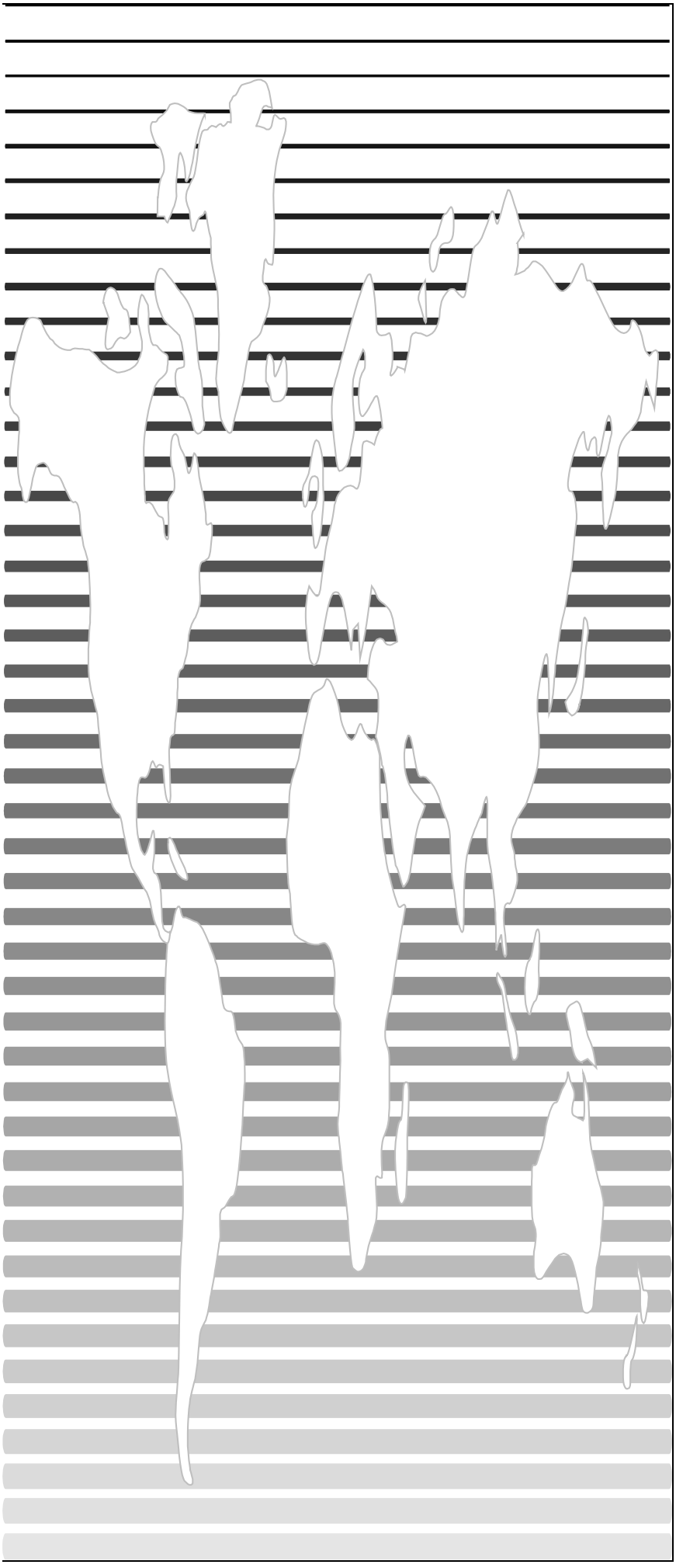


PIM 1.7 User Guide

13-January-1998

Computational Physics, Inc.

Table of Contents

Introducing PIM

The Parameterized Ionospheric Model (PIM) is a fast global ionospheric and plasmaspheric model based on the combined output of regional theoretical ionospheric models and a plasmaspheric model. It consists of portable FORTRAN source code and a large base of data. For specified geophysical conditions and spatial coordinates, PIM produces electron density profiles (*EDP*s) between 90 and 25000 km altitude, corresponding critical frequencies and heights for the ionospheric *E* and F_2 regions, and Total Electron Content (*TEC*).

The ionospheric portion of PIM is a parameterization of the results of several regional theoretical ionospheric models. This allows PIM to be computationally fast while retaining the physics of the theoretical ionospheric models. The parameterization compresses the output from the theoretical ionospheric models while to a large extent it preserves important characteristics such as density peaks and scale heights. The large base of data used by PIM contains coefficients from the parameterization.

The plasmaspheric portion of PIM is the Gallagher plasmaspheric model, a fast empirical model of plasmaspheric H^+.

Refer to Appendix A for further background information on PIM.

PIM considers the following geophysical parameters:

- Year
- Day of the year
- Universal Time
- Solar activity indices $F_{10.7}$ and Sunspot Number (*SSN*)
- Magnetic activity index K_p
- Orientation of the *y* and *z* components of the interplanetary magnetic field (*IMF* B_y and *IMF* B_z)

and the following spatial parameters:

- Coordinate system (geographic or corrected geomagnetic)
- Latitude and longitude
- Azimuth and elevation
- Altitude

All of these parameters are under your control, in addition to a number of program switches that modify PIM's behavior.

Refer to Section 4 for details on PIM inputs.

PIM offers a number of ionospheric parameters on output:

- Ionospheric *E*-region parameters f_oE and h_mE
- Ionospheric F_2-region parameters f_oF_2 and h_mF_2
- *TEC*
- *EDP*s

on several types of output grid:

- Rectangular latitude/longitude
- Latitude/longitude pairs
- Azimuth/elevation with a ground-based observer (origin)

Both the output ionospheric parameters and the output grid type are also under your control.

Refer to Section 4 for details on PIM inputs and Section 5 for details on PIM outputs.

Acquiring PIM

PIM is available from two sources: the Internet and CPI. It is distributed freely on the Internet, and you the user are responsible for its retrieval. PIM is also distributed on CD-ROM from CPI for a small fee.

From the Internet

Because no particular platform (i.e. hardware, operating system, and FORTRAN compiler) has been targeted for PIM, it is distributed on the Internet in as efficient a form as possible that is most likely to be recognizable by an arbitrary platform. Two freely-distributed utilities have been leveraged to distribute PIM on the Internet: TAR and GZIP. TAR is a utility that archives (combines) files in a directory structure into a single file while preserving file names, contents, and relative locations. GZIP is a lossless file compression utility that reduces the size of a file. Both of these utilities are available for most platforms. Some commercial archiving packages, such as WinZip, handle GZIP and TAR formats.

PIM is freely distributed from an Air Force Research Laboratory (AFRL) anonymous FTP site on the Internet in the form of several GZIP-compressed TAR files. The information below is helpful for acquiring PIM from the AFRL FTP site:

Internet address:	**andersun.plh.af.mil** (**146.153.4.202**)		
Login name:	**anonymous**		
Password:	*Your e-mail address*		
Directory:	**pub/pim/archives**		
Compressed TAR files:	**cgmdb.tar.gz**	< 1 MB	Corrected geomagnetic coordinate conversion database
	llfdb.tar.gz	3 MB	Parameterized LLF model database
	lmedb.tar.gz	17 MB	Parameterized LME model database
	mlfdb.tar.gz	7 MB	Parameterized MLF model database
	pim17src.tar.gz	4 MB	PIM 1.7 FORTRAN source code, PIM 1.7 User Guide, IDL procedures, and test cases
	ursidb.tar.gz	< 1 MB	URSI-88 coefficients database
	usudb.tar.gz	13 MB	Parameterized USU model database
	Total size	46 MB	
FTP transfer type:	**binary**		

If you are transferring the compressed TAR files to a file system that does not support long file names (e.g. DOS), then you should change the file name extension of the compressed TAR files

from **.tar.gz** to **.tgz** at the time of the transfer. All directory and file names in the compressed TAR files conform to the 8.3 naming convention.

The AFRL FTP site also contains the following PIM-related resources: the **pub/pim/utils** directory contains TAR and GZIP executables for a number of platforms, and the **pub/pim/help** directory contains the PIM User Guide in a number of formats.

On CD-ROM

PIM is also distributed by CPI on an ISO-9660 CD-ROM. A small fee is charged to cover the cost of creating and distributing the CD-ROM.

The PIM CD-ROM contains a "ready-to-run" PIM executable and binary database files for IBM PC-compatible platforms. It also contains all files (e.g. FORTRAN source code, text versions of database files) needed to install PIM on other platforms. It does not contain compressed TAR files.

Refer to Appendix C for CPI's address and telephone number if you want order PIM on CD-ROM from CPI.

Installing PIM

PIM consists mainly of FORTRAN source code and a large base of data. To minimize the time spent by PIM in accessing the database, PIM requires that the database be in binary form. Since binary formats are seldom platform-independent, the database is normally distributed in text form for portability. While this means extra work for you the user to install PIM, it allows PIM to be used by a wide community. The bulk of the PIM installation process described below involves converting the database from text form to binary form.

System Requirements

Before attempting to install and run PIM, you should be aware of its resource requirements. The information below details the resources necessary to install and run PIM on your platform:

Total size of compressed TAR files:	46 MB
Total size of decompressed TAR files:	113 MB
Maximum disk space required for installing PIM:	306 MB
Minimum disk space required after installing PIM:	41 MB
Memory required for running PIM:	2.5 MB
Compiler required for building PIM:	FORTRAN 77 with common extensions

Installation Instructions

This section describes the steps required for a successful installation of PIM on your platform. The installation instructions that follow assume a UNIX file system; if your platform's operating system is not UNIX, then you may need to translate the directory names in the instructions to the convention appropriate for your file system.

The installation instructions also assume that you have acquired PIM from the Internet. If you have purchased PIM on CD-ROM, then you should ignore the instructions for restoring the PIM file set from the TAR files; however, you will most likely want to follow the installation instructions in order to convert the PIM database from text form to binary form, and also to install PIM on a faster medium (e.g., a hard disk).

1. Create an installation directory and move the compressed TAR files to that location.

In step 3, the TAR utility will restore the PIM file set and directory structure into the directory where the TAR files reside. Create a directory where you want PIM to be installed and move the compressed TAR files into that directory.

2. **Decompress the compressed TAR files using the GZIP utility.**

   Typically, the GZIP command lines for decompressing the compressed TAR files are

   gzip -d cgmdb.tar.gz
   gzip -d llfdb.tar.gz
   gzip -d lmedb.tar.gz
   gzip -d mlfdb.tar.gz
   gzip -d pim17src.tar.gz
   gzip -d ursidb.tar.gz
   gzip -d usudb.tar.gz

   The **-d** switch instructs GZIP to decompress, rather than compress, a file. Note that GZIP replaces the file ***xxx*.tar.gz** with the TAR file ***xxx*.tar**, so you may wish to make backup copies of the compressed TAR files before decompressing them, disk space permitting.

3. **Restore the PIM file set and directory structure from the TAR files.**

   Typically, the TAR command lines for restoring the PIM file set from the TAR files is

   tar -x -f cgmdb.tar
   tar -x -f llfdb.tar
   tar -x -f lmedb.tar
   tar -x -f mlfdb.tar
   tar -x -f pim17src.tar
   tar -x -f ursidb.tar
   tar -x -f usudb.tar

   The **-x** switch instructs TAR to extract the contents of a TAR file; the **-f** switch instructs TAR to take the name of the TAR file from the command-line argument immediately following.

   Table 3.1 in Appendix F describes the PIM file set and directory structure that are restored from the TAR files. All directories and files conform to the 8.3 naming convention and are in lower-case (significant only if your file system is case-sensitive). The directory paths listed in Table 3.1 follow the UNIX convention and are assumed to be relative to the installation directory.

