, following the BSE-1979 option of IRI (Bilitza et al., 2022), developed by Bilitza et al. (1979) the hmF2 parameter is calculated from M(3000)F2 using:

$$hm\mathrm{F2} = \frac{1490}{\mathrm{M(3000)F2} + DM} - 176, \tag{6}$$

where the correction factor, DM, is:

$$DM = \frac{f_1 f_2}{\frac{fo\mathrm{F2}}{fo\mathrm{E}} - f_3} + f_4, \tag{7}$$

and the following functions, f, depend on 12-month running mean of sunspot number R_{12} and on M

$$f_1 = 0.00232 R_{12} + 0.222, \tag{8}$$

$$f_2 = 1 - \frac{R_{12}}{150}\exp\left[-\left(\frac{M}{40}\right)^2\right], \tag{9}$$

$$f_3 = 1.2 - 0.0116\exp\left(\frac{R_{12}}{41.84}\right), \tag{10}$$

$$f_4 = 0.096\frac{R_{12} - 25}{150}. \tag{11}$$

An additional limit is added to the ratio $\frac{foF2}{foE}$, where it has a firm lower limit of 1.7 to prevent the unrealistically large E layer.

After applying Equation 6, hmF2 is shown in Figure 10. Note that hmF2 is dependent on M shown in Equation 4 and through IGRF-13, which describes the secular variation of the geomagnetic field.

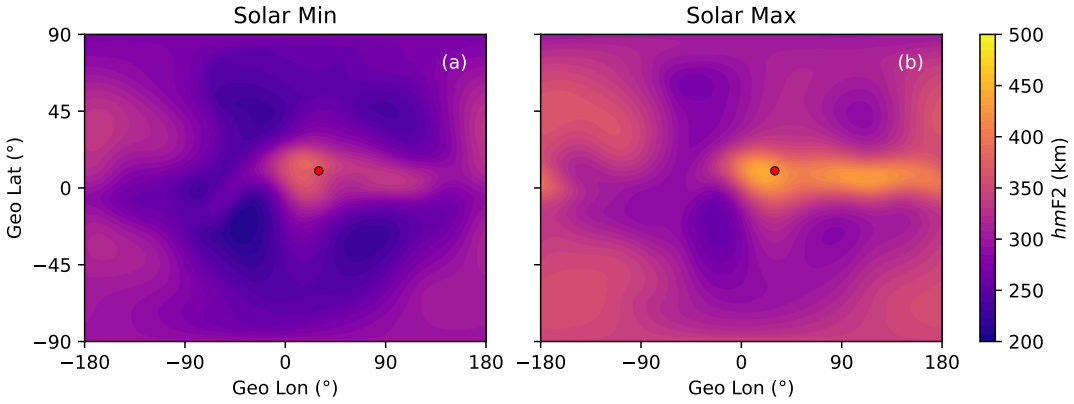


Figure 10. hmF2 for solar minimum (a) and solar maximum (b) for 10 UT of April 2020. The red circle shows the location of subsolar point.

2.4 E region calculation

The E region is the second density layer in the ionosphere, and it usually lies in the photochemical equilibrium regime. Although it can develop complex structuring, its major variation can be captured through its dependence on solar zenith angle, χ. The calculation method presented here follows the approach of the NeQuick model and is somewhat simpler than the current E region treatment in IRI. First the effective solar zenith angle, χ_{eff}, is calculated using:

$$\chi_{eff} = \frac{\chi + [90 - 0.24\exp(20 - 0.2\chi)]\exp(12(\chi - \chi_0))}{1 + \exp(12(\chi - \chi_0))}, \tag{12}$$

where χ_0 is the solar zenith angle at the solar terminator, which is set to 86.23°.

Further, a seasonal parameter s is defined as

$$s = s_0 \left(\frac{\exp(0.3\Phi) - 1}{\exp(0.3\Phi) + 1} \right), \tag{13}$$

with s_0 is defined by whether the month is near equinox or solstice:

$$s_0 = \begin{cases} -1, & \text{month} = 1, 2, 11, 12, \\ 0, & \text{month} = 3, 4, 9, 10, \\ 1, & \text{month} = 5, 6, 7, 8. \end{cases} \tag{14}$$

The critical frequency of E region is then calculated using:

$$fo\mathrm{E} = \sqrt{0.49 + (1.112 - 0.019s)^2 \sqrt{\mathrm{F10.7}} \cos^{0.6}(\chi_{eff})}, \tag{15}$$

where F10.7 is the solar radio flux at 10.7 cm.

Figure 11 shows the critical frequency of the E region for two levels of solar activity for 10 UT in April 2020. The climatology of the E region is mainly controlled by the solar ionization and therefore depends on the location of subsolar point, marked by a red circle in Figure 11.

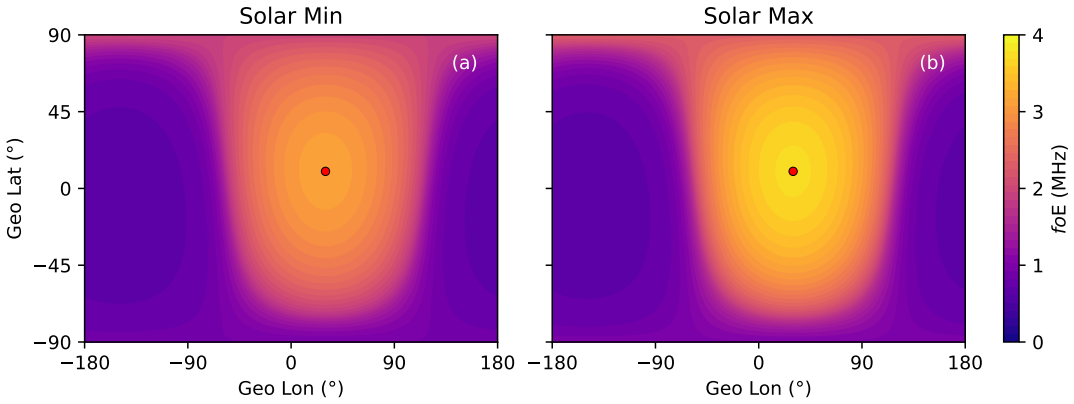


Figure 11. foE for solar minimum (a) and solar maximum (b) for 10 UT of April 2020. The red circle shows the location of the subsolar point.

3 Other dependent ionospheric parameters

This section describes other ionospheric parameters that are considered to be non-primary. These parameters can be derived from the core parameters of the IRI model that were described in Section 2.

3.1 Thicknesses of the ionospheric layers

This version of the PyIRI code has a simplified approach to the construction of the ionospheric profile, in comparison to the standard IRI source code and its options. For the thickness of bottom side of F2 region, an approach similar to NeQuick model is chosen, where the bottom side is described by the Epstein function (Bilitza, 1990), which has one parameter that describes its thickness, unlike the IRI model that uses a two parameter fit.

3.1.1 Thickness of the bottom side of the F2 layer B_{bot}^{F2}

The thickness of the F2 bottom layer B_{bot}^{F2} in PyIRI is modeled as a function of $fo\mathrm{F2}$ and M(3000)F2 using:

$$B_{bot}^{\mathrm{F2}} = \frac{47.74 \quad fo\mathrm{F2}^2}{\exp(-3.467 + 1.714\ln(fo\mathrm{F2}) + 2.02\ln(\mathrm{M(3000)F2}))}, \tag{16}$$

and is shown in Figure 12 for two levels of solar activity.

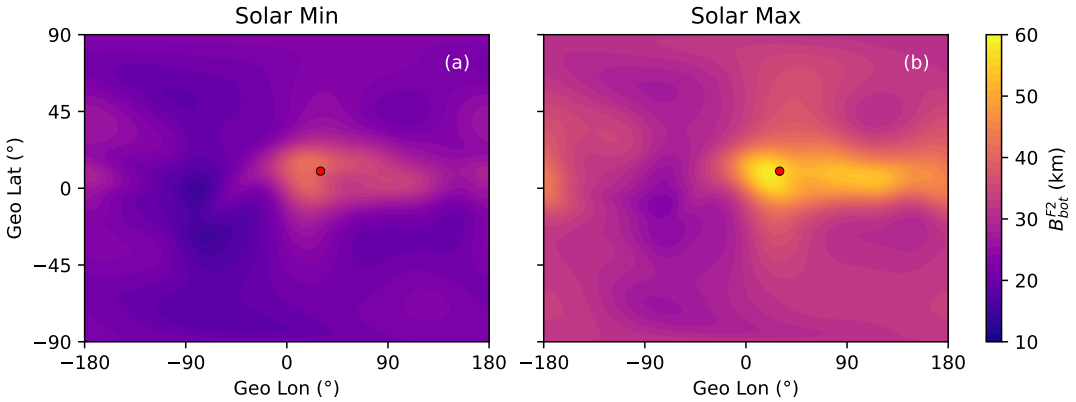


Figure 12. Thickness of the bottom side of F2 region B_{bot}^{F2} for solar minimum (a) and solar maximum (b) for 10 UT of April 2020. The red circle shows the location of the subsolar point.

3.1.2 Thickness of the top side of the F2 layer

The IRI model provides three different options to define the thickness of the topside of the F2 layer, B_{top}^{F2}, with the NeQuick approach being the standard option. However, there are slight differences in the definitions B_{top}^{F2} in NeQuick and IRI code. Here

we define B_{top}^{F2} the following way, combining the two approaches:

$$B_{top}^{\mathrm{F2}} = \frac{100x + 150}{0.041163x^2 - 0.183981x + 1.424472}, \tag{17}$$

where x depends on B_{bot}^{F2}:

$$x = \frac{kB_{bot}^{\mathrm{F2}} - 150}{100}, \tag{18}$$

and the shape parameter, k, is defined as:

$$k = 3.22 - 0.0538 foF2 - 0.00664 hmF2 + 0.113 \frac{hmF2}{B_{bot}^{\mathrm{F2}}} + 0.00257 R_{12}. \tag{19}$$

In NeQuick definition of k, the dependence on the solar activity is expressed in terms of effective ionization level, whereas in IRI the 12-month running mean of the sunspot number, R_{12}, is used. Since R_{12} is being used extensively in other parts of the PyIRI code, this approach was chosen. The B_{top}^{F2} is shown in Figure 13 for two levels of solar activity.

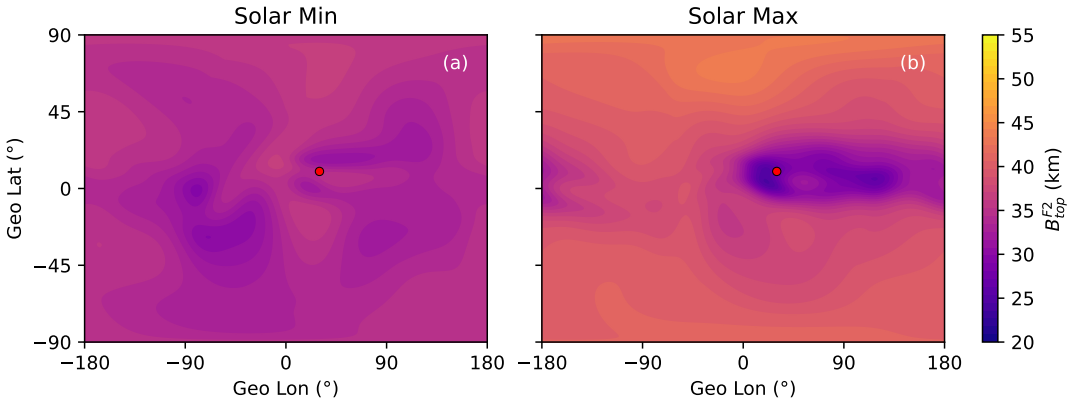


Figure 13. Thickness of the top side of F2 region B_{top}^{F2} for solar minimum (a) and solar maximum (b) for 10 UT of April 2020. The red circle shows the location of the subsolar point.

Additionally, a modification to the B_{top}^{F2} is applied as:

$$B_{modified} = B_{top}^{\mathrm{F2}} \left(1 + \frac{12.5(h - hmF2)}{100 B_{top}^{\mathrm{F2}} + 0.125(h - hmF2)} \right). \tag{20}$$

to make it height dependent. When EDPs are constructed, this $B_{modified}$ parameter is used as the thickness of the topside of F2 region, which is further discussed in Section 4.

3.1.3 Thickness of the E layer

The thickness of the bottom side of the E region B^E_{bot} is fixed at 5 km, and B^E_{top} is fixed at 7 km.

3.1.4 Thickness of the F1 layer

The thickness of the bottom side of the F1 region (that will be introduced in the following Section 3.2), B^{F1}_{bot}, is defined as:

$$B^{\mathrm{F1}}_{bot} = 0.5(hm\mathrm{F1} - hm\mathrm{E}), \tag{21}$$

and is shown in Figure 14.

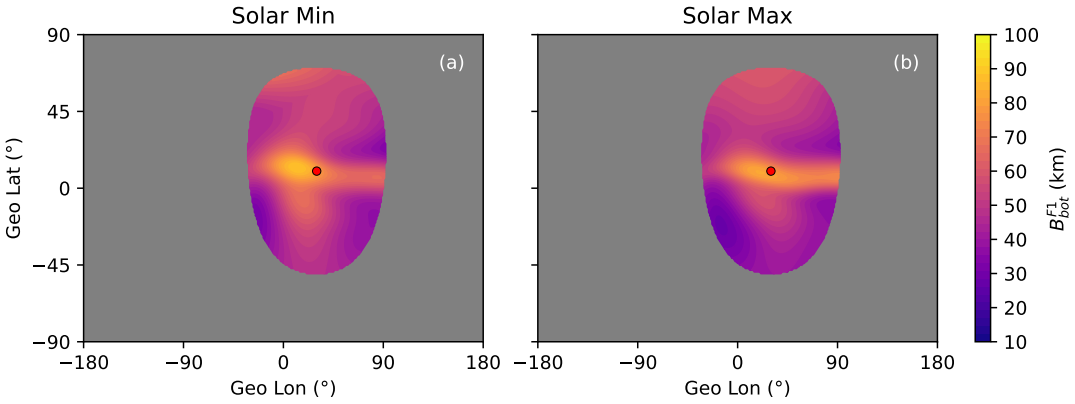


Figure 14. Thickness of the bottom side of the F1 region B^{F1}_{bot} for solar minimum (a) and solar maximum (b) for 10 UT of April 2020. The red circle shows the location of the subsolar point.