   After successfully restoring the PIM file set from the TAR files, you can optionally delete the TAR files to recover disk space.

4. **Convert the parameterized USU model database from text to binary.**

The parameterized USU model database consists of a set of binary direct-access files. The files restored from the TAR file are in text form for portability. Use the following steps to convert them from text to binaries:

a. Create a subdirectory named **usudb/unform**.

b. In subdirectory **usudb/utils**, compile and link the FORTRAN source files **usutounf.f** and **strnglib.f** to build the executable **usutounf**.

c. In subdirectory **usudb/utils**, edit the first line of file **usutounf.in**. Column 1 of the first line of this file must contain either **B** or **L**, depending on your platform. If record-lengths of direct-access files in FORTRAN OPEN statements are specified in bytes, then **B** should appear in column 1; if record-lengths of direct-access files in FORTRAN OPEN statements are specified in longwords (4-byte units), then **L** should appear in column 1. Most FORTRAN implementations in DOS, Windows NT, and UNIX platforms use bytes; the FORTRAN implementation in the VMS platform uses longwords.

d. Copy the executable **usutounf** and the file **usutounf.in** from subdirectory **usudb/utils** to subdirectory **usudb/form**.

e. In subdirectory **usudb/form**, run the executable **usutounf** with its input stream (console input) originating from file **usutounf.in**. In DOS, Windows NT, and UNIX platforms, the redirection symbol < can be used to define a file as the input stream, as in the example command line

usutounf < usutounf.in

In the VMS platform, the **run** command's **/input** qualifier can be used to define a file as the input stream, as in the example command line

run usutounf.exe /input=usutounf.in

f. Move the resulting binary USU database files, named **nh*.*** and **sh*.***, to subdirectory **usudb/unform**.

g. Optionally, delete subdirectory **usudb/form** and its contents to recover disk space.

h. Optionally, delete the executable **usutounf** and intermediate build files (e.g., object files) from subdirectory **usudb/utils** to recover disk space.

5. Convert the parameterized MLF model database from text to binary.

The parameterized MLF model database consists of a set of binary direct-access files. The files restored from the TAR file are in text form for portability. Use the following steps to convert them from text to binaries:

a. Create a subdirectory named **mlfdb/unform**.

b. In subdirectory **mlfdb/utils**, compile and link the FORTRAN source files **mlftounf.f** and **strnglib.f** to build the executable **mlftounf**.

c. In subdirectory **mlfdb/utils**, edit the first line of file **mlftounf.in**. Column 1 of the first line of this file must contain either **B** or **L**, depending on your platform. If record-lengths of direct-access files in FORTRAN OPEN statements are specified in bytes, then **B** should appear in column 1; if record-lengths of direct-access files in FORTRAN OPEN statements are specified in longwords (4-byte units), then **L** should appear in column 1. Most FORTRAN implementations in DOS, Windows NT, and UNIX platforms use bytes; the FORTRAN implementation in the VMS platform uses longwords.

d. Copy the executable **mlftounf** and the file **mlftounf.in** from subdirectory **mlfdb/utils** to subdirectory **mlfdb/form**.

e. In subdirectory **mlfdb/form**, run the executable **mlftounf** with its input stream (console input) originating from file **mlftounf.in**. In DOS, Windows NT, and UNIX platforms, the redirection symbol < can be used to define a file as the input stream, as in the example command line

mlftounf < mlftounf.in

In the VMS platform, the **run** command's **/input** qualifier can be used to define a file as the input stream, as in the example command line

run mlftounf.exe /input=mlftounf.in

f. Move the resulting binary MLF database files, named **nm*.*** and **sm*.***, to subdirectory **mlfdb/unform**.

g. Optionally, delete subdirectory **mlfdb/form** and its contents to recover disk space.

h. Optionally, delete the executable **mlftounf** and intermediate build files (e.g., object files) from subdirectory **mlfdb/utils** to recover disk space.

6. Convert the parameterized LLF model database from text to binary.

The parameterized LLF model database consists of a set of binary direct-access files. The files restored from the TAR file are in text form for portability. Use the following steps to convert them from text to binaries:

a. Create a subdirectory named **llfdb/unform**.

b. In subdirectory **llfdb/utils**, compile and link the FORTRAN source files **llftounf.f** and **strnglib.f** to build the executable **llftounf**.

c. In subdirectory **llfdb/utils**, edit the first line of file **llftounf.in**. Column 1 of the first line of this file must contain either **B** or **L**, depending on your platform. If record-lengths of direct-access files in FORTRAN OPEN statements are specified in bytes, then **B** should appear in column 1; if record-lengths of direct-access files in FORTRAN OPEN statements are specified in longwords (4-byte units), then **L** should appear in column 1. Most FORTRAN implementations in DOS, Windows NT, and UNIX platforms use bytes; the FORTRAN implementation in the VMS platform uses longwords.

d. Copy the executable **llftounf** and the file **llftounf.in** from subdirectory **llfdb/utils** to subdirectory **llfdb/form**.

e. In subdirectory **llfdb/form**, run the executable **llftounf** with its input stream (console input) originating from file **llftounf.in**. In DOS, Windows NT, and UNIX platforms, the redirection symbol < can be used to define a file as the input stream, as in the example command line

llftounf < llftounf.in

In the VMS platform, the **run** command's **/input** qualifier can be used to define a file as the input stream, as in the example command line

run llftounf.exe /input=llftounf.in

f. Move the resulting binary LLF database files, named **brz*.***, **ind*.***, **pac*.***, and **usa*.***, to subdirectory **llfdb/unform**.

g. Optionally, delete subdirectory **llfdb/form** and its contents to recover disk space.

h. Optionally, delete the executable **llftounf** and intermediate build files (e.g., object files) from subdirectory **llfdb/utils** to recover disk space.

7. Convert the parameterized LME model database from text to binary.

The parameterized LME model database consists of a set of binary direct-access files. The files restored from the TAR file are in text form for portability. Use the following steps to convert them from text to binaries:

a. Create a subdirectory named **lmedb/unform**.

b. In subdirectory **lmedb/utils**, compile and link the FORTRAN source files **lmetounf.f** and **strnglib.f** to build the executable **lmetounf**.

c. In subdirectory **lmedb/utils**, edit the first line of file **lmetounf.in**. Column 1 of the first line of this file must contain either **B** or **L**, depending on your platform. If record-lengths of direct-access files in FORTRAN OPEN statements are specified in bytes, then **B** should appear in column 1; if record-lengths of direct-access files in FORTRAN OPEN statements are specified in longwords (4-byte units), then **L** should appear in column 1. Most FORTRAN implementations in DOS, Windows NT, and UNIX platforms use bytes; the FORTRAN implementation in the VMS platform uses longwords.

d. Copy the executable **lmetounf** and the file **lmetounf.in** from subdirectory **lmedb/utils** to subdirectory **lmedb/form**.

e. In subdirectory **lmedb/form**, run the executable **lmetounf** with its input stream (console input) originating from file **lmetounf.in**. In DOS, Windows NT, and UNIX platforms, the redirection symbol < can be used to define a file as the input stream, as in the example command line

lmetounf < lmetounf.in

In the VMS platform, the **run** command's **/input** qualifier can be used to define a file as the input stream, as in the example command line

run lmetounf.exe /input=lmetounf.in.

f. Move the resulting binary LME database files, named **lm*.***, to subdirectory **lmedb/unform**.

g. Optionally, delete subdirectory **lmedb/form** and its contents to recover disk space.

h. Optionally, delete the executable **lmetounf** and intermediate build files (e.g., object files) from subdirectory **lmedb/utils** to recover disk space.

8. Convert the URSI-88 coefficients database from text to binary.

The URSI-88 coefficients database consists of a binary direct-access file. The file restored from the TAR file is in text form for portability. Use the following steps to convert it from text to binary:

a. Create a subdirectory named **ursidb/unform**.

b. In subdirectory **ursidb/utils**, compile and link the FORTRAN source file **ursistod.f** to build the executable **ursistod**.

c. Copy the executable **ursistod** from subdirectory **ursidb/utils** to subdirectory **ursidb/form**.

d. In subdirectory **ursidb/form**, run the executable **ursistod**. When prompted with the question **In OPEN statements, are record-lengths of unformatted direct-access files specified in bytes or longwords (a longword is a 4-byte unit) ([B]/L)?**, answer either **B** or **L**, depending on your platform. If record-lengths of direct-access files in FORTRAN OPEN statements are specified in bytes, then answer **B**; if record-lengths of direct-access files in FORTRAN OPEN statements are specified in longwords (4-byte units), then answer **L**. Most FORTRAN implementations in DOS, Windows NT, and UNIX platforms use bytes; the FORTRAN implementation in the VMS platform uses longwords.

e. Move the resulting binary URSI-88 coefficients database file, named **ursi88da.dat**, to subdirectory **ursidb/unform**.

f. Optionally, delete subdirectory **ursidb/form** and its contents to recover disk space.

g. Optionally, delete the executable **ursistod** and intermediate build files (e.g., an object file) from subdirectory **ursidb/utils** to recover disk space.

9. Build the PIM executable.

The PIM source code is written in ANSI FORTRAN 77 with common extensions (e.g., long variable names, INCLUDE statements, DO-ENDDO loop constructs, etc.). Since most modern FORTRAN 77 compilers extend the ANSI FORTRAN 77 standard in similar ways, the PIM source code should compile without modification. FORTRAN 77 compilers that strictly adhere to the ANSI FORTRAN 77 standard will not successfully compile PIM as it is distributed by CPI. Use the following steps to build the PIM executable:

a. Create a subdirectory named **pim/bin**.

b. If your platform supports a MAKE utility:

i. In subdirectory **pim/source**, modify the example MAKE file to work with your platform's MAKE implementation.

ii. In subdirectory **pim/source**, run MAKE to build the PIM executable.

c. If your platform does not support a MAKE utility:

i. In subdirectory **pim/source**, compile each of the FORTRAN source files (***.f**).

ii. In subdirectory **pim/source**, link the resulting object files to build the PIM executable.

d. Move the resulting PIM executable to subdirectory **pim/bin**.

e. Optionally, delete intermediate build files (e.g., object files) from subdirectory **pim/source** to recover disk space.

10. Create the path_nam.txt **input file.**

PIM requires the presence of a text file named **path_nam.txt** in the directory in which it is run (in this installation **pim/bin** is the working directory). The file contains the paths (locations) of the six database inputs to PIM. It also contains a flag that instructs PIM how to open direct-access database files. In subdirectory **pim/bin**, create the file **path_nam.txt** to reflect your file system and the directory structure of your PIM installation.

Refer to Section 4 for details of the contents of the file **path_nam.txt**.

11. Check your installation.

At this point you may want to review what you've done so far and make sure that you haven't missed any steps.

Table 3.2 in Appendix F describes the PIM file set and directory structure up to this point in the installation. Table 3.2 assumes that you've followed all of the above steps, including the optional steps to recover disk space, and that you've created files and directories as they are named in the instructions.

12. Run PIM with a test case.

A number of PIM runs are provided in subdirectory **pim/testcase** as a means of testing your installation. You may want to run PIM for at least one of the test cases to make sure that your installation is correct. Small differences between real numbers are due to round-off or differences in precision.

Refer to Section 6 for details on running PIM.

PIM Inputs

PIM requires user input from two sources and database input from six sources. User input is defined as input that you the user can modify to tailor PIM to your needs. It includes items such as the solar activity index $F_{10.7}$, latitude and longitude, and altitude. Database input is defined as input that PIM needs to function but which you should not modify unless you have an in-depth knowledge of the internals of PIM. It includes items such as the parameterized model databases.