3.2 F1 region parameters

The F1 region appears only during the day time. Similar to the E region it is strongly controlled by solar illumination. The NeQuick model fully describes foF1 through foE, whereas the IRI model calculates the occurrence probability of the F1 region through χ and foF1 through magnetic dip latitude and R_{12}. Here we follow the IRI approach.

The occurrence probability function, P is given by:

$$P = (0.5 + 0.5\cos(\chi))^{2.36}\,. \tag{22}$$

This probability function does not depend on R_{12} and magnetic latitude, following the suggestion made in Bilitza et al. (2022) to simplify this parameter. The distribution of P is shown in Figure 15a, with the color bar shown on the right of the figure. Its global distribution is very similar to the NmE distribution, indicating strong solar control. This is expected, given the F1 region is dependent on photochemistry. When the occurrence probability is greater than 0.5, the critical frequency of F1 layer can be modeled as:

$$foF1 = f_s \cos^n(\chi), \tag{23}$$

where:

$$\begin{aligned} f_s &= f_0 + \tfrac{(f_{100}-f_0)R_{12}}{100}, \\ f_0 &= 4.35 + 0.0058|M'| - 0.00012M'^2, \\ f_{100} &= 5.348 + 0.011|M'| - 0.00023M'^2, \\ n &= 0.093 + 0.0046|M'| - 0.000054M'^2 + 0.0003R_{12}, \\ M' &= \arctan(\tfrac{1}{2}\tan(M)), \end{aligned} \tag{24}$$

where M' is the magnetic dip latitude, which can be calculated from M. Figures 15b and 15c show the $foF1$ during solar minimum and solar maximum, respectively, for 10 UT of April 2020.

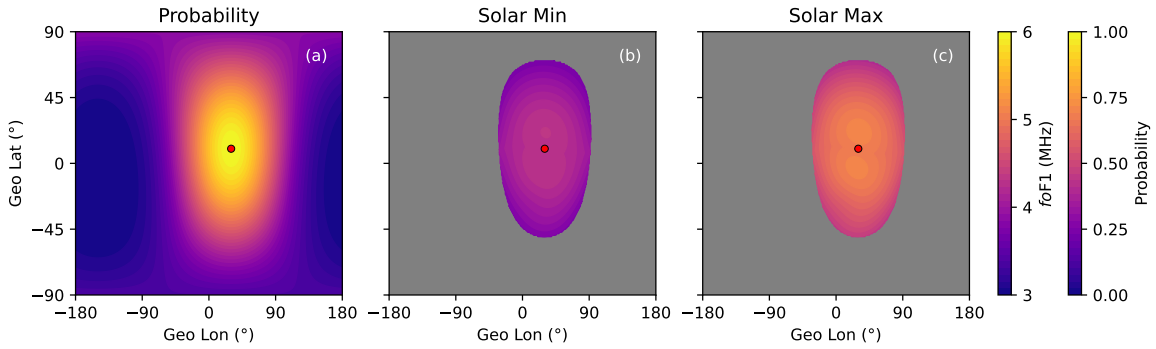


Figure 15. Occurrence probability function P for F1 region (a), with critical frequency $foF1$ during solar minimum (b) and solar maximum (c) for 10 UT of April 2020. The red circle shows the location of the subsolar point.

The height of the F1 layer $hmF1$ is found where the bottom side of F2 layer drops to the value of $NmF1$, in cases when F1 layer is present. This height is found analyt-

ically using the following expression derived from the Epstein function:

$$hm\mathrm{F1} = B_{bot}^{\mathrm{F2}} \log\left(-\left(1-\frac{2Nm\mathrm{F2}}{Nm\mathrm{F1}}\right)-\sqrt{\left(1-\frac{2Nm\mathrm{F2}}{Nm\mathrm{F1}}\right)^2-1}\right)+hm\mathrm{F2}. \qquad (25)$$

The hmF1 map is shown in Figure 16.

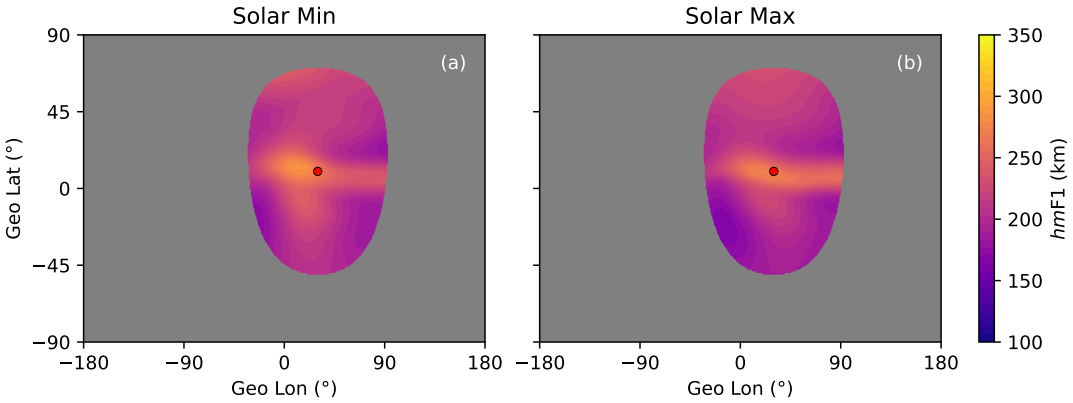


Figure 16. F1 region peak height hmF1 during solar minimum (a) and solar maximum (b) for 10 UT of April 2020. The red circle shows the subsolar point location.

3.3 Sporadic E layers

One of the important deviations from photochemical equilibrium in the E region is the formation of very thin sporadic E layers, Es. PyIRI includes a monthly mean specification of Es, using the Leftin et al. (1968) coefficients. They are provided in the same format as the CCIR coefficients, but with different truncation points for the higher degrees of the latitudinal expansion: $J = [10, 12, 6, 2, 0, 0]$. Figure 17 shows the critical frequency of Es for both levels of solar activity.

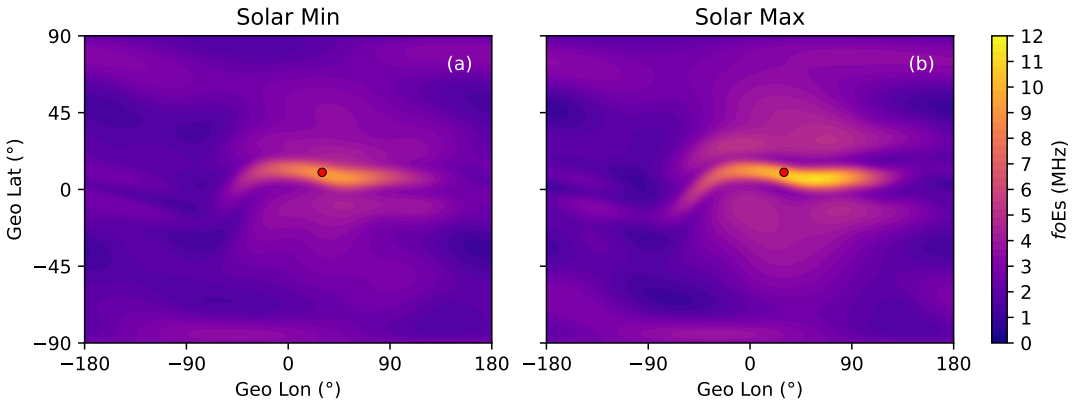


Figure 17. Critical frequency of Es for solar minimum (a) and solar maximum (b) for 10 UT of April 2020. The circle shows the subsolar point location.

The height of the Es is fixed at 100 km. The topside and bottomside thicknesses are fixed at $B_{top}^{Es} = B_{bot}^{Es} = 1$ km.

The Es parameters are provided as additional outputs of PyIRI model and are not included in the construction of the vertical EDP. This is due to the fact that a very high vertical resolution is required to be able to include this thin ionospheric layer. The next Section 4 describes the process of the EDP construction.

4 PyIRI construction of the 3-D ionosphere

This section explains how the 3-D ionosphere is constructed from the ionospheric parameters defined in Sections 2 to 3.2. This approach was specifically developed for PyIRI, with computational efficiency in mind.

First, consider a 1-D example of EDP construction from the ionospheric coefficients. The coefficients were chosen for $\phi = 0°$N and $\theta = 100°$E, when the F1 region is not present, and for solar maximum conditions. Figure 18a shows the $NmF2$ and NmE parameters with red and purple circles, located at the $hmF2$ and hmE heights, respectively.

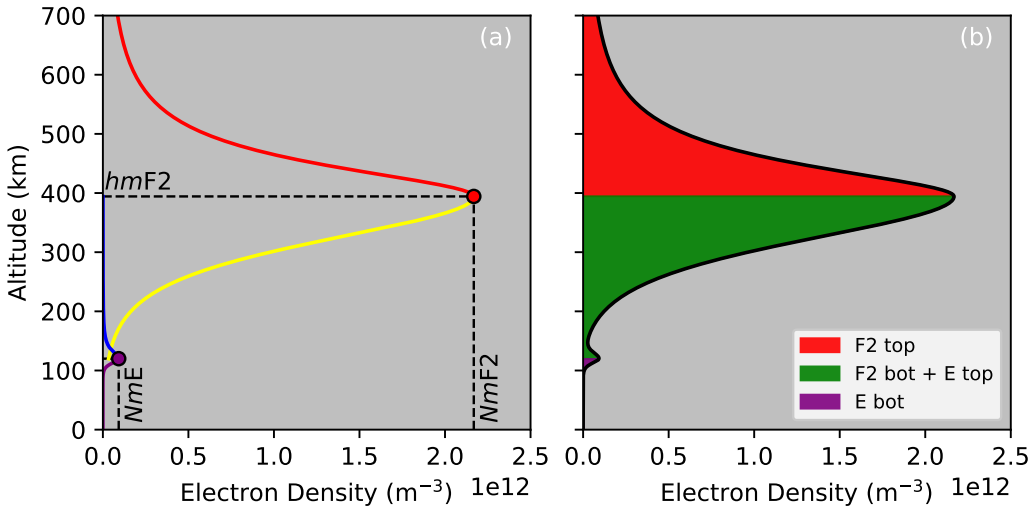


Figure 18. Construction of the EDP without F1 region.

The bottom side of the F2 region, the topside of the E region, and the bottom side of the E region, shown with yellow, blue, and purple colors, respectively, in Figure 18a are constructed using Epstein function:

$$Ne = 4Nm \frac{e^{\frac{h-hm}{B}}}{\left(1 - e^{\frac{h-hm}{B}}\right)^2}, \tag{26}$$

using corresponding peak densities (Nm), peak heights (hm), and thicknesses (B). The bottomside of the F2 region is evaluated from hmF2 down to hmE. The topside of the E region is evaluated from hmE up to hmF2. The topside of the F2 region, shown with red in Figure 18, is calculated using Equation 26 with the modified thickness of the profile defined in Equation 20.

Special care needs to be taken for the region between hmF2 and hmE to add two curves together. Unlike in IRI and NeQuick, a drop function was introduced to model the transition of the E region to the F2 region, without any re-scaling of the peaks. A drop function of the following form was implemented using:

$$Y_{up} = 1 - \left(\frac{h - hm\mathrm{E}}{hm\mathrm{F2} - hm\mathrm{E}}\right)^4. \tag{27}$$

Prior to the summation, the topside E region is multiplied by Y_{up} and the bottom side of $F2$ is multiplied by:

$$Y_{down} = 1 - \left(\frac{hm\mathrm{F2} - h}{hm\mathrm{F2} - hm\mathrm{E}}\right)^4. \tag{28}$$

Since the contribution of the E region is minimal above 150 km, the shape of the bottom side F2 region remains unchanged. However, the influence of the F2 region on the topside of E region will be reduced. Figure 18b shows the final profile with different regions indicated by the color.

When the F1 region is present, like it is at the location $\phi = 0°$N and $\theta = 0°$E, the bottomside F1 region is modeled using an Epstein function and the bottomside of the F2 region is not extended further than hmF1. The same drop function is used to reduce the influence of the F1 region on the topside of the E region and the influence of the topside E region on the bottom of the F1 region prior to their summation. Figure 19a shows the NmF1 with an orange circle and the bottom of the F1 region with a yellow curve. As can be seen in Figure 19b, the drop function approach works well to merge the two regions without changing the shapes of the individual regions and without any re-scaling of the peaks.

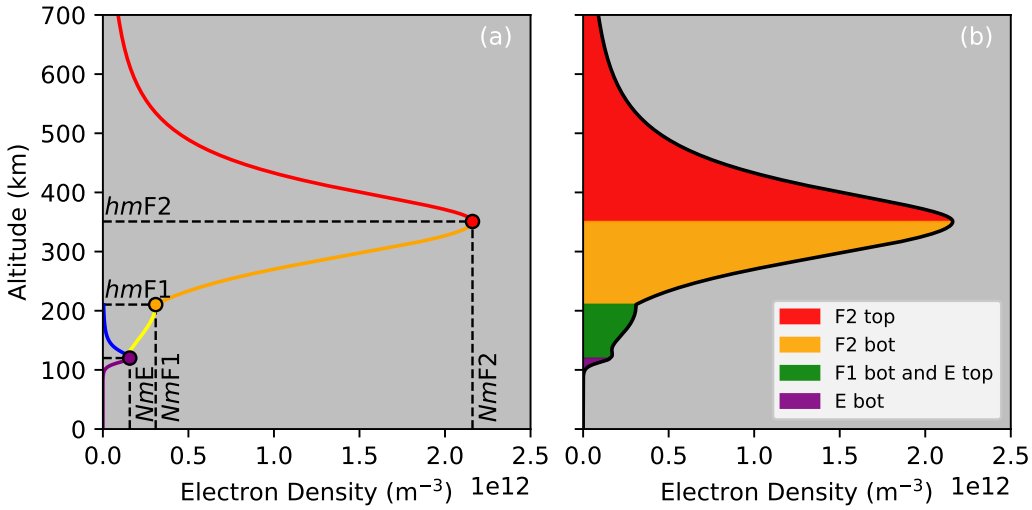


Figure 19. Construction of the EDP with F1 region.

It is important to mention that even though the computation time is dramatically reduced when constructing the maps of NmF2 and hmF2 from the CCIR coefficients and deriving all the other parameters in matrix form, it is still computationally expensive to build EDPs in the traditional way, by constructing each horizontal position individually.

The serial computational approach can be avoided with the help of the NumPy `where` function. This function can not only have multidimensional arguments, but also multidimensional conditions. This makes it possible to apply conditional EDP functions to matrix inputs.

Figure 20 helps to visualize this 2-D selection, where red, orange, and purple surfaces show hmF2, hmF1, and hmE, respectively. These surfaces represent the boundaries for the selection, similar to the 1-D example with red, orange and purple circles in Figure 19a. For example, to construct the topside of the ionosphere simultaneously for the entire global grid, all 3-D grid points that are located above the red surface need to be selected and passed to the Epstein function. Similarly, to construct the bottom side of the F2 region, when the F1 region is present, all the points between the green and red surfaces should be selected. The trick to make a 2-D selection is to present all parameters as 2-D matrices, by populating the same information at all heights. For example, a matrix for hmF2 will have dimensions $[N_G, N_V]$, where N_V is the number of desired vertical grid cells that correspond to an array of altitudes h. This matrix will have same elements at all N_V. The same should be done for all other parameters. However, the height matrix H should have $[N_V, N_G]$ elements, with the heights being equal at all horizontal locations. Further, the output of `numpy.where`$(H \geq hm\text{F2})$ gives 2-D indexes for the location of all the grid cells that correspond to the ionospheric top side of F2 region IND^{F2}_{top}. Finally, $Nm\text{F2}[IND^{\text{F2}}_{top}]$, $hm\text{F2}[IND^{\text{F2}}_{top}]$, $B^{\text{F2}}_{top}[IND^{\text{F2}}_{top}]$, and $H[IND^{\text{F2}}_{top}]$ can be given to the topside Epstein function using Equations 26 and 20 to perform the calculation of the electron density for the top side of the F2 region. The result will have the same shape $[N_G, N_V]$ as the input. It is not necessary to introduce a separate dimension for time with size N_T. Instead, all the matrices can have dimensions of $[N_G \times N_T, N_V]$. The final outputs can then be reshaped to the more intuitive dimensions of $[N_T, N_G, N_V]$.

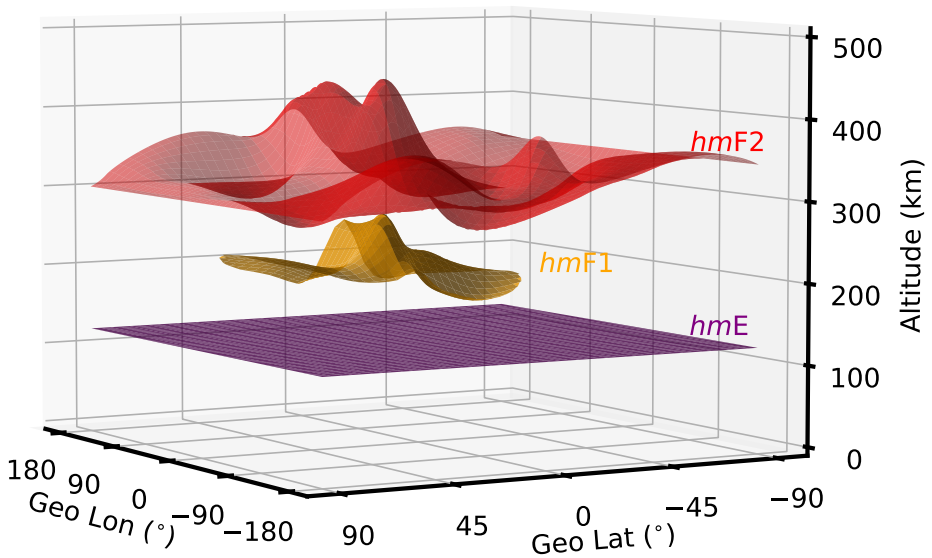


Figure 20. Surfaces of *hm*F2, *hm*F1, and *hm*E parameters that represent the boundaries for the 2-D selection of the 3-D grid points.

5 Solar and seasonal interpolation

As mentioned throughout, all the ionospheric parameters were determined for two solar reference points: solar minimum and solar maximum. Additionally, annual variations have been determined using monthly mean values. Therefore, to estimate electron density for a particular day of the year, and solar activity level, interpolation of the parameters is necessary.

To find the ionospheric parameters (referred to as X) for a particular day of the month, the mean parameters are found for the two consequent months around the day of interest. In case the day of interest is less than the 15^{th} of the month, the month before X_1 and the current month X_2 will be taken as a reference points. In case the day of interest is greater than the 15^{th}, the current month X_1 and the following month X_2

399 are considered. The following expression is used for the interpolation

$$X = X_1\delta_2 + X_2\delta_1, \tag{29}$$

400 where δ_1 is a ratio of number of days between the day of interest and the middle of the
401 month that corresponds to X_2, whereas δ_2 is the ratio of number of days between the
402 day of interest and the middle of the month that corresponds to X_1. This is how a smooth
403 transition is obtained.

404 In the original studies of Jones and Gallet (1962, 1965), the CCIR coefficients were
405 determined for years 1954-1955 representing solar minimum conditions with R_{12}=0 and
406 1956–1958 for the solar maximum with R_{12}=100. Further research found that better re-
407 sults can be achieved if these reference points are expressed in terms of the 12-month
408 running mean of the Ionosonde Global (IG) index, IG_{12} (Liu et al., 1983; Liu & Chen,
409 2009; Ma et al., 2009). The same coefficients for the lower reference level were assigned
410 to $IG_{12}^{min} = 0$ and for the upper limit to $IG_{12}^{max} = 100$, representing minimum and
411 maximum solar condition levels, respectively.

412 PyIRI uses F10.7 as a solar driver, but the interpolation between solar minimum
413 and solar maximum is determined based on the IG_{12} coefficient. The following quadratic
414 equations are used for the conversion between F10.7 and IG_{12}:

$$F10.7 = 0.00089R_{12}^2 + 0.728R_{12} + 63.7. \tag{30}$$

415

$$IG_{12} = -0.00268R_{12}^2 + 1.468R_{12} + 12.349, \tag{31}$$

416 Linear interpolation for all ionospheric parameters may be obtained in matrix form
417 using:

$$X = \frac{X_{min}(IG_{12}^{max} - IG_{12}) + X_{max}(IG_{12} - IG_{12}^{min})}{IG_{12}^{max} - IG_{12}^{min}}, \tag{32}$$

418 where IG_{12} is derived from F10.7 using Equations 31 and 30.

419 6 PyIRI and IRI-2020 Comparison

420 This section presents a comparison of PyIRI parameters to the parameters obtained
421 with IRI 2020. For this purpose, IRI was run online at the CCMC Instant Run System.
422 The following initial conditions were chosen:

423 • Version: IRI 2020

- Date Time: February 11, 2012
- Latitude: 10°
- Longitude: 110°
- Profile Type: Hour of Day
- Data Options: List of peak height and densities

Figure 21a shows NmF2, NmF1, and NmE from this run with red, green, and yellow circles, respectively. PyIRI with URSI coefficients was also executed for the same initial conditions, and the results are shown in Figure 21a with red, green, and yellow curves. The only major difference between the two models is seen for the NmF1 parameter, where the PyIRI shows longer presence of F1 region, based on the probability of F1 region occurrence, and the IRI model additionally restricts the occurrence based on the critical solar zenith angle.

A comparison among peak heights is shown in Figure 21b. hmF2 values are different between IRI and PyIRI, because PyIRI uses CCIR coefficients to derive hmF2, whereas the IRI uses SHU-2015 option as a default (Bilitza et al., 2022). The hmF1 values differ between the two models because this parameter directly depends on hmF2.

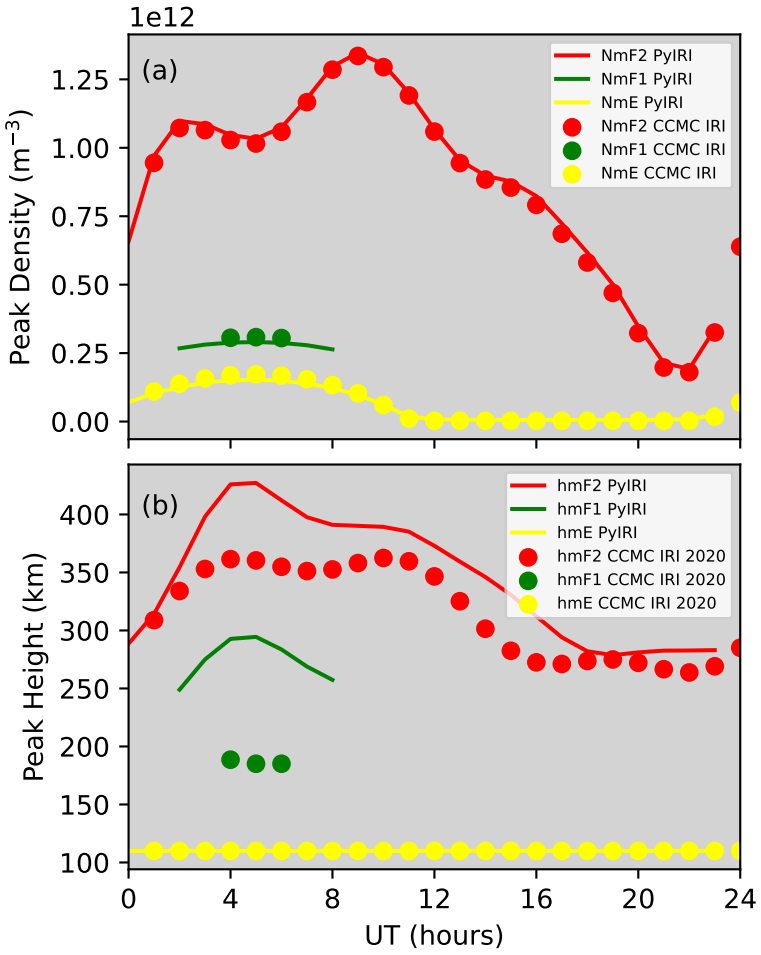


Figure 21. Comparison of the parameters obtained with PyIRI and IRI 2020

7 PyIRI Tool

PyIRI (Forsythe & Burrell, 2023) can be downloaded using the following command in a terminal window

```
pip install pyiri
```

Any Python3 environment can be used to run PyIRI. The package has three main submodules:

```
import PyIRI
import PyIRI.main_library as ml
import PyIRI.plotting as plot
```

To obtain the ionospheric parameters and the electron density for a particular day, the following command in Python can be used:

```
f2, f1, e_peak, es_peak, sun, mag, edp = ml.IRI_density_1day(year, month,
day, ahr, alon, alat, aalt, f107, PyIRI.coeff_dir, ccir_or_ursi)
```

where the inputs `year,month,day` are integers, and `ahr,alon,alat,aalt` are 1-D NumPy arrays with sizes $[N_T], [N_G], [N_G], [N_V]$, and units of (hours), (°), (°), and (km), respectively. They can be regularly or irregularly spaced. `f107` provides the user-determined solar driver. `ccir_or_ursi` allows user to chose between CCIR (if 0) or URSI coefficients (if 1) for $NmF2$ calculation.

The outputs `f2,f1,e_peak,es_peak` are the dictionaries with the ionospheric parameters `Nm`, `hm`, `fo`, `B_top`, and `B_bot`. `f2` dictionary also includes M3000 parameter, and `f1` dictionary also includes probability density `P` for the occurrence of F1 region. All parameters have size $[N_T, N_G]$. `edp` is the electron density of size $[N_T, N_V, N_G]$ expressed in (m^{-3}). The `sun` dictionary contains `lon` and `lat` for subsolar point location in (°), with size $[N_T]$. The `mag` dictionary includes magnetic inclination `inc` in (°), modified dip angle `modip` in (°), and magnetic dip latitude `mag_dip_lat` in (°), all of size $[N_G]$.

In case one is interested in monthly mean parameters for minimum and maximum levels of solar activity, the following command can be used:

```
f2, f1, e_peak, es_peak, sun, mag = ml.IRI_monthly_mean_par(year, month,
ahr, alon, alat, PyIRI.coeff_dir, ccir_or_ursi)
```

where the output arrays in the dictionaries `f2,f1,e_peak,es_peak` will have size $[N_T, N_G, 2]$ (with the last dimension indicating the two levels of solar activity). The monthly mean electron density can be constructed using:

```
edens_prof = ml.reconstruct_density_from_parameters(f2, f1, e_peak, aalt)
```

where the output has shape $[2, N_T, N_V, N_G]$. An example of how PyIRI can be used are located at `https://pyiri.readthedocs.io/en/latest/examples.html`.

8 Conclusion

This paper presented a novel approach for empirical modeling of the ionosphere that allows the evaluation of the model parameters simultaneously on the entire global grid and for the entire diurnal time array using well-known and validated CCIR and URSI coefficients. The derivation of other ionospheric parameters that depend on the coefficient maps were described. It presents an approach for modernizing legacy and empirical models to take advantage of current matrix programming frameworks. Finally, examples of how the open-source Python tool, PyIRI, can be used were provided. Future development of PyIRI will aim to meet community needs by expanding efficient computational support for additional IRI outputs. Community contributions are welcome.

9 Open Research

PyIRI software is made available to the community at GitHub (Forsythe & Burrell, 2023).

Acknowledgments

This work is sponsored by the Office of Naval Research. The IRI homepage at `http://irimodel.org` provides open access to the FORTRAN model code of all major version of the model, to online computations of IRI parameters, and to information about IRI members, workshops, and publications. The online IRI run was completed using CCMC Instant Run System at `https://kauai.ccmc.gsfc.nasa.gov/instantrun/iri/`. I am grateful for help from Douglas P. Drob, who helped finding sporadic E coefficients.

References

Alken, P., Thébault, E., Beggan, C. D., & et al. (2021). International Geomagnetic Reference Field: the thirteenth generation. *Earth, Planets and Space*, *73*(49). doi: 10.1186/s40623-020-01288-x

Bilitza, D. (1990). International Reference Ionosphere 1990. National Space Science Data Center. *National Space Science Data Center*, *Report 90-22*.