User Input

User input is input that you can modify to tailor PIM to your needs. PIM gets user input from two sources:

1. **File** path_nam.txt

   PIM requires the presence of a text file named **path_nam.txt** in the directory in which it is run. This file contains the paths (locations) of the six database inputs to PIM. It also contains a flag that instructs PIM how to open direct-access database files. You may need to create or modify **path_nam.txt** when PIM is installed and whenever the PIM file set is moved or rearranged.

Table 4.1 in Appendix F details the structure and contents of **path_nam.txt**.

2. **Input stream**

   The input stream (default or console input) is the main source of user input for PIM. Because PIM does not prompt you for what it needs from the input stream, it is recommended that you create a file containing the necessary information and define it as the input stream at run time.

Table 4.2 in Appendix F details the structure and contents of the PIM input stream. Refer to Section 6 for information on running PIM.

Database Input

Database input is defined as input that PIM needs to function but which you should not modify unless you have an in-depth knowledge of the internals of PIM. PIM gets database input from six sources:

1. Corrected geomagnetic coordinate conversion database

PIM uses the corrected geomagnetic coordinate conversion database whenever it needs to convert between the corrected geomagnetic and geographic coordinate systems. For example, the parameterized model databases are defined in geomagnetic coordinates; if you request output in geographic coordinates (see input stream element CRDTYPE), then PIM maps geographic coordinates to corrected geomagnetic coordinates in order to use the parameterized model databases. The corrected geomagnetic coordinate conversion database consists of a text file named **cglalo.dat** whose location is defined in the user input file **path_nam.txt**.

Refer to Table 4.1 in Appendix F for a description of **path_nam.txt**. The location of the corrected coordinate conversion database is defined by element CGMPATH in that table.

2. Parameterized USU model database

PIM uses the parameterized USU model database to construct vertical NO^+, O_2^+, and O^+ number density altitude profiles in the geomagnetic high-latitude regions. The parameterized USU model database was derived from the output of the theoretical USU ionospheric model, and consists of a set of binary direct-access files whose location is defined in the user input file **path_nam.txt**. The files' names encode the geophysical conditions under which the USU model was run.

Refer to Table 4.1 in Appendix F for a description of **path_nam.txt**. The location of the binary parameterized USU model database is defined by element USUPATH in that table. Table 4.3 in Appendix F describes the encoding of the file names in the parameterized USU model database.

3. Parameterized MLF model database

PIM uses the parameterized MLF model database to construct vertical O^+ number density altitude profiles in the geomagnetic mid-latitude regions. The parameterized MLF model database was derived from the output of the theoretical MIDLAT ionospheric model, and consists of a set of binary direct-access files whose location is defined in the user input file **path_nam.txt**. The files' names encode the geophysical conditions under which the MIDLAT model was run.

Refer to Table 4.1 in Appendix F for a description of **path_nam.txt**. The location of the binary parameterized MLF model database is defined by element MLFPATH in that table. Table 4.4 in Appendix F describes the encoding of the file names in the parameterized MLF model database.

4. Parameterized LLF model database

PIM uses the parameterized LLF model database to construct vertical O^+ number density altitude profiles in the geomagnetic low-latitude region. The parameterized LLF model database was derived from the output of the theoretical LOWLAT ionospheric model, and consists of a set of binary direct-access files whose location is defined in the user input file **path_nam.txt**. The files' names encode the geophysical conditions under which the LOWLAT model was run.

Refer to Table 4.1 in Appendix F for a description of **path_nam.txt**. The location of the binary parameterized LLF model database is defined by element LLFPATH in that table. Table 4.5 in Appendix F describes the encoding of the file names in the parameterized LLF model database.

5. Parameterized LME model database

PIM uses the parameterized LME model database to construct vertical NO^+ and O_2^+ number density altitude profiles in the geomagnetic low- and mid- latitude regions. The parameterized LME model database was derived from the output of the theoretical ECSD ionospheric model, and consists of a set of binary direct-access files whose location is defined in the user input file **path_nam.txt**. The files' names encode the geophysical conditions under which the LME model was run.

Refer to Table 4.1 in Appendix F for a description of **path_nam.txt**. The location of the binary parameterized LME model database is defined by element LMEPATH in that table. Table 4.6 in Appendix F describes the encoding of the file names in the parameterized LME model database.

6. URSI-88 coefficients database

PIM uses the URSI-88 coefficients database to normalize f_oF_2 (if instructed to do so through the PIM input stream). The URSI-88 coefficients database consists of a binary direct-access file named **ursi88da.dat** whose location is defined in the user input file **path_nam.txt**.

Refer to Table 4.1 in Appendix F for a description of **path_nam.txt**. The location of the binary URSI-88 coefficients database is defined by element URSIPATH in that table. Refer to Table 4.2 in Appendix F for a description of the PIM input stream. The normalization of f_oF_2 is controlled by element FOF2NORM in that table.

PIM Outputs

PIM sends output to two destinations: the output stream and an output file.

Output Stream

The output stream (default or console output) is used by PIM to display run-time status and error messages. It is recommended that you save the output stream in a file at run-time in order to have a permanent record (log) of any error messages produced by PIM.

Refer to Section 6 for information on running PIM.

Output File

PIM uses a text output file to store the results of its calculations based on your input. The name of the output file is defined in the PIM input stream.

Refer to Table 4.2 in Appendix F for a description of the PIM input stream. Table 5.1 in Appendix F details the structure and contents of the PIM output file.

Running PIM

If you are at all familiar with the command line of the DOS, Windows NT, UNIX, or VMS platforms, then you will find it easy to run PIM.

Prerequisites

Before you can successfully run PIM, you must do the following:

1. **Complete the installation of PIM.**

Refer to Section 3 for details on the installation of PIM.

2. **Define a PIM input stream.**

Refer to Section 4 for details on the PIM input stream.

The Command Line

Because PIM is a *text-mode* program (i.e., it contains no implicit graphical capabilities), and because it makes use of the input and output streams (default or console input and output), PIM is most easily run from a command line.

In DOS, Windows NT, and UNIX platforms, the general form of the command line for running PIM is:

pim-executable **[<** ***pim-input-stream*****] [>** ***pim-output-stream*****]**

where ***pim-executable*** is the name of the PIM executable, ***pim-input-stream*** is the name of the file containing the PIM input stream, and ***pim-output-stream*** is the name of the file that will contain the PIM output stream. The redirection symbol < instructs the command-line processor to take the input stream (console input) for PIM from the file ***pim-input-stream*** rather from the default input device (e.g., a keyboard), and the redirection symbol > instructs the command-line processor to send the output stream (console output) from PIM to the file ***pim-output-stream*** rather than to the default output device (e.g., a monitor). The square brackets **[]** delimit optional sections of the command line; however, it is recommended that you at least define the PIM input stream as a file because PIM does not prompt you for what it needs.

In the VMS platform, the general form of the command line for running PIM is:

run *pim-executable* [/input= *pim-input-stream*] [/output= *pim-output-stream*]

where ***pim-executable*** is the name of the PIM executable, ***pim-input-stream*** is the name of the file containing the PIM input stream, and ***pim-output-stream*** is the name of the file that will contain the PIM output stream. The **run** command qualifier **/input=** instructs the **run** command to take the input stream (console input) for PIM from the file ***pim-input-stream*** rather from the default input device (e.g., a keyboard), and the **run** command qualifier **/output=** instructs the **run** command to send the output stream (console output) from PIM to the file ***pim-output-stream*** rather than to the default output device (e.g., a monitor). The square brackets **[]** delimit optional sections of the command line; however, it is recommended that you at least define the PIM input stream as a file because PIM does not prompt you for what it needs.

Refer to Section 4 for details of the PIM input stream. Refer to Section 5 for details of the PIM output stream.

The input file **path_nam.txt** must be present in the directory in which PIM is run.

Refer to Section 4 for details of the PIM input file **path_nam.txt**.

When running PIM, it is recommended that you use the same root name for the PIM input stream file, PIM output stream file, and PIM output file so that you can later associate the inputs and outputs of a PIM run.

An Example

Suppose that you want to test your installation of PIM by running PIM with the input stream file **testcas1.in** provided for test case 1. Also suppose that you wish to save the output stream to a file named **testcas1.log**. The name of the output file defined in the input stream file for this case is **testcas1.out**.

In DOS, Windows NT, and UNIX platforms, a typical command line to run PIM for this case would be:

pim < testcas1.in > testcas1.log

In the VMS platform, a typical command line to run PIM for this case would be:

run pim.exe /input=testcas1.in /output=testcas1.log

Figure 6.1 shows an example of a path_nam.txt input file for DOS and Windows NT platforms. Figures 6.2, 6.3, and 6.4 in Appendix G show respectively the input stream **testcas1.in**, output stream **testcas1.log**, and output file **testcas1.out** for this case.

Background of PIM

(The following article from the journal *Radio Science* gives background and details on PIM. It has been included verbatim with the permission of the authors, with some reformatting.)

PIM: A global ionospheric parameterization based on first principles models

R. E. Daniell,[1] Jr., L. D. Brown,[1] D. N. Anderson,[2] M. W. Fox,[3] P. H. Doherty,[4] D. T. Decker,[4] J. J. Sojka,[5] and R. W. Schunk[5]

Abstract. We describe a parameterized ionospheric model (PIM), a global model of theoretical ionospheric climatology based on diurnally reproducible runs of four physics based numerical models of the ionosphere. The four numerical models, taken together, cover the *E* and *F* layers for all latitudes, longitudes, and local times. PIM consists of a semianalytic representation of diurnally reproducible runs of these models for low, moderate, and high levels of both solar and geomagnetic activity and for June and December solstice and March equinox conditions. PIM produces output in several user selectable formats including global or regional latitude/longitude grids (in either geographic or geomagnetic coordinates), a set of user-specified points (which could lie along a satellite orbital path), or an altitude/azimuth/elevation grid for a user-specified location. The user selectable output variables include profile parameters (f_oF_2, h_mF_2, total electron content, etc.), electron density profiles, and ion composition (O^+, NO^+, and O_2^+).

1. Introduction

For the past 7 years, we have been developing a real-time ionospheric specification model for the Air Force Air Weather Service (AFAWS) for use at the Air Force Space Forecast Center (AFSFC, also known as the 50th Weather Squadron). This model, PRISM (Parameterized Real-time Ionospheric Specification Model) uses both ground based and space based data to update a climatological model in near real time. The climatological model, a parameterized ionospheric model known as PIM, is a composite of diurnally reproducible runs of several physical ionospheric models: (1) the time dependent ionospheric model (TDIM) of Utah State University (USU) [*Schunk*, 1988], (2) the low latitude F region model (LOWLAT) developed by *Anderson* [1973], (3) the midlatitude version of LOWLAT (called MIDLAT) developed by D. N. Anderson and modified by D. T. Decker, and (4) an E region local chemistry code (ECSD) developed by D. T. Decker and incorporating photoelectrons using the continuous slowing down method [*Jasperse*, 1982]. Thus unlike previous empirical models (such as the International Reference Ionosphere, IRI) that are based on empirical climatology, PIM is based on theoretical climatology.

We have made PIM available to the ionospheric community on an informal basis for several years, and the feedback from PIM users has resulted in substantial improvements. The purpose of this paper is to provide a description of PIM and to make it generally available to interested users. PRISM will be made available to the ionospheric research community through the AFAWS and will be described in a future publication.

PIM can produce output in three formats: (1) gridded output on a regional or global grid in geographic or geomagnetic latitude and longitude, (2) output at a set of user specified points in either geographic or geomagnetic coordinates, and (3) output on a grid of altitude, azimuth, and elevation from a user specified point. The second option may be used to specify output along a satellite orbital track, but the output is at fixed UT. The output data is of two types: electron density and ion composition profiles, and profile parameters (f_oF_2, h_mF_2, total electron content (TEC), etc.). The user may select either type or both.