Bilitza, D., Pezzopane, M., Truhlik, V., Altadill, D., Reinisch, B. W., & Pignalberi, A. (2022). The International Reference Ionosphere model: a review and description of an ionospheric benchmark. *Rev. Geophys.*, *60*(4). doi: 10.1029/2022RG000792

Bilitza, D., Sheik, N., & Eyfrig, R. (1979). A global model for the height of the F2-peak using M3000 values from the CCIR numerical map. *ITU Telecommunication Journal*, *46*, 549-553.

Forsythe, V. V., & Burrell, A. G. (2023). victoriyaforsythe/PyIRI: v0.0.1. *GitHub*. doi: 10.5281/zenodo.8235173

Ilma, R. (2017, January). rilma/pyIRI2016: Official release of the IRI2016 wrapper in Python. *GitHub*. Retrieved from `https://doi.org/10.5281/zenodo.240895` doi: 10.5281/zenodo.240895

Jones, W. B., & Gallet, R. M. (1962). Representation of diurnal and geographical variations of ionospheric data by numerical methods. *Journal of Research of the National Bureau of Standards*, *66*(4), 129–147. doi: 10.6028/jres.066D.043

Jones, W. B., & Gallet, R. M. (1965). Representation of diurnal and geographic variations of ionospheric data by numerical methods, II. Control of instability. *ITU Telecommunication Journal*, *32*(1), 18–28.

Jones, W. B., Graham, R. P., & Leftin, M. (1966). Advances in ionospheric mapping by numerical methods. *ESSA Technical Report ERL 107-ITS75, U.S. Department of Commerce*.

Leftin, M., Ostrow, S. M., & Preston, C. (1968). Numerical Maps of foEs for Solar Cycle Minimum and Maximum. *ESSA Technical Report*, *ERL 73-ITS 63*.

Liu, L., & Chen, Y. (2009). Statistical analysis of solar activity variations of total electron content derived at Jet Propulsion Laboratory from GPS observations. *J. Geophys. Res.*, *114*. doi: 10.1029/2009JA014533

Liu, R. Y., Smith, P. A., & King, J. W. (1983). A new solar index which leads to improved fof2 predictions using the ccir atlas. *Telecommunication Journal*, *50(8)*, 408-414.

Ma, R., Xu, J., Wang, W., & Yuan, W. (2009). Seasonal and latitudinal differences of the saturation effect between ionospheric NmF2 and solar activity indices. *J. Geophys. Res.*, *114(A10)*.

Nava, B., Coïsson, P., & Radicella, S. (2008). A new version of the NeQuick ionosphere electron density model. *J. Atmos. Sol. Terr. Phys.*, *70*(15), 1856-1862. doi: 10.1016/j.jastp.2008.01.015

Rush, M. C., Fox, M., Bilitza, D., Davies, K., McNamara, L., Stewart, F. G., &

Pokempner, M. (1989). Ionospheric mapping - an update of fof2 coefficients. *Telecommunication Journal*, *56*, 179-182.

Figure 1.

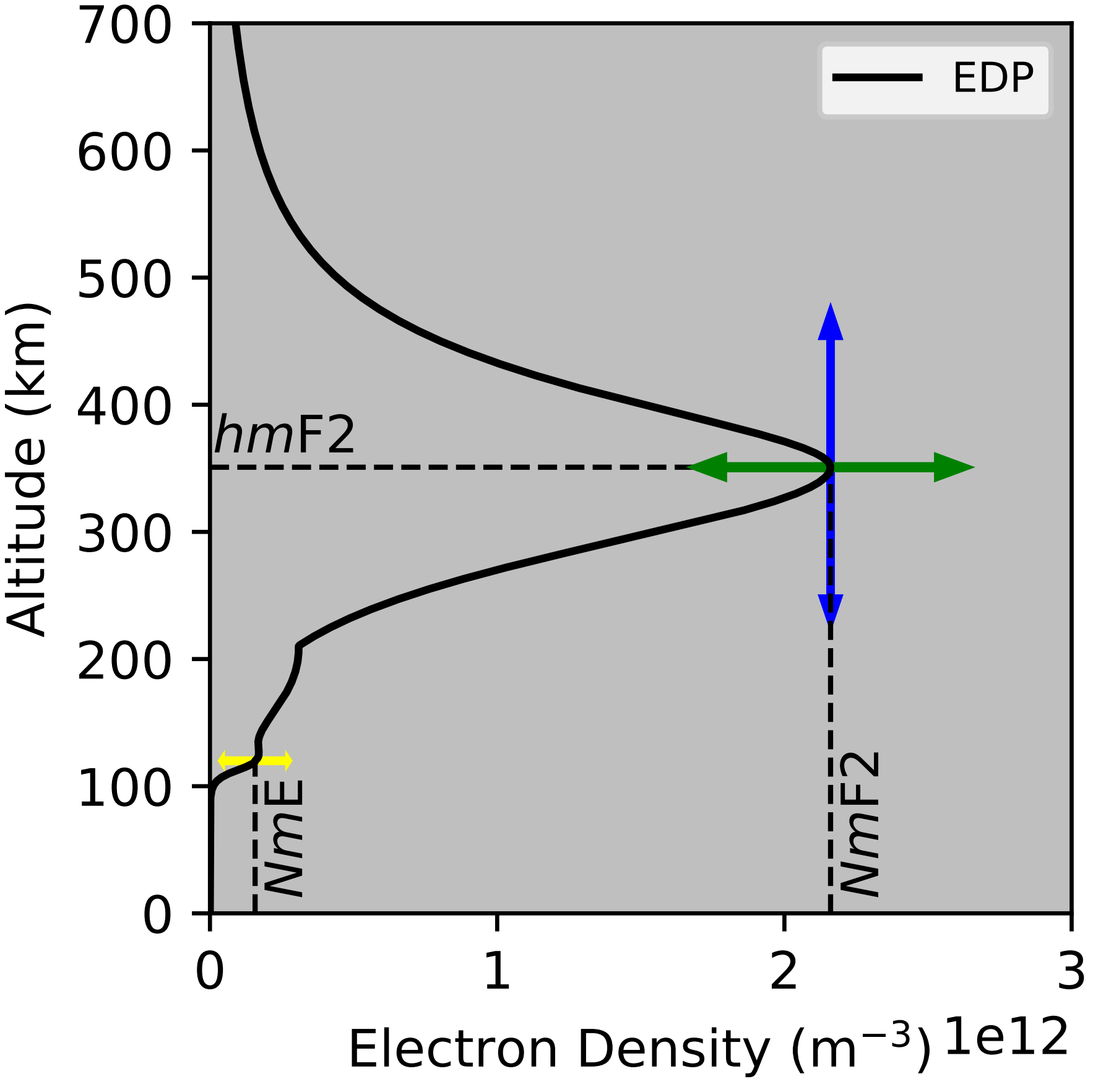

EDP
hmF2
NmE
NmF2
Altitude (km)
Electron Density (m⁻³)
1e12
0
100
200
300
400
500
600
700
0
1
2
3

Figure 2.

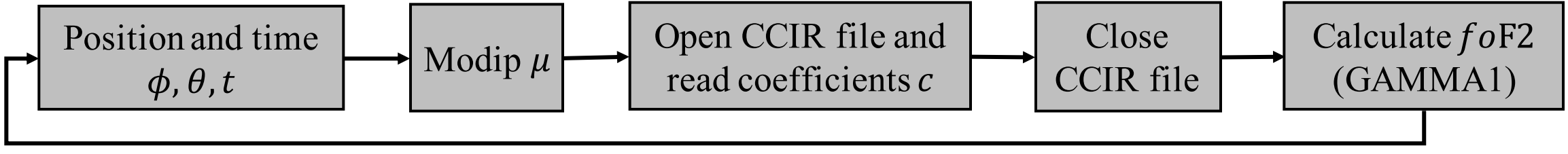

Position and time
ϕ, θ, t
Modip μ
Open CCIR file and
read coefficients c
Close
CCIR file
Calculate foF2
(GAMMA1)

Figure 3.

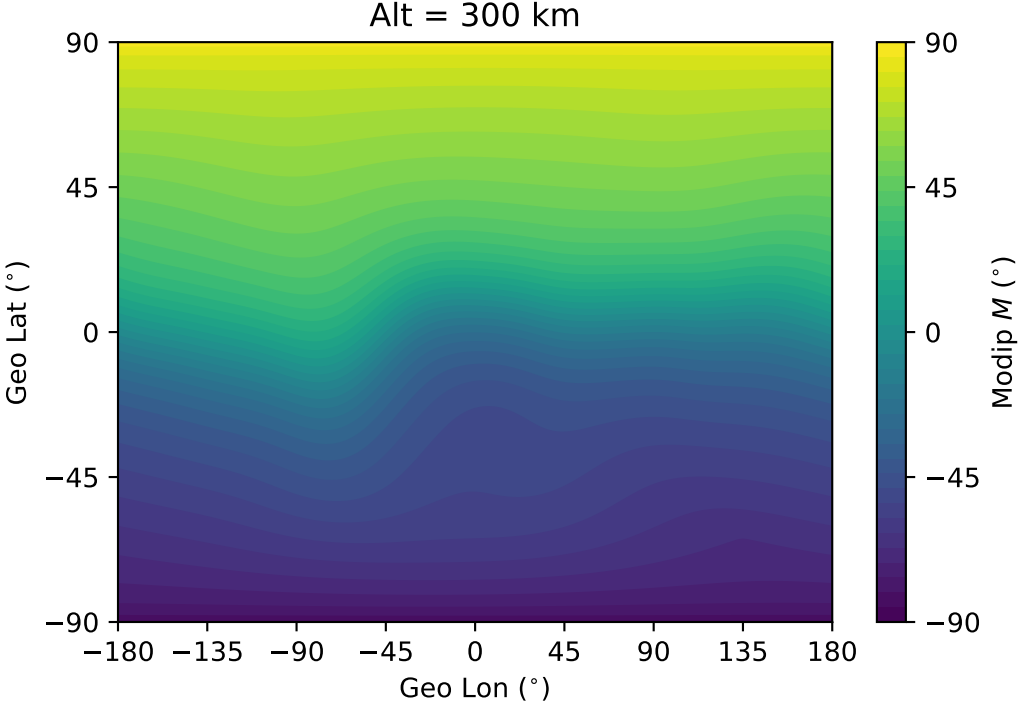

Alt = 300 km
90
45
0
−45
−90
Geo Lat (°)
−180
−135
−90
−45
0
45
90
135
180
Geo Lon (°)
90
45
0
−45
−90
Modip M (°)

Figure 4.