Our primary objective in the development of PIM was to produce a convenient summary of the output of physics-based numerical models for a variety of geophysical conditions ("theoretical climatology"). We believe that for many applications this approach has significant advantages over empirical models. First, empirical models are limited by the amount and kind of available data. For example, the original CCIR coefficients [*International Radio*

Consultative Committee (CCIR), 1967] were obtained from monthly median values of f_oF_2 during 1954-1958 from approximately 150 ionosondes around the world. Since these ionosondes were sparsely distributed in ocean areas and in the southern hemisphere, the resulting maps of f_oF_2 were of limited value in those regions. *Rush et al.* [1983, 1984] improved the maps by supplementing the data with theoretical calculations using LOWLAT. The result was an improved set of coefficients [*Fox and McNamara,* 1988], generally known as the URSI-88 coefficient set [*Rush et al.*, 1989]. However, these coefficients are based on monthly median values organized in terms of solar activity [low and high]. As *Klobuchar and Doherty* [1992] have demonstrated, the daily variation of the ionosphere (especially the *F* region) is poorly correlated with the daily variation of solar activity as tracked by indices such as $F_{10.7}$ or sunspot number. This is probably due to the large daily variability of thermospheric winds, which in turn is due partly to daily variations in geomagnetic activity and partly to daily variability of gravity wave sources in the lower atmosphere. The net result is that empirical models average data over very different ionospheric conditions corresponding to the same solar activity level. Thus ionospheric features that move around on a daily basis will be smeared out or broadened and reduced in amplitude. Any empirical model which does not organize the data in terms of all of the driving forces that govern the ionosphere will have this property.

The principle distinction between empirical climatology and theoretical climatology may be stated as follows:

Empirical climatology yields an "average" ionosphere in which the average may be taken over very different ionospheric configurations. Persistent features such as the subauroral trough, auroral oval, or equatorial anomaly may be smeared out or broadened as a result of the averaging process. Empirical climatology is limited by the amount of data and the spatial and temporal distribution of that data.

Theoretical climatology yields a "representative" ionosphere, i.e., an ionosphere that corresponds to a potentially realizable set of specific geophysical conditions. Ionospheric features will have locations, widths, amplitudes similar to those that might be observed on any given day under the specified geophysical conditions. Theoretical climatology is limited by the accuracy and completeness of the physics and chemistry included in the theoretical models on which it is based and the computer resources required to span the full range of geophysical conditions.

For many purposes, the average ionosphere of an empirical model is all that is required. However, many users have the need to simulate the performance of an operational system under representative conditions. For those users, a representative ionosphere is more useful than an average ionosphere. PIM was designed for just this purpose.

2. The Physical Models

Four separate physical models were used as the basis of PIM: (1) a low-latitude *F* layer model (LOWLAT), (2) a midlatitude *F* layer model (MIDLAT), (3) a combined low and middle latitude E layer model (ECSD), and (4) a high-latitude *E* and *F* layer model (TDIM). All four models are based on a tilted dipole representation of the geomagnetic field and a corresponding geomagnetic coordinate system. (Hereafter, "latitude" means "geomagnetic latitude" unless otherwise noted.) All four models use the MSIS-86 neutral atmosphere model [*Hedin*, 1987]. Chemical reaction rates, collision frequencies, and similar data are consistent among all the models.

2.1 The Low-Latitude *F* Layer Model

The low latitude *F* region model (LOWLAT) was originally developed by *Anderson* [1973]. (See also *Moffett* [1979]). It solves the diffusion equation for O^+ along a magnetic flux tube. Normally, the entire flux tube is calculated with chemical equilibrium boundary conditions at both feet of the flux tube. A large number of flux tubes must be calculated in order to build up an altitude profile.

Since heat transport is not included in this model, ion and electron temperature models must be used. For the PRISM development effort we chose the temperature model of *Brace and Theis* [1981] with altitude interpolation based on the analytic forms of *Strobel and McElroy* [1970]. The Horizontal Wind Model (HWM) of *Hedin* [1988] was used to describe thermospheric winds. (Most of the model runs were made well before the latest version [*Hedin*, 1994] became available.)

The critical feature incorporated in the low-latitude model is the dynamo electric field. The horizontal component of this field drives upward convection through $\mathbf{E}\times\mathbf{B}$ drift, and this can significantly modify profile shapes and densities. This phenomenon is responsible for the equatorial anomaly, crests in ionization on either side of the magnetic equator at $\pm 15-20^\circ$ magnetic latitude. In the current version of PIM (version 1.3) the $\mathbf{E}\times\mathbf{B}$ driven vertical drift used for these calculations was based on the empirical models derived from data from the Atmospheric Explorer-E (AE-E) satellite [*Fejer et al.*, 1995], which are consistent with the drifts measured at Jicamarca [*Fejer*,

1981; *Fejer et al.*, 1989] but include longitudinal variations as well. We used the *Fejer et al*. [1995] empirical drifts for high solar activity. Following their discussion, we modified these drifts by reducing or eliminating the prereversal enhancement for moderate or low solar activity. In all cases, the published drift model was modified to ensure no net vertical motion after 24 hours, as is necessary for diurnally reproducible runs. Horizontal drifts were neglected in the PRISM runs.

Since its original development this model has undergone extensive validation by comparison with data. The most recent such comparison is by *Preble et al*. [1994], who used electron density profiles measured by the incoherent scatter radar facility at Jicamarca, Peru.

2.2 The Midlatitude F Layer Model

The midlatitude *F* region model (MIDLAT) is the same as the low latitude version except that the dynamo electric field is not included. Complete flux tubes are followed, but neither horizontal nor vertical convection is included. The computer resource requirements of MIDLAT are far less than those of LOWLAT. As long as the boundary between low and middle latitudes is chosen so that the electric field is negligible on the boundary flux tubes, the two models give identical results at the boundary, ensuring continuity across that boundary. For the PRISM development effort we used the same temperature model [*Brace and Theis*, 1981] and the same thermospheric wind model [*Hedin*, 1988]. For appropriate production, loss, and diffusion rates for both LOWLAT and MIDLAT, see *Decker et al.* [1994].

2.3 The Low and Midlatitude E Layer Model

The low- and midlatitude *E* region model (ECSD) was developed by D. T. Decker and J. R. Jasperse and incorporates photoelectrons calculated using the continuous slowing down (CSD) approximation [*Jasperse*, 1982]. Ion concentrations are calculated assuming local chemical equilibrium. A small nighttime source is included to ensure that an *E* layer is maintained throughout the night.

2.4 The High Latitude Model

The high latitude model (incorporating both *E* and *F* layers) is the Utah State University (USU) time dependent ionospheric model (TDIM). (See *Schunk* [1988] for a review.) This model is similar to the low- and midlatitude models except that the flux tubes are truncated and a flux boundary condition is applied at the top. In addition, the flux tubes move under the influence of the high latitude convection electric field. In the low latitudes, because the magnetic field is mainly horizontal, the effect of the electric field is primarily to move the ionization in altitude. In contrast, the high-latitude magnetic field is mainly vertical, and the electric field driven convection is horizontal. Like LOWLAT, this model has a long history and has been validated by numerous comparisons with data [e.g., *Sojka et al.*, 1994].

TDIM includes an *E* layer model that incorporates the effects of ionization by precipitating auroral particles. The ion production rates used were calculated using the B3C electron transport code [*Strickland et al*., 1976, 1994] and incident electron spectra representative of DMSP SSJ/5 data. The characteristics of the electron spectra were taken from the *Hardy et al*. [1987] electron precipitation model. The high-latitude convection patterns were those developed by *Heppner and Maynard* [1987] for southward directed B_z. Until high-latitude convection under northward B_z is better understood, we suggest using PIM with low *Kp* for B_z north conditions.

3. Parameterization of the Physical Models

Parameterization of the physical models proceeded in two steps. First, the models were used to generate a number of "databases" for a discrete set of geophysical conditions. Each database consists of ion density profiles on a discrete grid of latitudes and longitudes for a 24-hour period in UT. Second, to reduce storage requirements, the databases were approximated with semianalytic functions. These two processes are described in the following subsections.

3.1 Geophysical Parameters

All the physical models were parameterized in terms of season and solar activity. The mid- and high-latitude models were also parameterized in terms of magnetic activity, while the high latitude model was additionally

parameterized in terms of the sign of the interplanetary magnetic field component B_y. (The high-latitude model was only run using B_z southward. Northward B_z conditions are modeled using the low magnetic activity databases.) For the middle and low latitudes, the *F* layer (O^+) and the *E* layer (NO^+ and O_2^+) were computed and parameterized separately. The high-latitude model (TDIM) produced all three ions simultaneously.

Due to time and computer resource limitations, only a few values of each parameter were used. The season "values" are the June and December solstices and the March equinox (which also stands in for the September equinox). We expect to change from seasonal to monthly values in the next major version of PIM. The values of the other parameters are summarized for each latitude region in Table 1. Note that the USU TDIM and LOWLAT models produce output in magnetic local time (MLT), while MIDLAT and ECSD produce output in magnetic longitude.

Table 1. Geophysical Parameter Values

	Solar Activity, $F_{10.7}$	Magnetic Activity, *Kp*	IMF B_y Direction	Number of databases
Low Latitude F layer	70, 130, 210	N/A	N/A	36[a]
Midlatitude F layer	70, 130, 210	1, 3.5, 6	N/A	54[b]
Low & Midlatitude E layer	70, 130, 210	1, 3.5, 6	N/A	54[c]
High Latitude E & F layer	70, 130, 210	1, 3.5, 6	+, -	324[d]

N/A, not applicable.

[a]Three seasons times three solar activities times four longitude sectors.

[b]Three seasons times three solar activities times three magnetic activities times two hemispheres.

[c]Three seasons times three solar activities times three magnetic activities times two species.

[d]Three seasons times three solar activities times three magnetic activities times two B_y's times three species times two hemispheres.

3.2 Representation of the Databases

When the models are run for any one set of geophysical parameters (e.g., June, $F_{10.7}$ = 130, *Kp* = 1), they produce ion densities (O^+, NO^+, and O_2^+) on a four-dimensional grid. MIDLAT and ECSD use a grid of magnetic latitude (λ), magnetic longitude (φ), altitude (*z*), and universal time (τ). TDIM uses magnetic local time (MLT or ψ) instead of magnetic longitude, while LOWLAT uses MLT instead of UT. In order to make this mass of numbers more manageable, we produced a semianalytical representation of each database. The space and time grid parameters are summarized for each latitude region in Table 2.

Table 2. Horizontal Grid Parameters

Latitude Region	Magnetic Latitude	Magnetic Longitude	UT	Altitude Profiles per Database
Low-latitude *F* layer	−32° to 32° in 2° steps	80°, 180°, 260°, and 320°	MLT: 0.0 to 23.5 in 0.5 hour steps	1,584
Midlatitude *F* layer	30° to 74° and −30° to −74° in 4° steps	0° to 345° in 15° steps	0100 to 2300 in 2 hour steps	3,456
Low- and midlatitude *E* layer	−76° to 76° in 4° steps	0° to 345° in 15° steps	0100 to 2300 in 2 hour steps	11,232
High-latitude *E* and *F* layer	51° to 89° and −51° to −89° in 2° steps	MLT: 0.5 to 23.5 in 1 hour steps	0100 to 2300 in 2 hour steps	5,760

Because of the computer resource requirements of the low latitude *F* layer code, it was used to generate databases at four discrete longitudes (corresponding to longitude sectors for which **E**×**B** drift measurements were available). Each longitude sector was parameterized separately, and the necessary longitude interpolation is carried out in PIM and PRISM during execution, as described below.