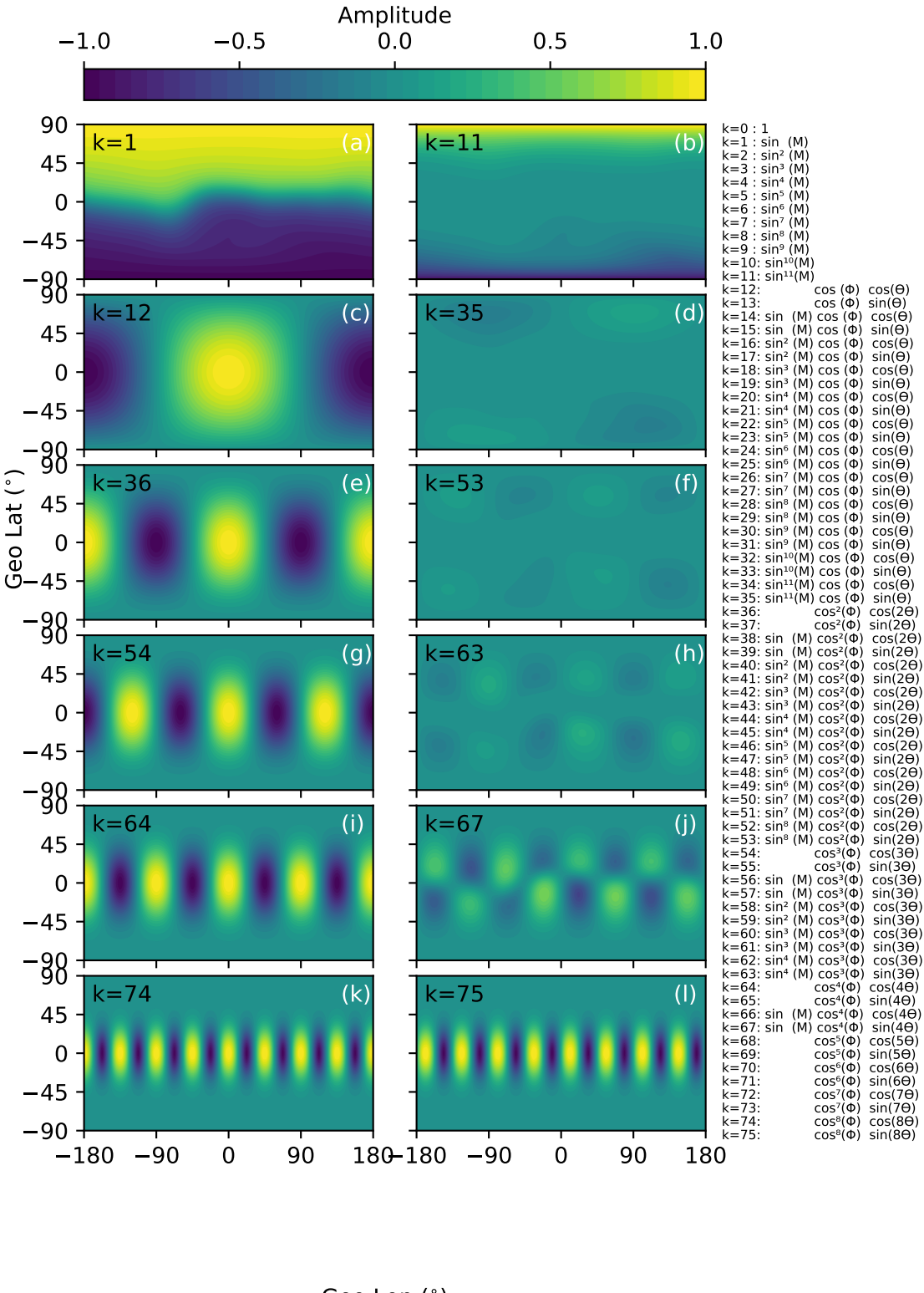

Amplitude
−1.0
−0.5
0.0
0.5
1.0
k=1 (a)
k=11 (b)
k=12 (c)
k=35 (d)
k=36 (e)
k=53 (f)
k=54 (g)
k=63 (h)
k=64 (i)
k=67 (j)
k=74 (k)
k=75 (l)
90
45
0
−45
−90
−180
−90
0
90
180
Geo Lat (°)
Geo Lon (°)
k=0 : 1
k=1 : sin (M)
k=2 : sin² (M)
k=3 : sin³ (M)
k=4 : sin⁴ (M)
k=5 : sin⁵ (M)
k=6 : sin⁶ (M)
k=7 : sin⁷ (M)
k=8 : sin⁸ (M)
k=9 : sin⁹ (M)
k=10: sin¹⁰(M)
k=11: sin¹¹(M)
k=12: cos (Φ) cos(Θ)
k=13: cos (Φ) sin(Θ)
k=14: sin (M) cos (Φ) cos(Θ)
k=15: sin (M) cos (Φ) sin(Θ)
k=16: sin² (M) cos (Φ) cos(Θ)
k=17: sin² (M) cos (Φ) sin(Θ)
k=18: sin³ (M) cos (Φ) cos(Θ)
k=19: sin³ (M) cos (Φ) sin(Θ)
k=20: sin⁴ (M) cos (Φ) cos(Θ)
k=21: sin⁴ (M) cos (Φ) sin(Θ)
k=22: sin⁵ (M) cos (Φ) cos(Θ)
k=23: sin⁵ (M) cos (Φ) sin(Θ)
k=24: sin⁶ (M) cos (Φ) cos(Θ)
k=25: sin⁶ (M) cos (Φ) sin(Θ)
k=26: sin⁷ (M) cos (Φ) cos(Θ)
k=27: sin⁷ (M) cos (Φ) sin(Θ)
k=28: sin⁸ (M) cos (Φ) cos(Θ)
k=29: sin⁸ (M) cos (Φ) sin(Θ)
k=30: sin⁹ (M) cos (Φ) cos(Θ)
k=31: sin⁹ (M) cos (Φ) sin(Θ)
k=32: sin¹⁰(M) cos (Φ) cos(Θ)
k=33: sin¹⁰(M) cos (Φ) sin(Θ)
k=34: sin¹¹(M) cos (Φ) cos(Θ)
k=35: sin¹¹(M) cos (Φ) sin(Θ)
k=36: cos²(Φ) cos(2Θ)
k=37: cos²(Φ) sin(2Θ)
k=38: sin (M) cos²(Φ) cos(2Θ)
k=39: sin (M) cos²(Φ) sin(2Θ)
k=40: sin² (M) cos²(Φ) cos(2Θ)
k=41: sin² (M) cos²(Φ) sin(2Θ)
k=42: sin³ (M) cos²(Φ) cos(2Θ)
k=43: sin³ (M) cos²(Φ) sin(2Θ)
k=44: sin⁴ (M) cos²(Φ) cos(2Θ)
k=45: sin⁴ (M) cos²(Φ) sin(2Θ)
k=46: sin⁵ (M) cos²(Φ) cos(2Θ)
k=47: sin⁵ (M) cos²(Φ) sin(2Θ)
k=48: sin⁶ (M) cos²(Φ) cos(2Θ)
k=49: sin⁶ (M) cos²(Φ) sin(2Θ)
k=50: sin⁷ (M) cos²(Φ) cos(2Θ)
k=51: sin⁷ (M) cos²(Φ) sin(2Θ)
k=52: sin⁸ (M) cos²(Φ) cos(2Θ)
k=53: sin⁸ (M) cos²(Φ) sin(2Θ)
k=54: cos³(Φ) cos(3Θ)
k=55: cos³(Φ) sin(3Θ)
k=56: sin (M) cos³(Φ) cos(3Θ)
k=57: sin (M) cos³(Φ) sin(3Θ)
k=58: sin² (M) cos³(Φ) cos(3Θ)
k=59: sin² (M) cos³(Φ) sin(3Θ)
k=60: sin³ (M) cos³(Φ) cos(3Θ)
k=61: sin³ (M) cos³(Φ) sin(3Θ)
k=62: sin⁴ (M) cos³(Φ) cos(3Θ)
k=63: sin⁴ (M) cos³(Φ) sin(3Θ)
k=64: cos⁴(Φ) cos(4Θ)
k=65: cos⁴(Φ) sin(4Θ)
k=66: sin (M) cos⁴(Φ) cos(4Θ)
k=67: sin (M) cos⁴(Φ) sin(4Θ)
k=68: cos⁵(Φ) cos(5Θ)
k=69: cos⁵(Φ) sin(5Θ)
k=70: cos⁶(Φ) cos(6Θ)
k=71: cos⁶(Φ) sin(6Θ)
k=72: cos⁷(Φ) cos(7Θ)
k=73: cos⁷(Φ) sin(7Θ)
k=74: cos⁸(Φ) cos(8Θ)
k=75: cos⁸(Φ) sin(8Θ)

Figure 5.

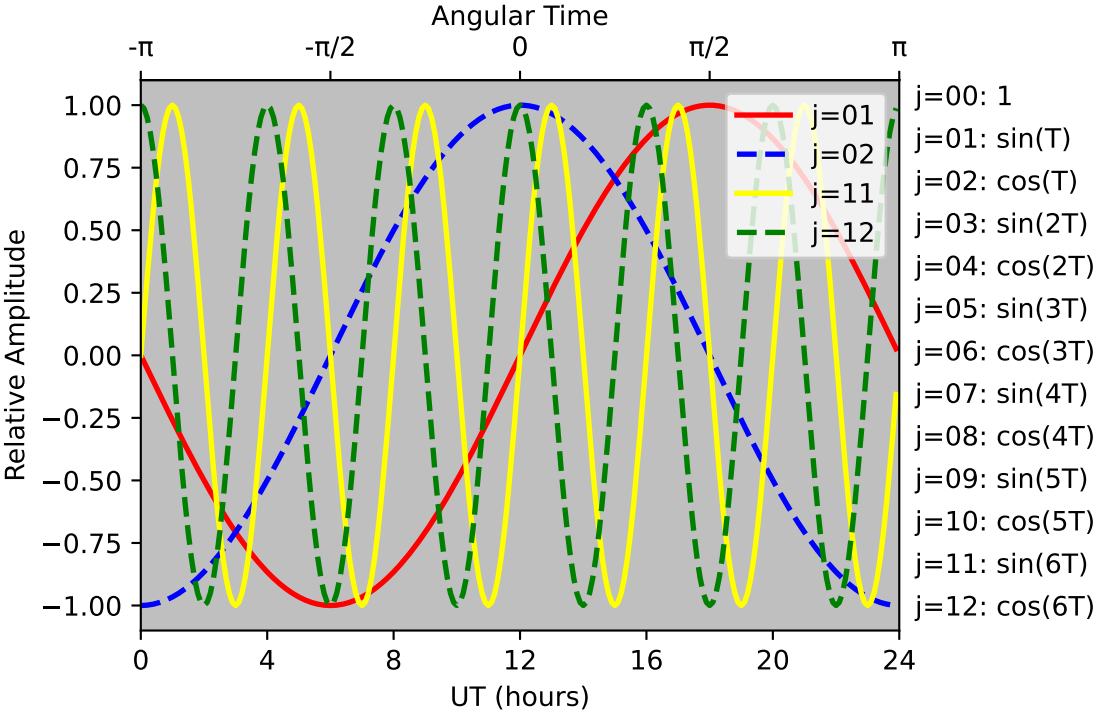

Angular Time
-π
-π/2
0
π/2
π
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
Relative Amplitude
0
4
8
12
16
20
24
UT (hours)
j=01
j=02
j=11
j=12
j=00: 1
j=01: sin(T)
j=02: cos(T)
j=03: sin(2T)
j=04: cos(2T)
j=05: sin(3T)
j=06: cos(3T)
j=07: sin(4T)
j=08: cos(4T)
j=09: sin(5T)
j=10: cos(5T)
j=11: sin(6T)
j=12: cos(6T)

Figure 6.

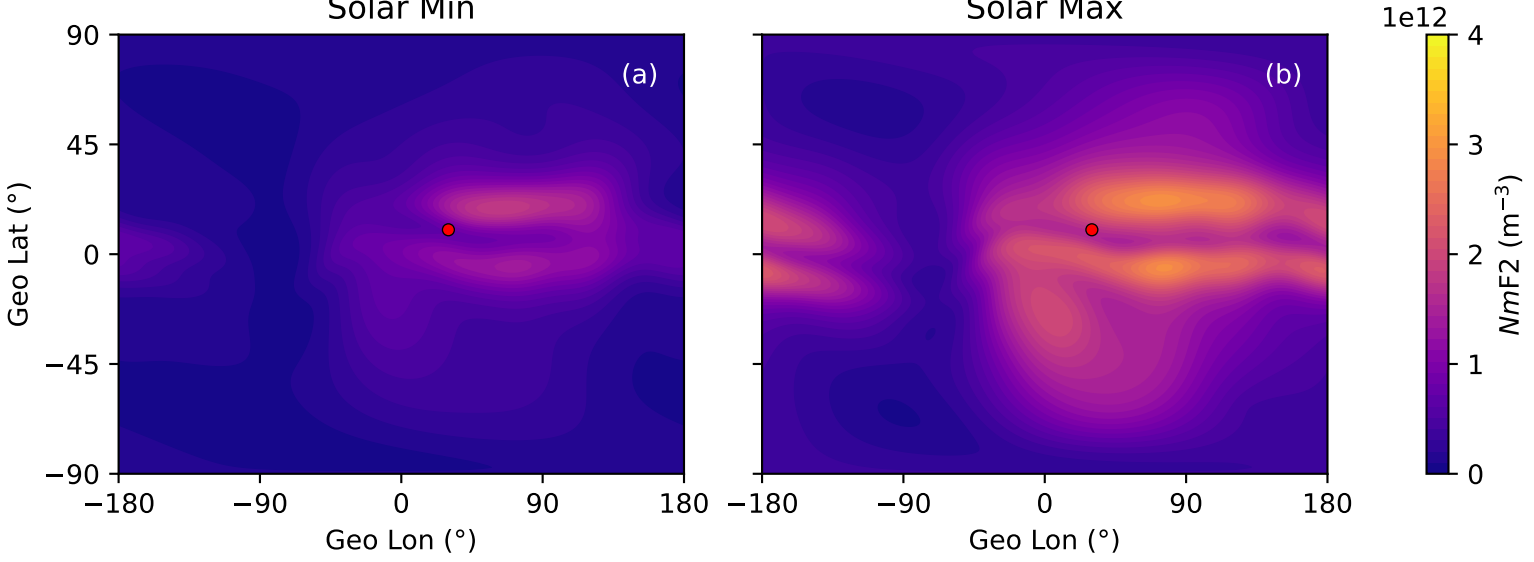

Solar Min
Solar Max
(a)
(b)
Geo Lat (°)
Geo Lon (°)
Geo Lon (°)
90
45
0
−45
−90
−180
−90
0
90
180
1e12
4
3
2
1
0
NmF2 (m−3)

Figure 7.

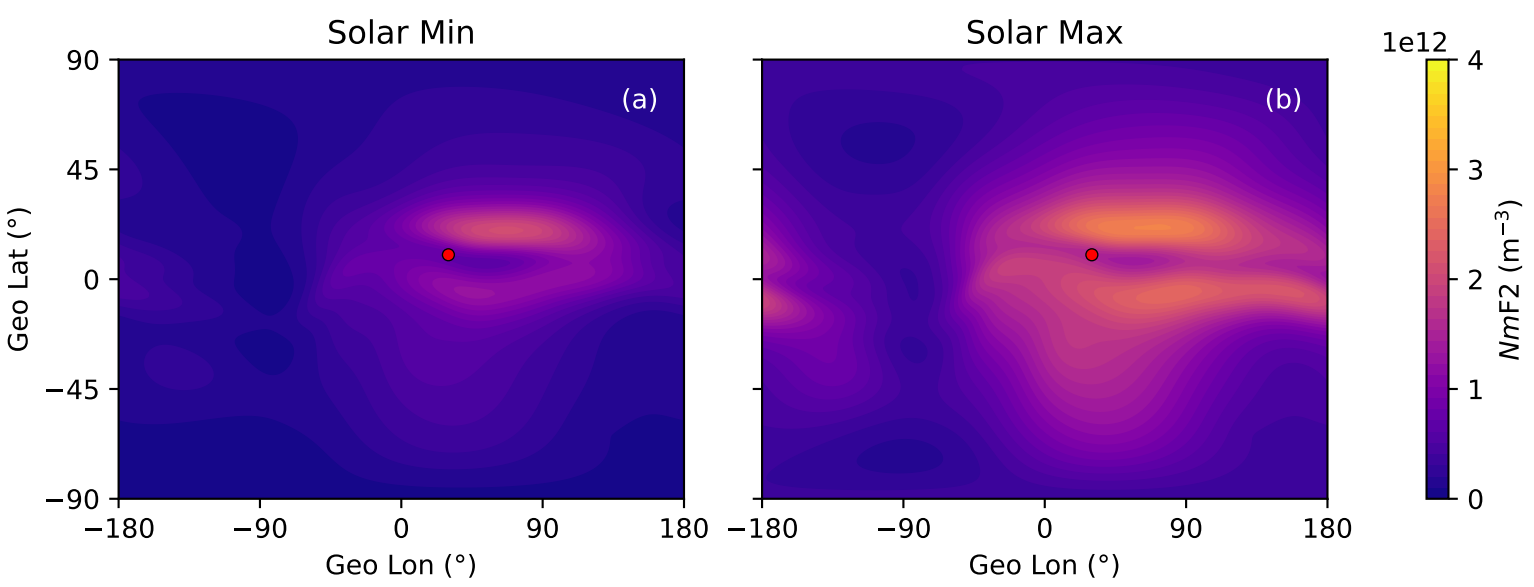

Solar Min
Solar Max
(a)
(b)
90
45
0
−45
−90
Geo Lat (°)
−180
−90
0
90
180
Geo Lon (°)
1e12
4
3
2
1
0
NmF2 (m−3)

Figure 8.

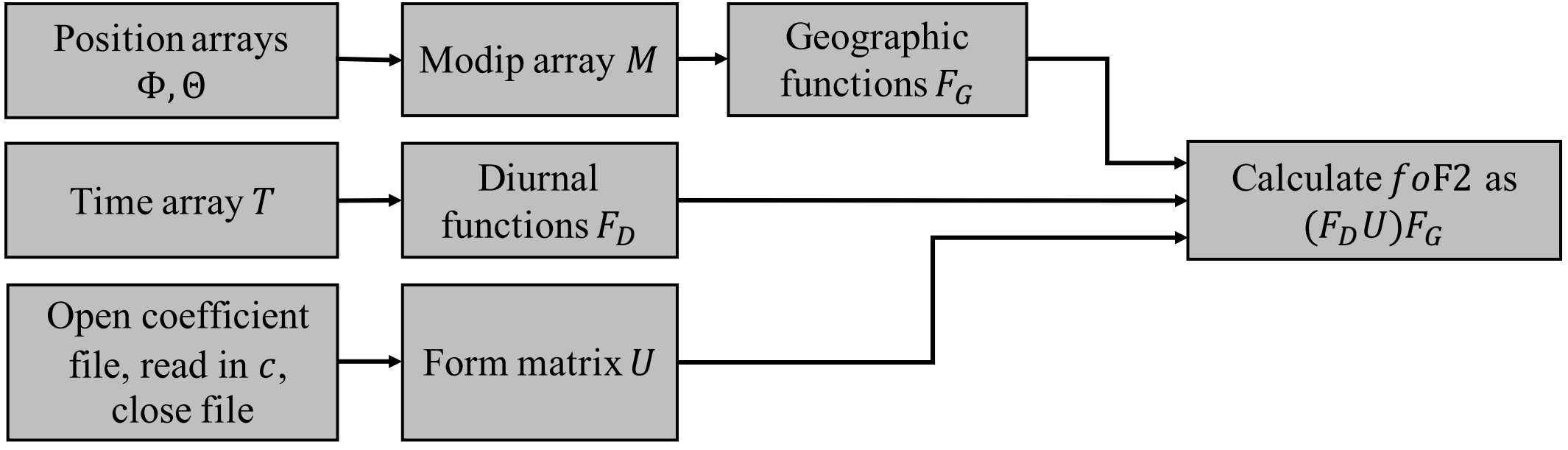

Position arrays
Φ, Θ
Modip array M
Geographic functions F_G
Time array T
Diurnal functions F_D
Calculate $foF2$ as $(F_D U)F_G$
Open coefficient file, read in c, close file
Form matrix U

Figure 9.

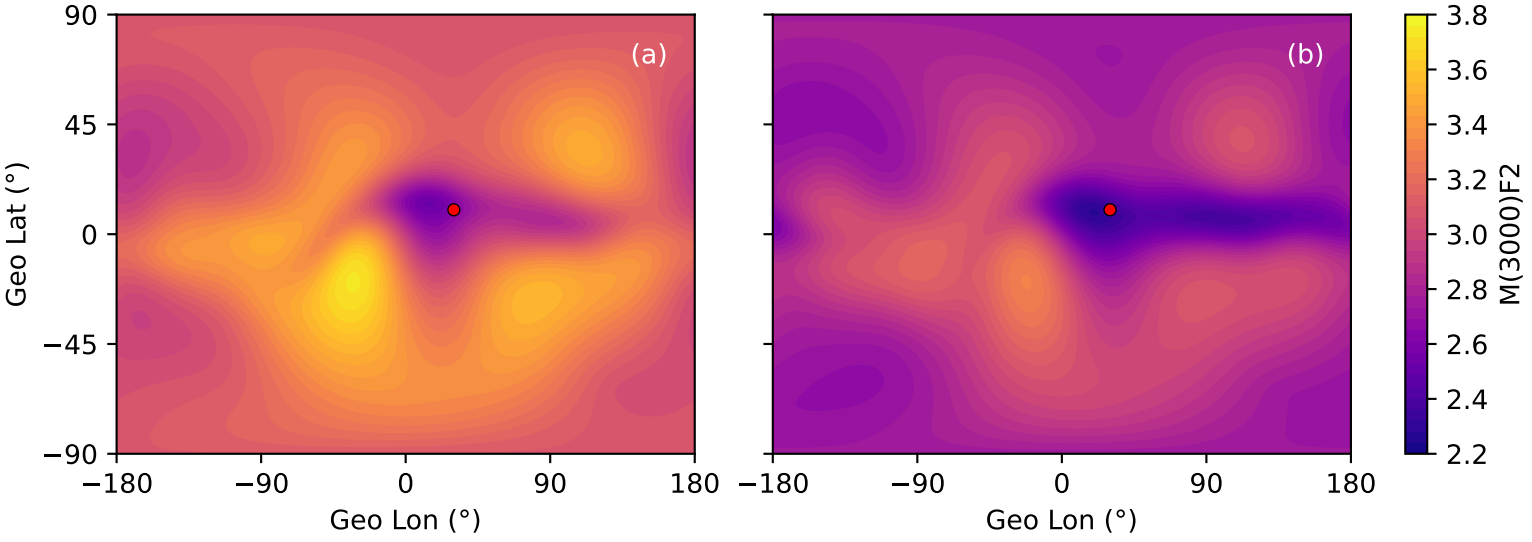

Solar Min
Solar Max
(a)
(b)
90
45
0
−45
−90
Geo Lat (°)
−180
−90
0
90
180
Geo Lon (°)
3.8
3.6
3.4
3.2
3.0
2.8
2.6
2.4
2.2
M(3000)F2

Figure 10.

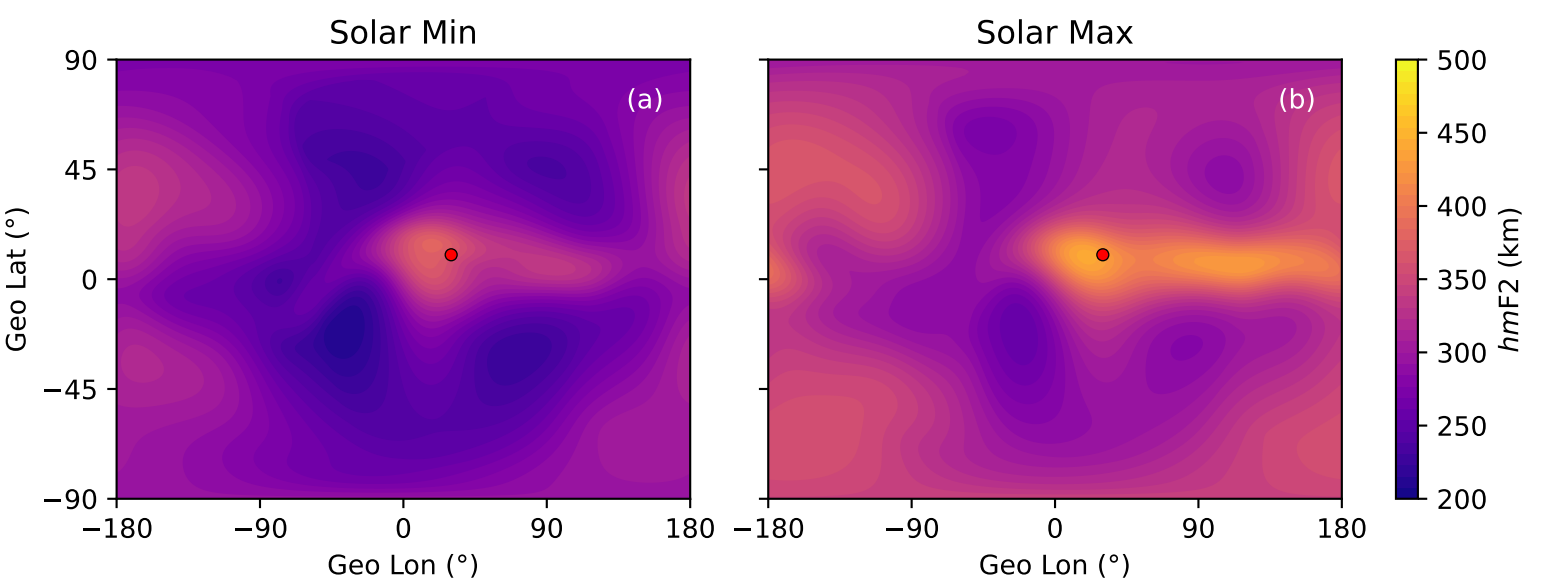

Solar Min
Solar Max
(a)
(b)
90
45
0
−45
−90
Geo Lat (°)
−180
−90
0
90
180
Geo Lon (°)
500
450
400
350
300
250
200
hmF2 (km)

Figure 11.

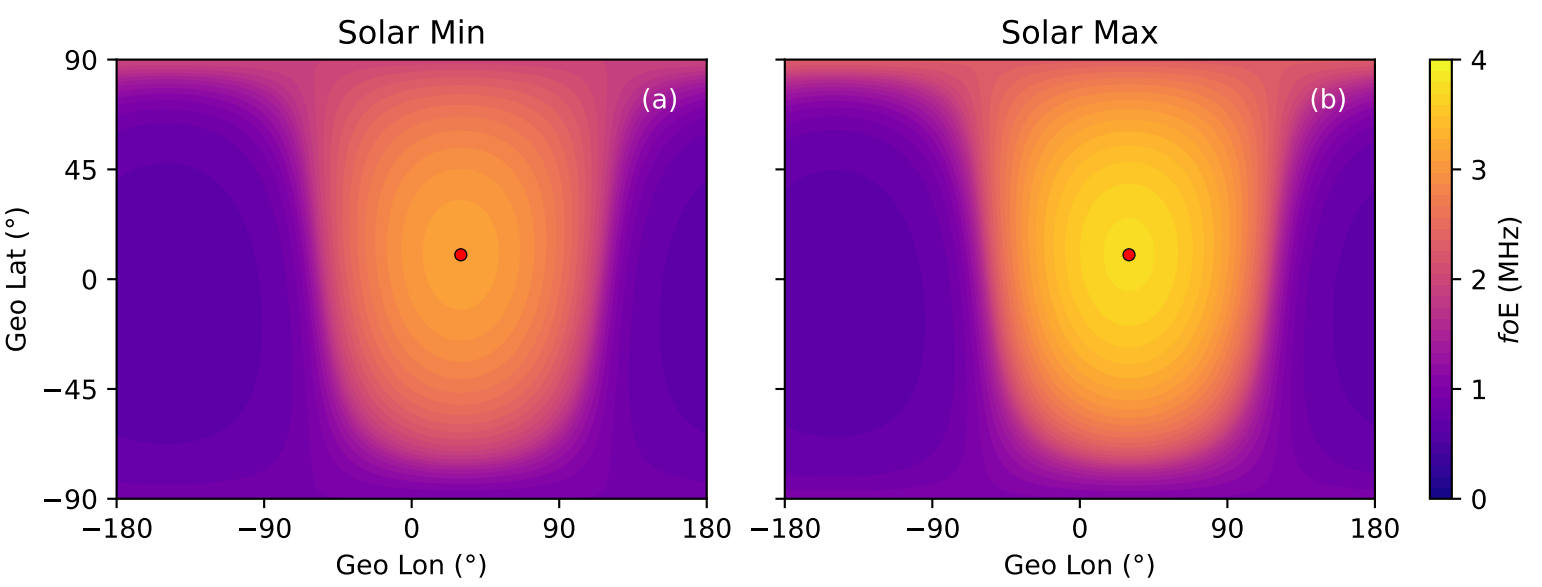

Solar Min
Solar Max
(a)
(b)
90
45
0
−45
−90
Geo Lat (°)
−180
−90
0
90
180
Geo Lon (°)
4
3
2
1
0
foE (MHz)

Figure 12.

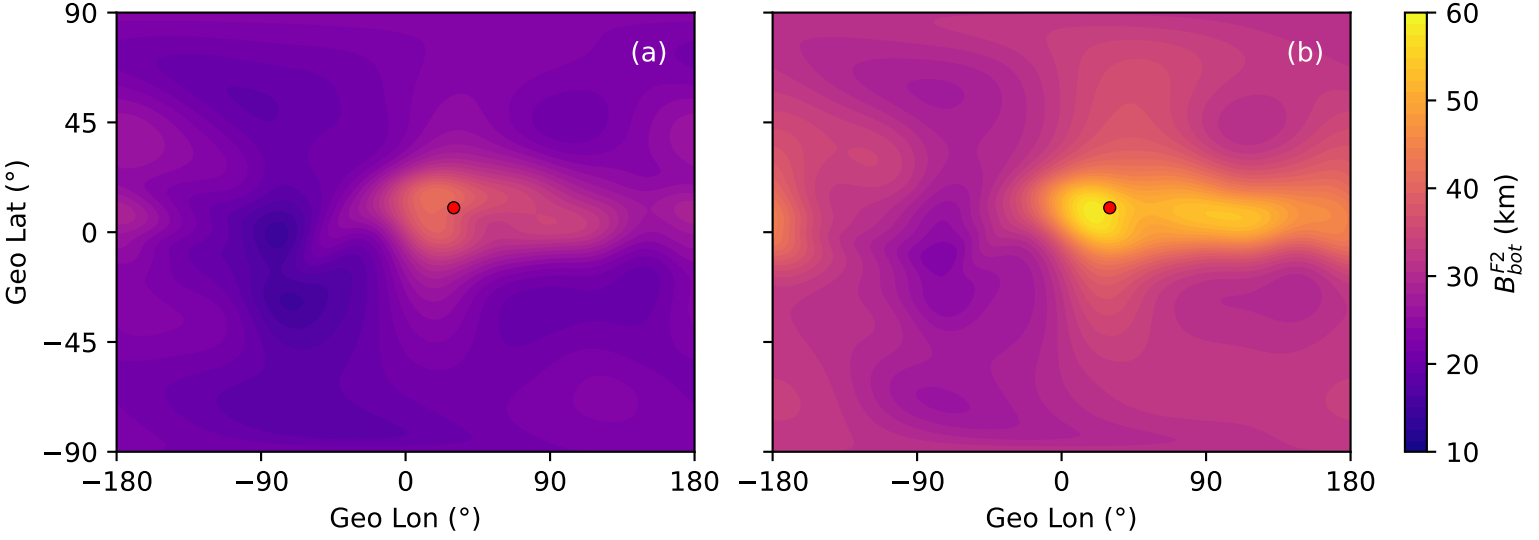

Solar Min
Solar Max
(a)
(b)
90
45
0
−45
−90
Geo Lat (°)
−180
−90
0
90
180
Geo Lon (°)
60
50
40
30
20
10
B_{bot}^{F2} (km)

Figure 13.