Because we were trying to represent discrete data (rather than continuous functions), and because we were working with regional rather than global data sets, we felt that the usual spherical harmonic expansion techniques were not appropriate. Instead we concentrated on the use of orthogonal functions of discrete variables.

We first considered the use of modified Chapman functions for representing altitude profiles of ion densities. These functions have the advantage that peak height and peak density are explicit parameters, but the extremely nonlinear nature of these functions necessitates the use of nonlinear least squares fitting methods. While such methods produced excellent representations of individual profiles, the variation of the fitted parameters with latitude, longitude (or MLT), and UT was unacceptably noisy. Consequently, we chose to use empirical orthonormal functions for the altitude representation.

Empirical orthonormal functions (EOFs) have been used extensively to represent meteorological and climatological data [*Lorenz*, 1956; *Kutzbach*, 1967; *Davis*, 1976; *Peixota and Oort*, 1991]. They have also been used for empirical ionospheric modeling [*Secan and Tascione*, 1984] (EOFs are described in Appendix A). They have the advantage of providing a representation in terms of linear combinations of orthogonal functions, which allows for straightforward determination of coefficients. However, because peak density and peak height are not explicit parameters of the representation, these parameters can be determined only by reconstructing the entire profile and invoking a peak-finding algorithm. We expect to revisit this problem in future versions of PIM and PRISM and implement a new representation that combines the attractive features of both methods, that is, that includes peak density and peak height as explicit parameters yet relies on linear combinations of orthogonal functions to describe the profile shape.

For longitude (or local time) variations (and for the low-latitude *F* layer UT variation), the obvious choice is a Fourier series, since trigonometric functions retain their orthogonality properties on uniform discrete grids and because the data is periodic in the independent variable. These worked quite well for the high-latitude models under all conditions and for the low- and midlatitude models under low to moderate solar activity conditions. However, they did not work well for the low- and midlatitude models under high solar activity conditions, apparently because the EOF coefficients exhibited exceptionally large gradients at dawn and dusk. Therefore we decided to tabulate the coefficients in longitude for all the low- and midlatitude databases.

For the latitude variations we chose to generate grid-specific orthogonal polynomials using the algorithm derived by *Beckmann* [1973] and described in Appendix B. To help keep the notation straight, we summarize it in Table 3.

Table 3. Notation Summary

Grid	Variable	Variable Index	Orthogonal Function	Function Index
Altitude	z_i	$1 \le i \le I$	EOF: $g_m(z_i)$	$1 \le m \le M$
Latitude	λ_j	$1 \le j \le J$	polynomial: $u_n(\lambda)$	$0 \le n \le N$
Longitude	φ_k	$1 \le k \le K$	not used	
MLT	ψ_k, ψ_l	$1 \le k \le K$, $1 \le l \le L$	TDIM: $a_{mp}^{(s)}(\lambda_j, \tau_l)\cos(p\psi) + b_{mp}^{(s)}(\lambda_j, \tau_l)\sin(p\psi)$ all others: not used	$0 \le p \le P$
UT	τ_l	$1 \le l \le L$	not used	

The semianalytic representation of each database was generated in several steps. For all ionospheric regions the first step was the determination of the EOFs from the ion densities in the database and a set of coefficients

$c_{lm}(\lambda_j, \varphi_k)$ for representing each ion density profile on the latitude, longitude, UT grid (see Appendix A).

TDIM:

$$n_s(z_i, \lambda_j, \psi_k, \tau_l) \approx \sum_{m=1}^{M} c_m^{(s)}(\lambda_j, \psi_k, \tau_l) g_m^{(s)}(z_i) \qquad (1a)$$

MIDLAT, ECSD:

$$n_s(z_i, \lambda_j, \varphi_k, \tau_l) \approx \sum_{m=1}^{M} c_m^{(s)}(\lambda_j, \varphi_k, \tau_l) g_m^{(s)}(z_i) \qquad (1b)$$

LOWLAT:

$$n_s(z_i, \lambda_j, \varphi_k, \psi_l) \approx \sum_{m=1}^{M} c_m^{(s)}(\lambda_j, \varphi_k, \psi_l) g_m^{(s)}(z_i) \qquad (1c)$$

where z_i, λ_j, φ_k, ψ_k, τ_j, and ψ_l are all points on the model output grid, and $g_m^{(s)}(z_i)$ is the m^{th} EOF evaluated at z_i. (Note, however, that a different set of $g_m^{(s)}(z_i)$ functions are used for each ion, for each set of geophysical conditions, and for each model.)

For the high latitude model (TDIM, both *E*- and *F*-layers), the second step was the generation of Fourier coefficients in MLT, $a_{mp}^{(s)}(\lambda_j, \tau_l)$ and $b_{mp}^{(s)}(\lambda_j, \tau_l)$, for each point on the latitude, UT grid.

TDIM:

$$n_s(z_i, \lambda_j, \psi, \tau_l) \approx \sum_{m=1}^{M} \sum_{p=0}^{P} \left\{ a_{mp}^{(s)}(\lambda_j, \tau_l) \cos(p\psi) + b_{mp}^{(s)}(\lambda_j, \tau_l) \sin(p\psi) \right\} g_m^{(s)}(z_i) \qquad (2)$$

For the low- and midlatitude models, we found that a truncated Fourier series often introduced spurious longitudinal dependences, apparently driven by the steep gradients at dawn and dusk. The effect was particularly pronounced at high solar activity when the day/night contrast is the greatest. Consequently, for these models the EOF coefficients remain tabulated in longitude.

For all models the next step was the generation of orthogonal polynomials from the latitude grid (Appendix B). For the high latitude model (TDIM) the coefficients are $\alpha_{mnp}^{(s)}(\tau_l)$ and $\beta_{mnp}^{(s)}(\tau_l)$, and the ion density is approximated by

TDIM:

$$n_s(z_i, \lambda, \psi, \tau_l) \approx \sum_{m=1}^{M} \sum_{n=0}^{N} \sum_{p=0}^{P} \left\{ \alpha_{mnp}^{(s)}(\tau_l) \cos(p\psi) + \beta_{mnp}^{(s)}(\tau_l) \sin(p\psi) \right\} g_m(z_i) u_n(\lambda) \qquad (3a)$$

For MIDLAT and ECSD the coefficients are $\gamma_{mn}^{(s)}(\lambda_k, \tau_l)$ and the ion density is approximated by

MIDLAT, ECSD:

$$n_s(z_i,\lambda,\varphi_k,\tau_l) \approx \sum_{m=1}^{M}\sum_{n=0}^{N} \gamma_{mn}^{(s)}(\varphi_k,\tau_l) g_m(z_i) u_n(\lambda) \qquad (3b)$$

For LOWLAT the coefficients are $\eta_{mn}^{(s)}(\varphi_k,\psi_l)$ and the ion density is approximated by

LOWLAT:

$$n_s(z_i,\lambda,\varphi_k,\psi_l) \approx \sum_{m=1}^{M}\sum_{n=0}^{N} \eta_{mn}^{(s)}(\varphi_k,\psi_l) g_m(z_i) u_n(\lambda) \qquad (3c)$$

The number of terms in each series are listed in Table 4 for each region.

Table 4. Altitude Grids and Emperical Orthonormal Functions

Database	Altitude Points	Minimum Altitude (km)	Maximum Altitude (km)	EOFs
Low-latitude O^+	55	160	1600	9
Midlatitude O^+	49	125	1600	8
Low- and midlatitude NO^+ and O_2^+	28	90	400	7
High-latitude O^+, NO^+, & O_2^+	37	100	800	6

Note that in none of these cases was the altitude spacing uniform.

Because of the extensive use of tabulated coefficients, the ion density at an arbitrary point must be obtained by interpolation. In PIM and PRISM, altitude interpolation is quadratic, while UT interpolation is linear. For the MIDLAT databases, the longitude interpolation is also linear, as is the local time interpolation in the LOWLAT databases. However, the longitude interpolation in the LOWLAT databases is more complicated. First, the O^+ profile for the desired magnetic latitude and local time is reconstructed for each of the four longitude sectors. Then the peak height and peak density is determined for each profile. The peak height for the desired longitude is determined by Fourier interpolation, and all four profiles are shifted to match the interpolated peak height. Then Fourier interpolation is used again at each altitude to obtain the interpolated ion density profile.

3.3 Merging the Regional Models

Because we used four different regional models in the development of PRISM, the models must be merged at region boundaries. Specifically, the low latitude and midlatitude O^+ models have to be merged across the boundary between low and middle latitudes, while all three ions (O^+, NO^+, and O_2^+) must be merged across the boundary between midlatitudes and high latitudes.

The transition from low latitude O^+ profiles to midlatitude O^+ profiles takes place between 30° and 34° in both hemispheres. The transition is accomplished by taking a weighted average of the h_mF_2 values from the two models in which the weight shifts linearly from 100% low latitude at 30° to 100% midlatitude at 34°. The profiles are shifted to match the averaged h_mF_2 values and then a similar weighted average of the shifted profiles is taken to produce the final merged profile. No transition for NO^+ and O_2^+ is necessary since a single model was used for these ions.

The transition from midlatitude to high latitude takes place over an 8° wide zone extending from 50° to 58°. The transition process is similar to the low to midlatitude transition, except that the high-latitude profiles are shifted to

match the h_mF_2 and h_mE values given by the midlatitude models. The final profile is produced by a weighted average of midlatitude and (shifted) high latitude profiles.

Although PIM and PRISM use geomagnetic coordinates internally, they can produce output in either geomagnetic or geographic coordinates. The conversion from geomagnetic coordinates to geographic coordinates is made using corrected geomagnetic (CGM) coordinates. Although this is not entirely consistent with the dipole coordinates used in the physical models, it does result in a more realistic representation of magnetically controlled features when presented in geographic coordinates. A contour map of N_mF_2 in geographic coordinates (cylindrical projection) for the June solstice at high solar activity and moderate magnetic activity is displayed in Figure 1. The equatorial anomaly is clearly visible between geographic longitudes 30°E and 180°E, corresponding to local times of 1400 and 2400. The high latitude is more clearly seen in a polar projection such as is displayed in Figure 2, again in geographic coordinates. The figure shows the northern hemisphere at the December solstice. B_y is positive, and the tongue of ionization resulting from a steady convection pattern is clearly visible on the evening side.

Figure 1. Contours of N_mF_2 (in units of 10^5 cm^{-3}) in cylindrical equidistant projection from PIM for high solar activity, moderate magnetic activity, at 1200 UT near the June solstice. The equatorial anomaly is clearly evident from about 1400 to 2400 local time.

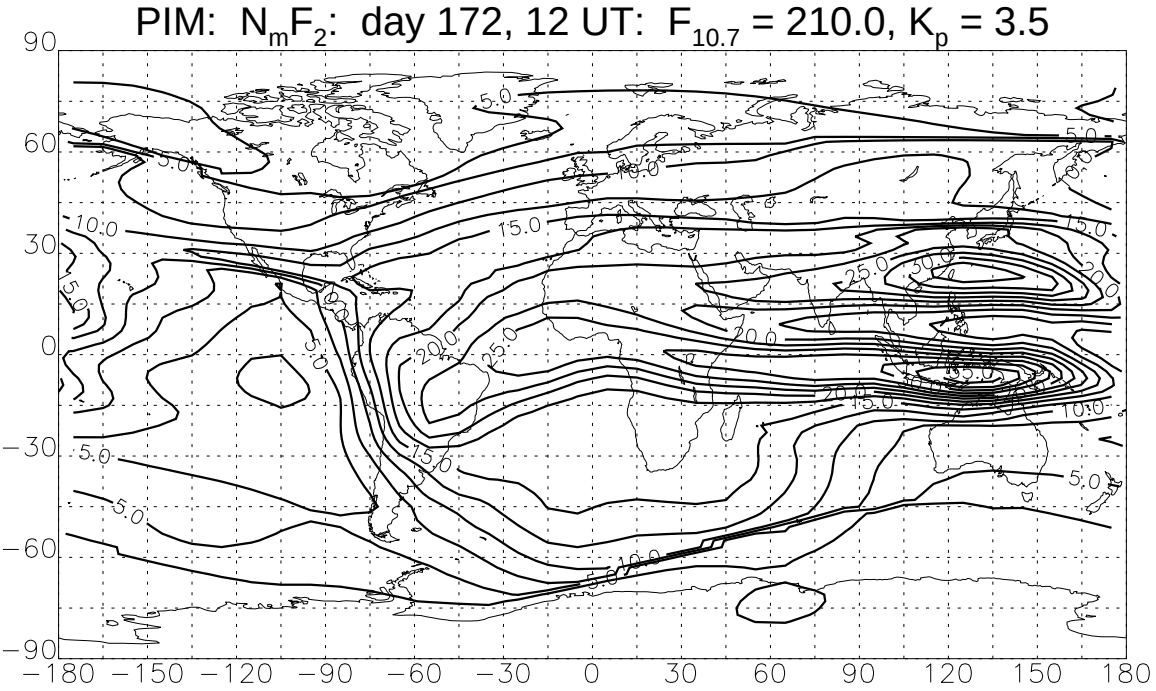
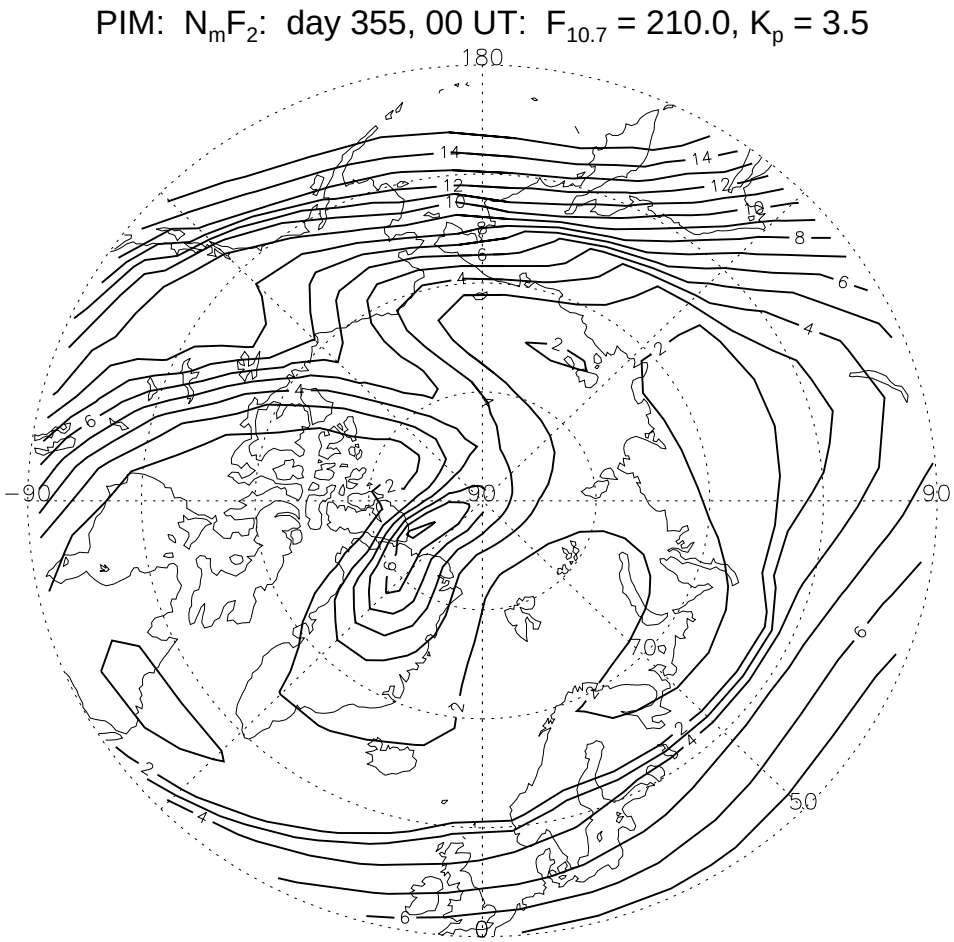


Figure 2. Contours of $N_m F_2$ (in units of 10^5 cm^{-3}) in polar projection from PIM for the same conditions as Figure 1 except for 0000 UT and December solstice. The "tongue of ionization" produced by a steady convection pattern is clearly evident. Local midnight is at the bottom.



4. Discussion

We have described PIM, a global ionospheric model based on theoretical climatology in the form of diurnally reproducible runs of a set of physics based, numerical ionospheric models. This model has been distributed to about 60 users around the world and is now available to the ionospheric community over the Internet. (Those interested in obtaining PIM should contact L. D. Brown, R. E. Daniell, or D. N. Anderson for more information.) For many users, the principal advantage of PIM over empirical models will be the more realistic representation of low- and high-latitude ionospheric features.

A number of compromises were required in the development of PIM. First, of course, was the necessity of using parameterizations (in the form of diurnally reproducible runs) of the physical models, rather than the physical models

themselves. Second, we had to use empirical models (e.g., MSIS-86) instead of physical models to provide the necessary inputs to the ionospheric models. Third, we had to use a tilted dipole representation of the Earth's magnetic field instead of a more realistic model. This last compromise is mitigated somewhat by the use of the corrected geomagnetic (CGM) coordinate system to convert from internal geomagnetic coordinates to geographic coordinates. While not fully self-consistent, this does allow a more realistic representation of geomagnetically controlled features such as the equatorial anomaly. A fourth compromise was the use of only three seasons. We expect to remove these compromises one by one as available computing power increases in the future.

The particular features described here apply to PIM version 1.3. Significant enhancements to PIM are planned for the near future. H^+ ion densities based on a parameterization of the plasmasphere model of *Bailey and Sellek* [1990] will be added so that PIM can give electron density profiles up to the plasmapause. At the same time the coefficient files will be regenerated using a single global ionospheric model, eliminating the need to merge models across region boundaries. Monthly coefficients (instead of seasonal) coefficients will also be used. The analytic representations will also be reexamined in order to produce a more accurate, more efficient, and fully analytic fit to the model runs. The resulting version of PIM should be even more useful to the ionospheric community than the current version.

Appendix A. Empirical Orthonormal Functions

This treatment of empirical orthogonal functions (EOFs) is based on the appendix of *Secan and Tascione* [1984], which was based on work by *Lorenz* [1956], *Kutzbach* [1967], and *Davis* [1976]. See also *Peixota and Oort* [1991]. The reader is referred to these references for mathematical proofs of the assertions made below. In the following discussion we use the notation given in Table 3.

A database consists of altitude profiles at certain longitudes, certain latitudes, and certain universal times. (See Tables 1-4.) Let *S* be the number of altitude profiles in a database, and let *I* be the number of points in each altitude profile. We would like to represent each altitude profile of the quantity Ψ (e.g., O^+concentration) as an expansion in orthogonal functions, $g_m(z_i)$:

$$\Psi_s(z_i) = \sum_{m=1}^{M} \alpha_{sm} g_m(z_i) + r_s(z_i), \qquad s = 1..S,\ i = 1..I \tag{A1}$$

where $r_s(z_i)$ is the residual, and the coefficients α_{sm} are calculated from

$$\alpha_{sm} = \sum_{i=1}^{I} \Psi_s(z_i) g_m(z_i) \tag{A2}$$

In principle, any orthogonal set of functions may be used. However, the references cited above provide an algorithm for finding the set which minimizes the RMS error for a given number of terms, $M \le I$. We summarize the algorithm here.

First define the $I \times I$ covariance matrix **C** with elements

$$C_{ij} = \frac{1}{S} \sum_{s=1}^{S} \Psi_s(z_i) \Psi_s(z_j), \qquad i, j = 1,2,...I \tag{A4}$$

Now consider the eigenvalue/eigenvector problem $\mathbf{C}\varphi = \varphi\mathbf{L}$ or

$$\sum_{j=1}^{I} C_{ij} \varphi_{jk} = \sum_{j=1}^{I} \varphi_{ij} \delta_{jk} \lambda_k = \varphi_{ik} \lambda_k \tag{A5}$$

where $\mathrm{j} = \{\varphi_{ij}\}$ is the matrix of eigenvectors of $\mathbf{C} = \{C_{ij}\}$, and $\mathbf{L} = \{\delta_{ij}\lambda_j\}$ is a diagonal matrix whose elements are the corresponding eigenvalues. (The k^{th} column of φ is the eigenvector corresponding to the k^{th} eigenvalue, λ_k.) By convention, the eigenvectors and eigenvalues are ordered so that $\lambda_1 > \lambda_2 > ... > \lambda_I$. Because **C**

is a real symmetric matrix, eigenvectors corresponding to unique eigenvalues are guaranteed to be orthogonal [e.g., *Hildebrand*, 1965]. Because of the origin of the matrix $\mathbf{C}$, it is unlikely that any of its eigenvalues will be degenerate, so we may assume that φ is an orthogonal set. According to *Secan and Tascione* [1984] and references therein, the set of orthogonal functions that minimizes the RMS error for *M* terms is just the first *M* eigenvectors:

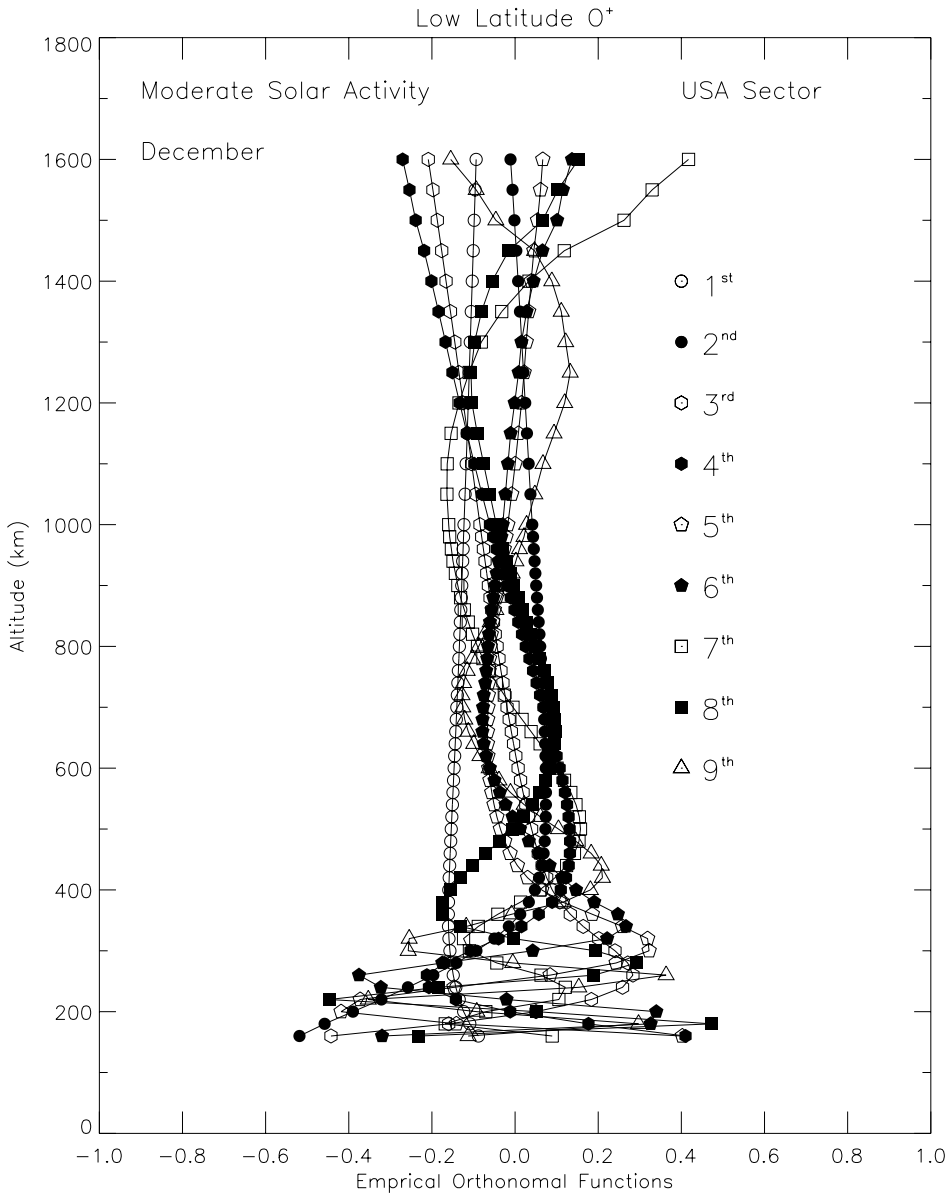
$$g_m(z_i) = \varphi_{im}, \qquad i = 1,2,\ldots I;\ m = 1,2,\ldots M \qquad \text{(A6)}$$

These are the EOFs.