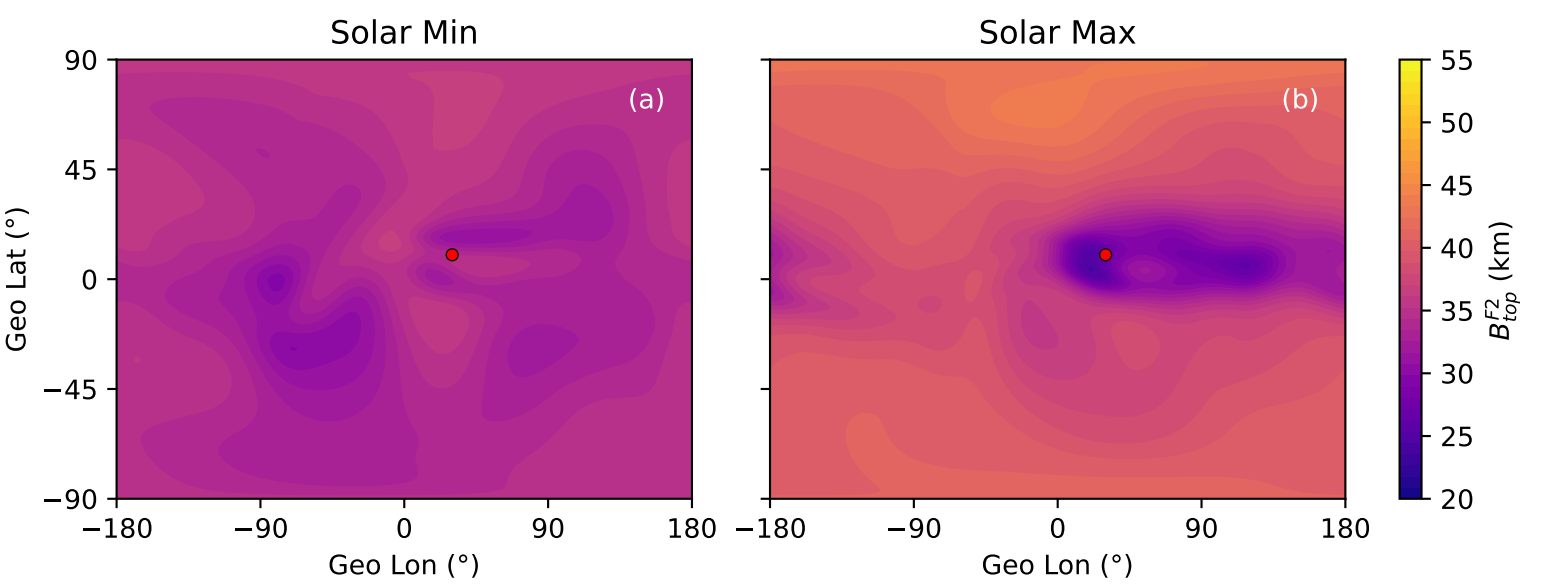

Solar Min
Solar Max
(a)
(b)
90
45
0
−45
−90
Geo Lat (°)
−180
−90
0
90
180
Geo Lon (°)
55
50
45
40
35
30
25
20
B_{top}^{F2} (km)

Figure 14.

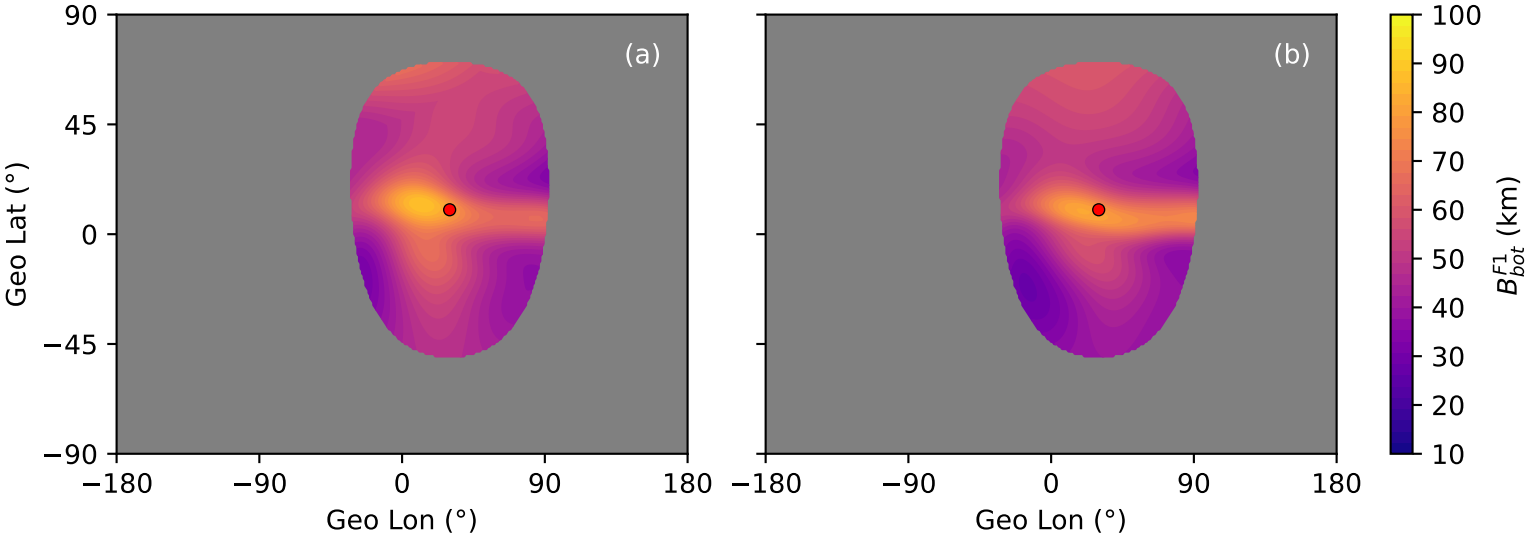

Solar Min
Solar Max
(a)
(b)
90
45
0
−45
−90
Geo Lat (°)
−180
−90
0
90
180
Geo Lon (°)
100
90
80
70
60
50
40
30
20
10
B_{bot}^{F1} (km)

Figure 15.

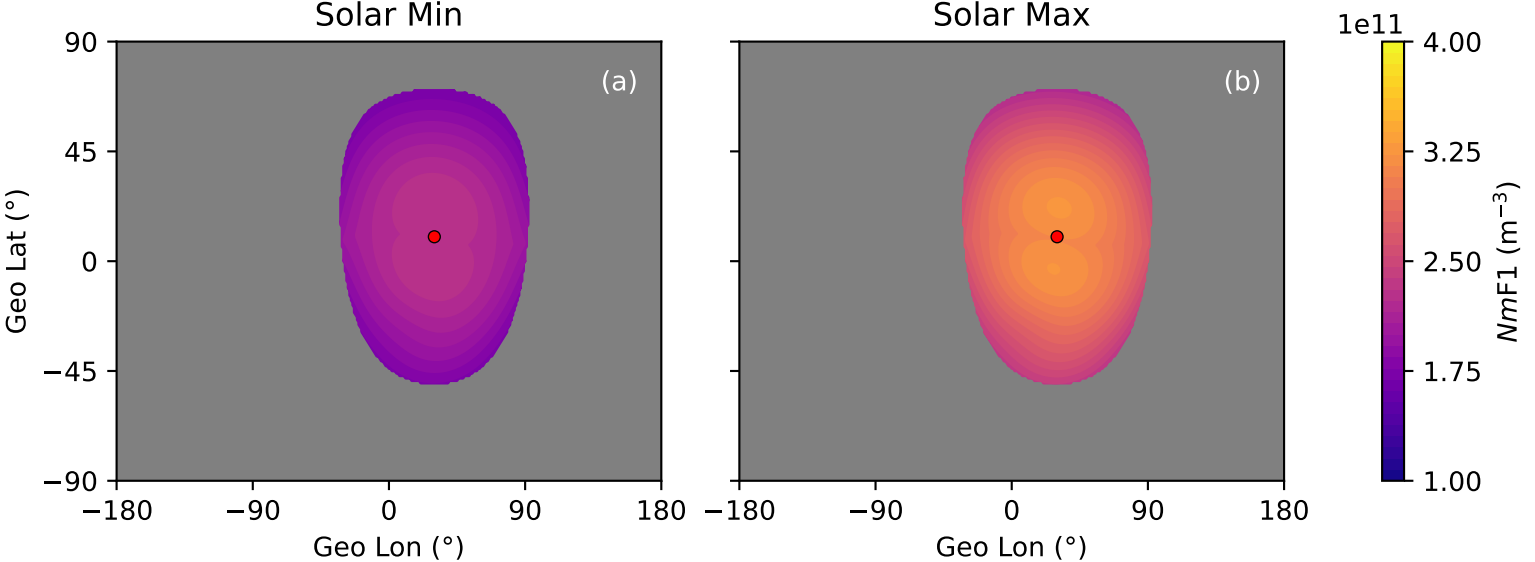

Solar Min
Solar Max
(a)
(b)
90
45
0
−45
−90
Geo Lat (°)
−180
−90
0
90
180
Geo Lon (°)
1e11
4.00
3.25
2.50
1.75
1.00
NmF1 (m−3)

Figure 16.

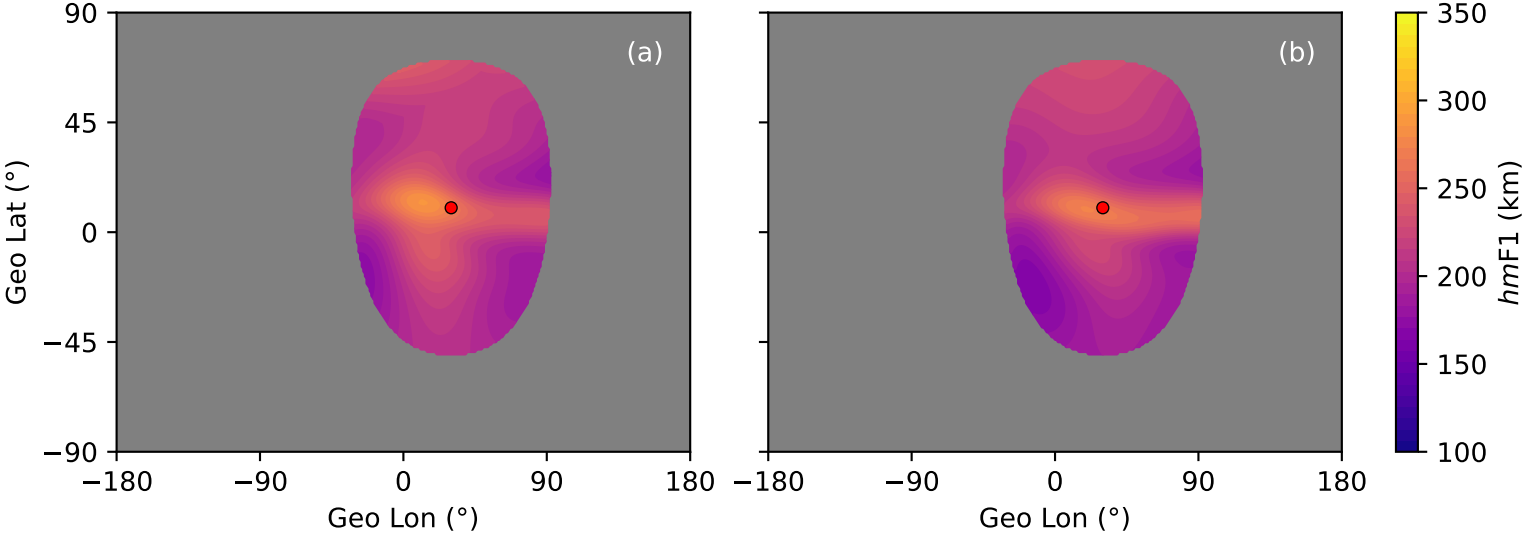

Solar Min
Solar Max
(a)
(b)
90
45
0
−45
−90
Geo Lat (°)
−180
−90
0
90
180
Geo Lon (°)
350
300
250
200
150
100
hmF1 (km)

Figure 17.

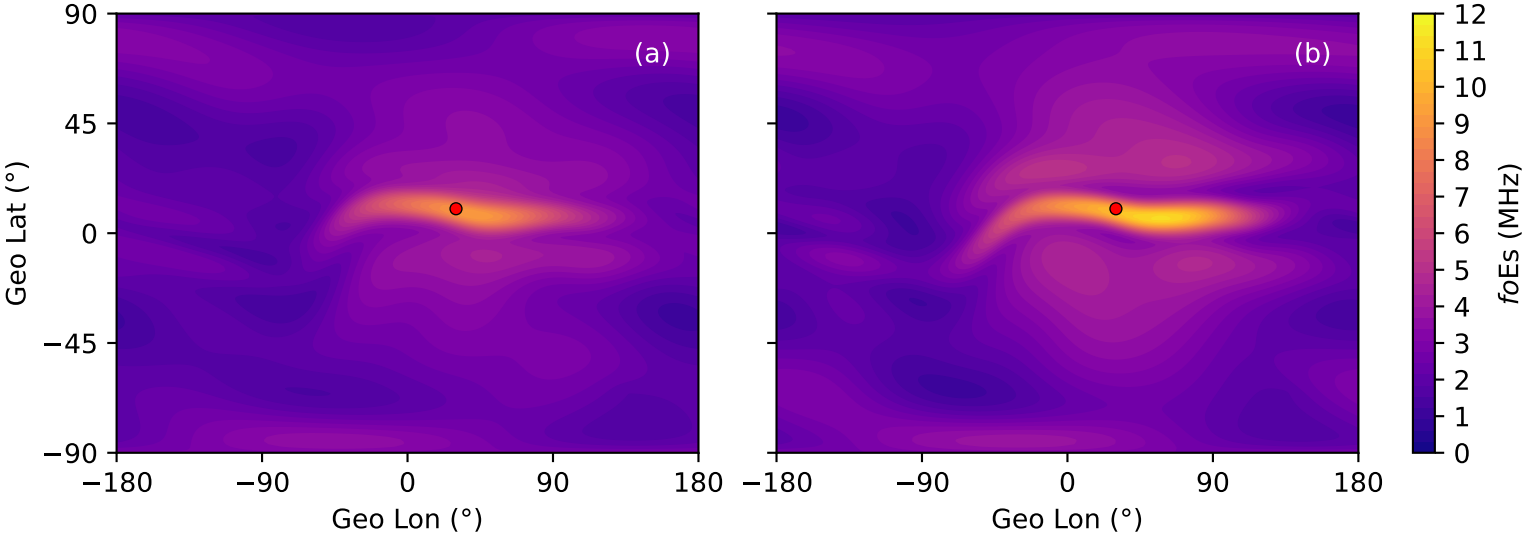

Solar Min
Solar Max
(a)
(b)
Geo Lat (°)
Geo Lon (°)
Geo Lon (°)
90
45
0
−45
−90
−180
−90
0
90
180
foEs (MHz)
0
1
2
3
4
5
6
7
8
9
10
11
12

Figure 18.