As a practical matter, we have found that the number of EOFs needed to provide a reasonably good representation for all the profiles is about *I*/6, as illustrated in Table 4. The only exception is the low- and midlatitude *E* layer (NO^+ and O_2^+), probably because these databases covered both hemispheres simultaneously. We have also found that substantial improvement in representation does not occur until the number of EOFs is about *I*/2. Furthermore, the EOFs derived for one database were inadequate for any other database, and the EOFs simultaneously derived from several databases produce noticeably poorer representations than those derived for each database individually. Consequently, we have derived separate EOF sets for each database.

The first nine EOF's derived from the low latitude F region (O^+) database for the US longitude sector, the December solstice, and moderate solar activity, are shown in Figure A1. The first EOF always has the least structure, and successive EOF's become progressively more structured. Although differing in detail, the EOF's for the other databases are qualitatively similar.

Figure A1. The empirical orthonormal functions (EOFs) for low latitude O^+ derived from the LOWLAT output databases for the American longitude sector, December solstice, moderate magnetic activity, and moderate solar activity. Only the first nine EOF's are plotted because these are the ones used in PIM.



Appendix B. Orthogonal Polynomials of Discrete Variables

Because the databases for which we desire analytic approximations have discrete latitude grids, we preferred to use polynomials whose orthogonality is defined in terms of that grid, rather than in terms of integrals over the interval. The algorithm for generating orthogonal polynomials on a specified grid is given by *Beckmann* [1973]. Let us denote the desired polynomials by $u_n(\lambda)$ and define $u_{-1}(\lambda) \equiv 0$ and $u_0(\lambda) \equiv 1$. Note that the polynomials are continuous functions of the continuous variable λ even though their orthogonality is defined in terms of the discrete grid $\{\lambda_j, j = 1,2,\ldots J\}$. The recursion relation for the polynomials is

$$u_{n+1}(\lambda) = (\lambda - B_n)u_n(\lambda) - \frac{h_n^2}{h_{n-1}^2}u_{n-1}(\lambda) \qquad \text{(B1)}$$

where the norms h_n are given by

$$h_n = \sum_{j=1}^{J} u_n^2(\lambda_j) \qquad \text{(B2)}$$

and the recursion constants B_n are given by

$$B_n = \frac{1}{h_n^2} \sum_{j=1}^{J} \lambda_j u_n(\lambda_j) \qquad \text{(B3)}$$

For example, the polynomials generated by this algorithm may be used to represent the latitude variations of the Fourier coefficients a_{mp} and b_{mp} used in the high-latitude representation:

$$a_{mp}(\lambda_j, \tau_l) = \sum_{n=1}^{N} \alpha_{mnp}(\tau_l) u_n(\lambda_j) \qquad \text{(B4)}$$

$$b_{mp}(\lambda_j, \tau_l) = \sum_{n=1}^{N} \beta_{mnp}(\tau_l) u_n(\lambda_j) \qquad \text{(B5)}$$

where

$$\alpha_{mnp}(\tau_l) = \frac{1}{h_n^2} \sum_{j=1}^{J} a_{mp}(\lambda_j, \tau_l) u_n(\lambda_j) \qquad \text{(B6)}$$

$$\beta_{mnp}(\tau_l) = \frac{1}{h_n^2} \sum_{j=1}^{J} b_{mp}(\lambda_j, \tau_l) u_n(\lambda_j) \qquad \text{(B7)}$$

Acknowledgments. Much of the development of PIM and PRISM was supported by the Geophysics Directorate of Phillips Laboratory through contracts with Computational Physics, Inc. (F19628-92-C-0044) and Boston University (F19628-K-90-0014).

References

Anderson, D. N., A theoretical study of the ionospheric *F*-region equatorial anomaly, II, Results in the American and Asian sectors, *Planet. Space. Sci., 21*, 421-442, 1973.

Bailey, G. J., and R. Sellek, A mathematical model of the Earth's plasmasphere and its application in a study of He^+ at L = 3, *Ann. Geophys., 8*, 171, 1990.

Beckmann, P., *Orthogonal Polynomials for Engineers and Physicists*, pp. 91092, Golem Press, Boulder, Colo., 1973.

Brace, L. H., and R. F. Theis, Global empirical models of ionospheric electron temperature in the upper F-region and plasmasphere based on in situ measurements from the Atmosphere Explorer-C, ISIS 1, and ISIS 2 satellites, *J. Atmos. Terr. Phys., 43*, 1317, 1981.

Davis, R. E., Predictability of Sea Surface Temperature and Sea Level Pressure Anomalies Over the North Pacific Ocean, *J. Phys. Oceanogr.*, 6, 249, 1976.

Decker, D. T., C. E. Valladares, R. Sheehan, Su. Basu, D. N. Anderson, and R. A. Heelis, Modeling daytime *F* layer patches over Sondrestrom, *Radio Sci., 29*, 249-268, 1994.

Fejer, B. G., The equatorial ionospheric electric fields, a review, *J. Atmos. Terr. Phys., 43*, 377-386, 1981.

Fejer, B. G., E. R. de Paula, I. S. Batista, E. Bonelli, and R. F. Woodman, Equatorial *F* region vertical plasma drifts during solar maxima, *J. Geophys. Res., 94*, 12,049-12,054, 1989.

Fejer, B. G., E. R. de Paula, R. A. Heelis, and W. B. Hanson, Global equatorial ionospheric vertical plasma drifts measured by the AE-E satellite, *J. Geophys. Res., 100*, 5769-5776, 1995.

Fox, M. W., and L. F. McNamara, Improved world-wide maps of monthly median f_oF_2, *J. Atmos. Terr. Phys., 50*, 1077, 1988.

Hardy, D. A., M. S. Gussenhoven, R. R. Raistrick, and W. J. McNeil, Statistical and functional representations of the pattern of auroral energy flux, number flux, and conductivity, *J. Geophys. Res., 92*, 12,275-12,294, 1987.

Hedin, A. E., MSIS-86 Thermospheric Model, *J. Geophys. Res., 92*, 4649-4662, 1987.

Hedin, A. E., Empirical global model of upper thermosphere winds based on Atmospheric and Dynamics Explorer satellite data, *J. Geophys. Res., 93*, 9959-9978, 1988.

Hedin, A. E., M. J. Buonsanto, M. C. Codrescu, M.-L. Duboin, C. G. Fesen, M. E. Hagan, K. L. Miller, and D. P. Sipler, Solar activity variations in midlatitude thermospheric meridional winds, *J. Geophys. Res.*, *99*, 17,601-17,608, 1994.

Heppner, J. P., and N. C. Maynard, Empirical high-latitude electric field models, *J. Geophys. Res*., *92*, 4467-4489, 1987.

Hildebrand, F. B., *Methods of Applied Mathematics*, pp. 30-34, Prentice-Hall, Englewood Cliffs, N. J., 1965.

International Radio Consultative Committee (CCIR), CCIR atlas of ionospheric characteristics, *Rep. 340-4*, Int. Telecommun. Union, Geneva, 1967.

Jasperse, J. R., The photoelectron distribution function in the terrestrial ionosphere, in *Physics of Space Plasmas*, edited by T. S. Chang, B. Coppi, and J. R. Jasperse, pp. 53-84, Sci. Publ., Cambridge, Mass., 1982.

Klobuchar, J. A., and P. H. Doherty, The correlation of daily solar flux values with total electron content, paper presented at the International Beacon Satellite Symposium, Plasma Fusion Center, Mass. Inst. of Technol., Cambridge, July 6-10, 1992.

Kutzbach, J. E., Empirical eigenvectors of sea-level pressure, surface temperature, and precipitation complexes over North America, *J. Appl. Meteorol*., *6*, 791, 1967.

Lorenz, E. N., *Empirical orthogonal functions and statistical weather prediction*, Sci. Rep. 1, Contract AF19(604)1566, AFCRC-TN-57-256, Dept. Meteorol., Mass. Inst. of Technol., Cambridge, 1956.

Moffett, R. J., The equatorial anomaly in the electron distribution of the terrestrial F-region, *Fundam. of Cosmic Phys.*, *4*, 313-391, 1979.

Peixota, J. P., and A. H. Oort, *Physics of Climate*, Appendix B, Am. Inst. of Phys., New York, 1991.

Preble, A. J., D. N. Anderson, B. G. Fejer, and P. H. Doherty, Comparison between calculated and observed *F* region electron density profiles at Jicamarca, Peru, *Radio Sci.*, *29*, 857-866, 1994.

Rush, C. M., M. PoKempner, D. N. Anderson, F. G. Stewart, and J. Perry, Improving ionospheric maps using theoretically derived values of f_oF_2, *Radio Sci.*, *18*, 95, 1983.

Rush, C. M., M. PoKempner, D. N. Anderson, J. Perry, F. G. Stewart, and R. Reasoner, Maps of f_oF_2 derived from observations and theoretical data, *Radio Sci*., *19*, 1083, 1984.

Rush, C. M., M. Fox, D. Bilitza, K. Davies, L. McNamara, F. Stewart, and M. PoKempner, Ionospheric mapping: An update of f_oF_2 coefficients, *Telecommun. J.*, *56*, 179-182, 1989.

Schunk, R. W., A Mathematical Model of the Middle and High Latitude Ionosphere, *Pure Appl. Phys.*, *127*, 255-303, 1988.

Secan, J. A., and T. F. Tascione, The 4D Ionospheric Objective Analysis Model, paper presented at the 4th Ionospheric Effects Symposium, Naval Research Laboratory, Springfield, Va., May 1-3, 1984.

Sojka, J. J., R. W. Schunk, and W. F. Dennig, Ionospheric response to the sustained high geomagnetic activity during the March '89 great storm, *J. Geophys. Res.*, *99*, 21,341-21,352, 1994.

Strickland, D. J., D. L. Book, T. P. Coffey, and J. A. Fedder, Transport equation techniques for the deposition of auroral electrons, *J. Geophys. Res*., *81*, 2755-2764, 1976.