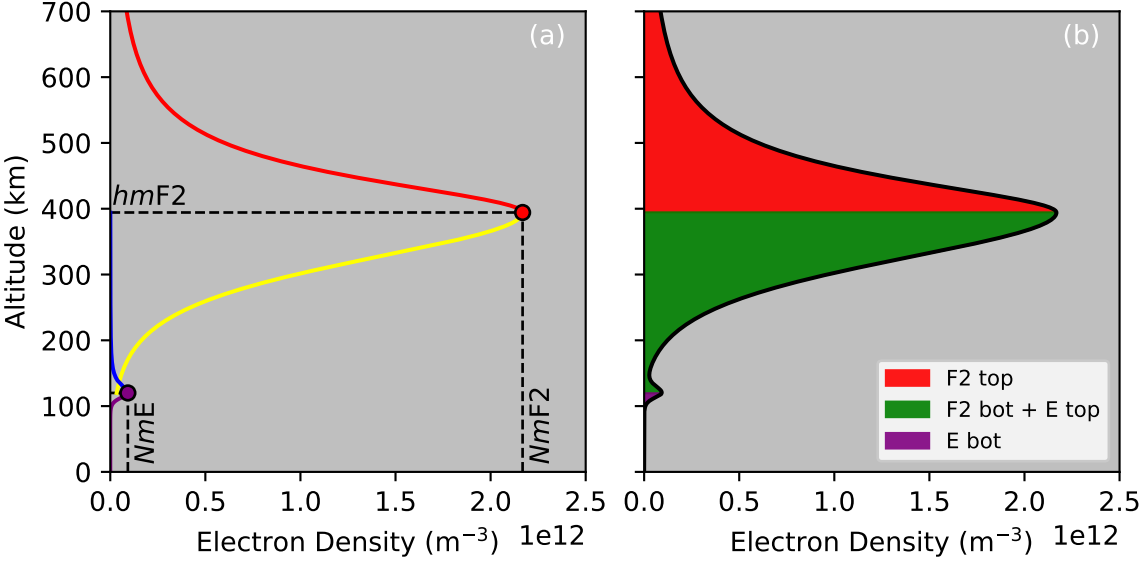

(a)
(b)
Altitude (km)
700
600
500
400
300
200
100
0
hmF2
NmE
NmF2
0.0
0.5
1.0
1.5
2.0
2.5
Electron Density (m^{-3})
1e12
F2 top
F2 bot + E top
E bot

Figure 19.

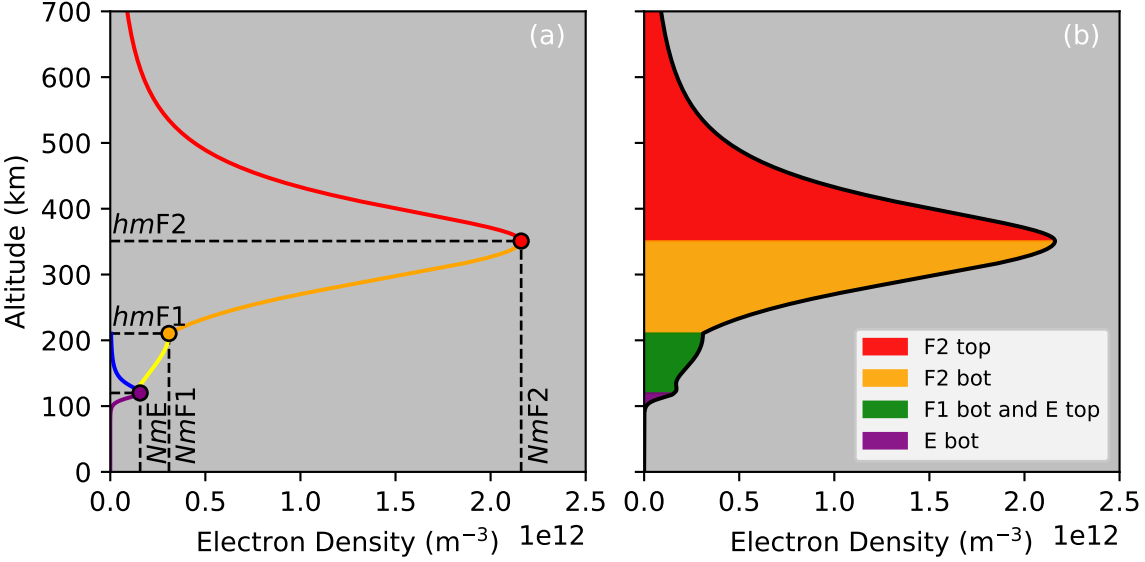

(a)
(b)
Altitude (km)
700
600
500
400
300
200
100
0
hmF2
hmF1
NmE
NmF1
NmF2
0.0
0.5
1.0
1.5
2.0
2.5
Electron Density (m^{-3})
1e12
F2 top
F2 bot
F1 bot and E top
E bot

Figure 20.

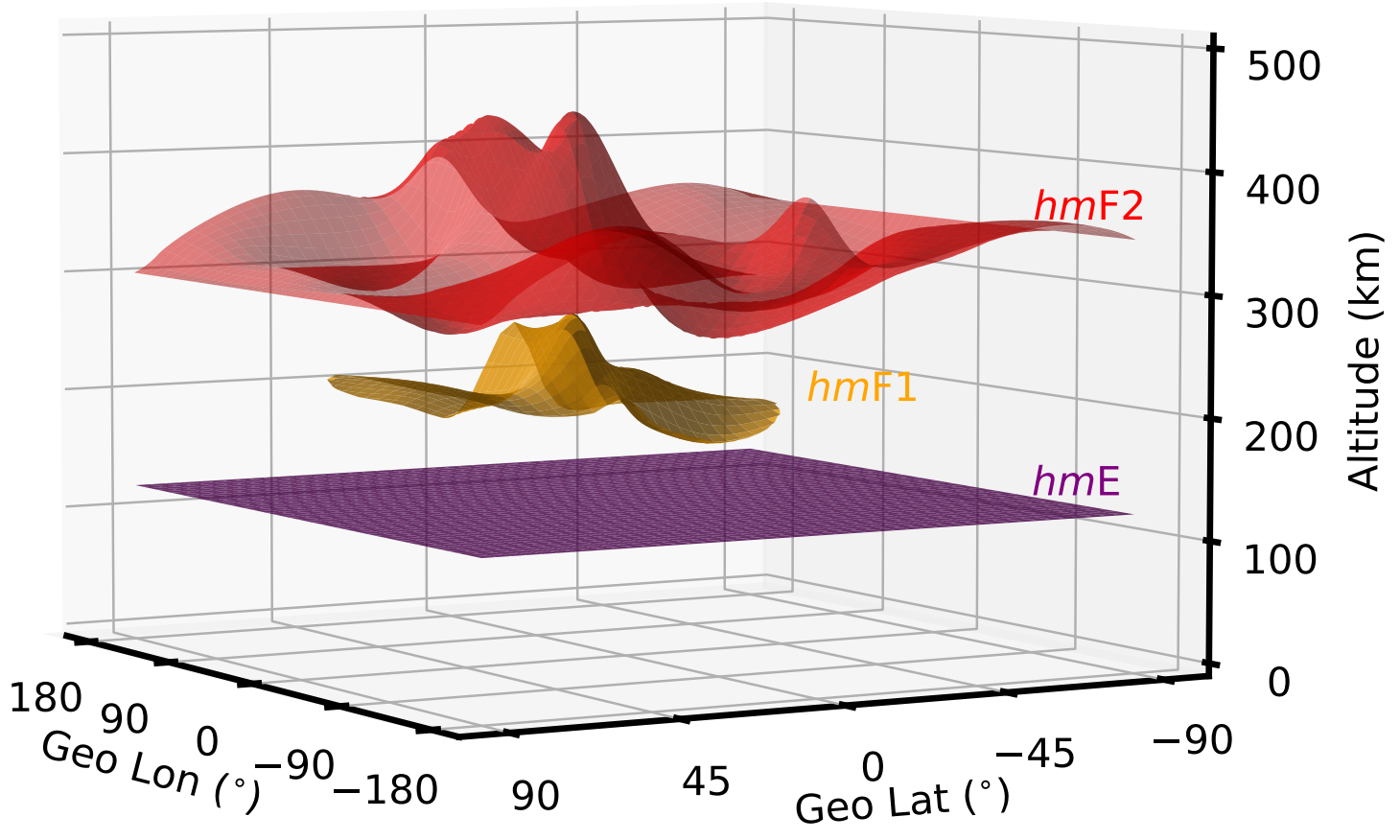
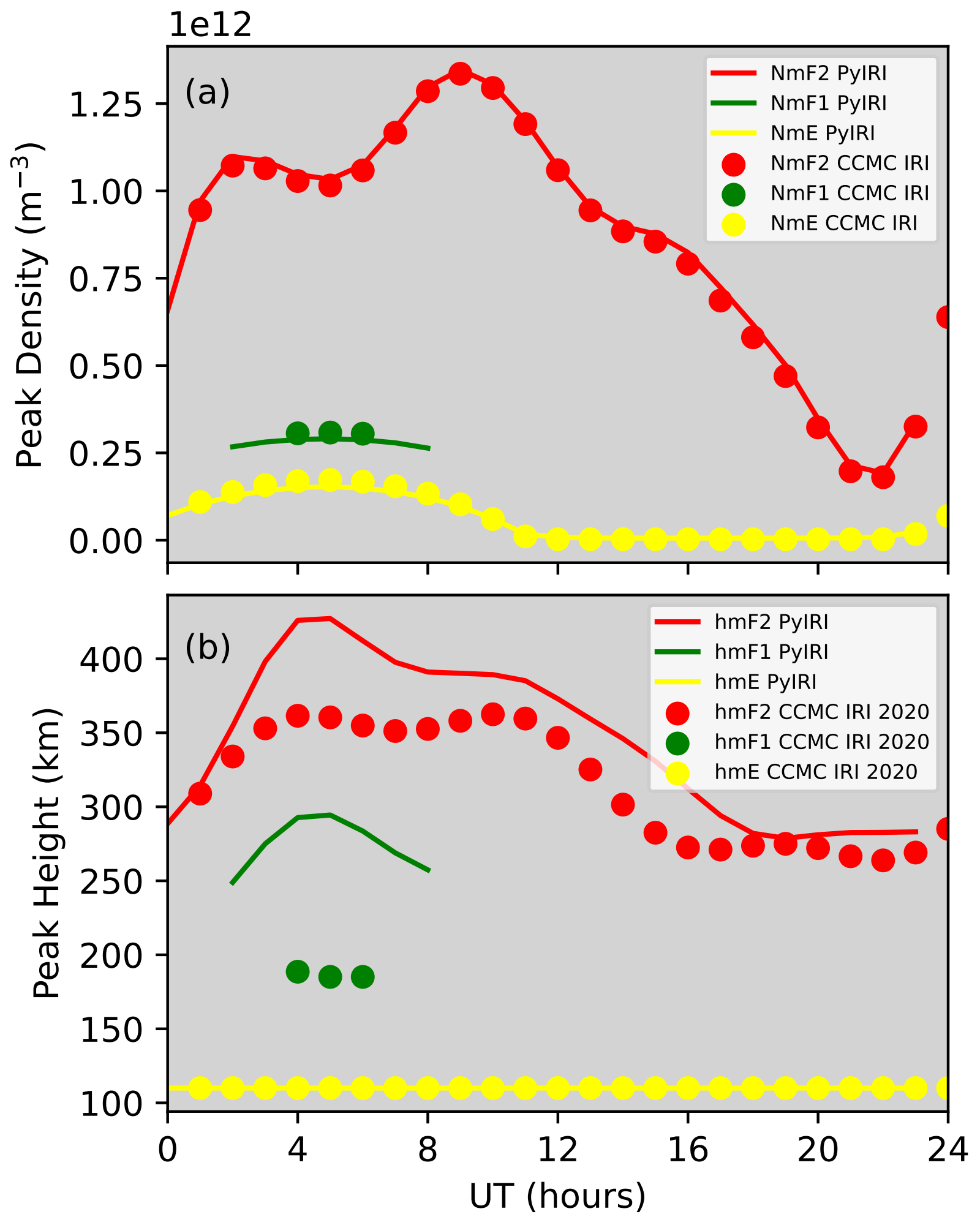

hmF2
hmF1
hmE
500
400
300
200
100
0
Altitude (km)
180
90
0
−90
−180
Geo Lon (°)
90
45
0
−45
−90
Geo Lat (°)

Figure 21.


1e12
(a)
NmF2 PyIRI
NmF1 PyIRI
NmE PyIRI
NmF2 CCMC IRI
NmF1 CCMC IRI
NmE CCMC IRI
Peak Density (m^{-3})
1.25
1.00
0.75
0.50
0.25
0.00
(b)
hmF2 PyIRI
hmF1 PyIRI
hmE PyIRI
hmF2 CCMC IRI 2020
hmF1 CCMC IRI 2020
hmE CCMC IRI 2020
Peak Height (km)
400
350
300
250
200
150
100
0
4
8
12
16
20
24
UT (hours)