Strickland, D. J., R. E. Daniell, J. R. Jasperse, and B. Basu, Transport-theoretic model for the electron-proton-hydrogen atom aurora, 2, Model results, *J. Geophys. Res*., *98*, 21,533-21,548, 1993.

Strobel, D. F., and M. B. McElroy, The F2-layer at middle latitudes, *Planet. Space Sci.*, *18*, 1181, 1970.

D. N. Anderson, PL/GPIM, 29 Randolph Road, Hanscom AFB, MA 01731-3010. (e-mail: danderson@plh.af.mil)

L. D. Brown and R. E. Daniell, Jr., Computational Physics, Inc., Suite 202A, 240 Bear Hill Road, Waltham, MA 02154-1126. (e-mail: brown@plh.af.mil; daniell@plh.af.mil)

D. T. Decker and P. H Doherty, Institute for Space Research, Boston College, Chestnut Hill, MA 02159. (e-mail: decker@plh.af.mil; doherty@plh.af.mil)

M. W. Fox, Center for Space Physics, Boston University, Boston, MA 02215. (e-mail: matthewf@spica.bu.edu)

R. W. Schunk and J. J. Sojka, Center for Atmospheric and Space Science, Utah State University, Logan UT 84322-4405.

(Received May 4, 1994; revised June 7, 1994; accepted June 14, 1995.)

[1]Computational Physics, Incorporated, Waltham, Massachusetts.

[2]Geophysics Directorate, Phillips Laboratory, Hanscom Air Force Base, Massachusetts.

[3]Center for Space Physics, Boston University, Boston, Massachusetts.

[4]Institute for Space Research, Boston College, Chestnut Hill, Massachusetts.

[5]Center for Atmospheric and Space Science, Utah State University, Logan.

Paper number 95RS01826.
0048-6604/95/95RS-01826$08.00

Code History

(The following text from internal CPI memorandums describes incremental changes to PIM since version 1.0.8. It has been included verbatim with the permission of the author, with some reformatting. However, the tables in the memorandums that detail changes to the PIM source code have been excluded).

Changes to PIM 1.6 for PIM 1.7
13-January-1998

The changes to PIM 1.6 for PIM 1.7 focus on replacing the LLF and MLF parameterized models. The changes are summarized as follows:

1. **BUG FIX**: A minor bug in the selection of magnetic latitudes for interpolation in the MLF parameterized model has been fixed. This bug should not have affected past results.

2. **BUG FIX**: A minor bug in the calculation of scale height for the plasmaspheric extrapolation of E-layer and F-layer densities has been fixed. This bug should not have affected past results.

3. In order to eliminate problems in merging the LLF and MLF parameterized models due to differences between the two theoretical models (LOWLAT and MIDLAT) previously used as their basis, the LLF and MLF parameterized models have been regenerated from a single theoretical model (LOWLAT). In addition, the following improvements have been made to LOWLAT:

    a. The equatorial vertical E×B drift and its radial derivative now vary smoothly to zero in the drift transition region. Previously, a simple linear fall-off was used, resulting in discontinuities in the radial derivative at the drift transition region endpoints.

    b. The radial derivative of the equatorial vertical E×B drift is now zero above the drift transition region. Previously, it was never set to zero above the transition region.

    c. The dayside and nightside electron temperatures now merge at 0600 and 1800 solar local time. Previously, the electron temperature was discontinuous at 0600 and 1800 solar local time.

    d. The neutral wind is now calculated for the correct geographic longitude at all points along a field line. Previously, a fixed geographic longitude was used, resulting in an error in the neutral wind due to the magnetic declination of the field line.

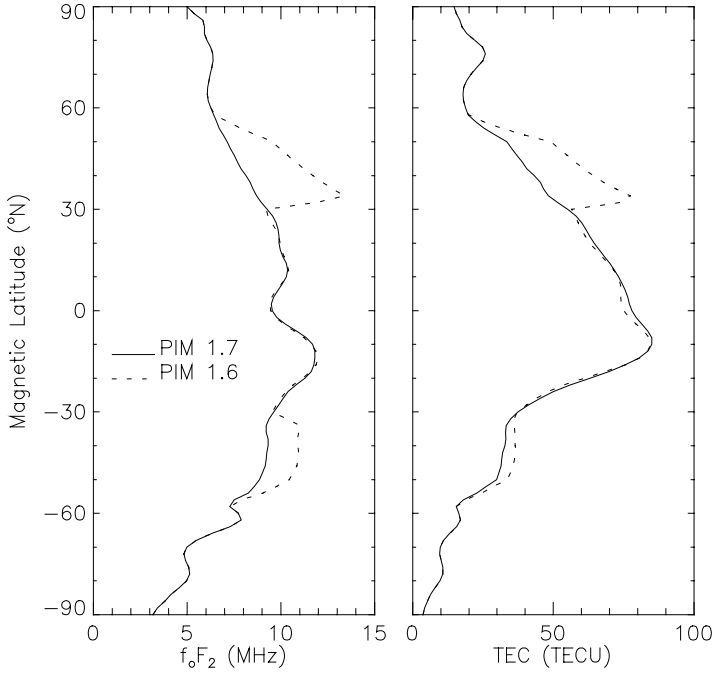
4. The LLF parameterized model has been extended to 44° in absolute latitude, in order to broaden the merge region between the LLF and MLF parameterized models. Previously, the LLF parameterized model only went up to 34° in absolute latitude.

5. The lower absolute latitude boundary of the MLF parameterized model is now 34°, in order to reduce the error due to the assumption of verticality of the midlatitude field lines and to make sure that the midlatitude field lines are outside the region of vertical E×B drift. Previously, the lower absolute latitude boundary of the MLF parameterized model was 30°.

6. The merge region for the LLF and MLF parameterized models has been broadened to the absolute latitude range 34°-44°, in order to improve the quality of the merge. Previously, the range was 30°-34°, too narrow to effectively merge the two parameterized models.

7. Extrapolation of the parameterized models in $F_{10.7}$ above 210 is now allowed. Previously, no extrapolation in $F_{10.7}$ above 210 was allowed, effectively capping $F_{10.7}$ at 210.

The plots of f_oF_2 and TEC vs. magnetic latitude below illustrate the improvement in PIM 1.7 regarding the agreement and merging of the LLF and MLF parameterized models. Notice the discontinuities in f_oF_2 and TEC at the LLF/MLF merge region of PIM 1.6 (±30°-34° magnetic latitude), and their absence in PIM 1.7. The plots were generated by PIM for the following conditions: year 1981, day of the year 173, Universal Time 0000, $F_{10.7}$ 210, K_p 3.5, IMF B_y positive, IMF B_z negative, 270°E magnetic longitude, no URSI f_oF_2 normalization, and no CCIR f_oE normalization.



Changes to PIM 1.5 for PIM 1.6
14-February-1997

The changes to PIM 1.5 for PIM 1.6 focus on adding a plasmasphere to PIM. The changes are summarized as follows:

8. The empirical Gallagher plasmaspheric model has been incorporated into PIM. To enable the user to control the inclusion of the plasmasphere in the total electron density, a new parameter has been added to the end of the PIM input stream. For compatibility with pre-PIM 1.6 input streams, the absence of the new parameter defaults to excluding the plasmasphere from the total electron density rather than causing a fatal read error.

9. PIM's internal altitude grid now extends to 25,000 km. The E-layer and F-layer densities from PIM's parameterized regional models are extrapolated to plasmaspheric altitudes using gravitationally-corrected scale heights above 1400 km altitude. Positive scale heights are enforced to ensure that the E-layer and F-layer densities fall off with altitude above 1400 km. Vertical TEC calculated by PIM for output grid types 0 (rectangular latitude/longitude) and 1 (latitude/longitude pairs) is now defined in the altitude range 90 to 25,000 km instead of 90 to 1,600 km. Slant TEC calculated by PIM for output grid type 2 (azimuth/elevation) continues to be defined in the range specified by the output altitude grid.

10. The output altitude grid is now limited to the range 90 to 25,000 km instead of 90 to 1,600 km.

11. The record format of the PIM output file has not changed. However, the plasmasphere usage parameter has been encoded into the information in record 6 of the PIM output file, and the altitude display format used in the PIM output file has been widened to accommodate plasmaspheric altitudes.

12. A number of comments have been updated to reflect the replacement of the semi-empirical IRI f_oE model with the CCIR recommendation.

Changes to PIM 1.4 for PIM 1.5
30-September-1996

The changes to PIM 1.4 for PIM 1.5 focus on replacing the LLF coefficient set again, and on improving the merging of the parameterized model density profiles. The changes are summarized as follows:

1. **BUG FIX**: The LLF parameterized model coefficient set has been replaced again. Because of a bug in the processing of the LOWLAT output, the magnetic latitude grid in the PIM 1.4 LLF coefficient files was defined to start at -33°N instead of the correct value of -34°N. This resulted in an erroneous northward 1° magnetic latitude shift in the O^+ densities from the LLF parameterized model. The LLF coefficient files have been regenerated with the correct starting value for the magnetic latitude grid. Note that this fix requires no change to the PIM source code.

2. **BUG FIX**: The conversion of UT from hours to HHMM in routine PARAM for the URSI f_oF_2 model has been corrected. Previously, the conversion of UT from decimal hours to HHMM resulted in truncation of the minutes to zero, e.g. a UT of 1.5 hours was erroneously converted to 0100. The conversion has been modified so that the minutes are not truncated to zero, e.g. a UT of 1.5 hours is correctly converted to 0130.

3. Based on work done by Pat Doherty, we now recommend that a 27-day running mean of $F_{10.7}$ be used instead of a daily value. Pat found that a mean $F_{10.7}$ correlates much better with the ionosphere than a daily value, either because the solar EUV has less day-to-day variability than $F_{10.7}$ or because the day-to-day variability in solar EUV is uncorrelated with the day-to-day variability in $F_{10.7}$. The same applies to Sunspot Number (use a 27-day running mean instead of a daily value), although we recommend using $F_{10.7}$ instead of Sunspot Number as the solar activity index.

4. The top-level parameterized model routine PARAM has undergone substantial changes. The original motivation was a report from Jim Secan at Northwest Research Associates illustrating a discontinuity in PIM 1.4 TEC at the latitude transition region between the mid-latitude and high-latitude parameterized models. A closer examination of routine PARAM has yielded the following improvements:

    - It has been completely rewritten for optimization and readability. The number of routine calls has been reduced.

    - The transition between the mid-latitude and high-latitude parameterized models is smoother. Previously, in the mid/high-latitude transition region, the mid-latitude peak density and height were used to merge the mid-latitude and high-latitude density profiles, resulting in a discontinuity in density at the boundary between the high-latitude region and the mid/high-latitude transition region. Now, weighted averages of mid-latitude and high-latitude peak densities and heights are used in the merging, resulting in a smoother transition.

    - The transition between the low-latitude and mid-latitude parameterized models is smoother. Previously, a weighted average of low-latitude and mid-latitude critical frequencies was used to calculate the peak density for the merging of the low-latitude and mid-latitude density profiles. Now, the peak density used in the merging is calculated from a weighted average of low-latitude and mid-latitude peak densities. This is consistent with the profile merging process, and results in a smoother transition between the low-latitude and mid-latitude parameterized models.

5. The mid-level parameterized model routines LOW_PARAM, MID_PARAM, and USUMODEL have been rewritten to accommodate changes in the top-level parameterized model routine PARAM, to provide small performance gains by reducing routine calls and eliminating unnecessary calculations, and to improve readability.

6. The low-level parameterized USU model routines RECON, FOUR_COEFF, and FULL_DATA have been modified to accommodate changes to the mid-level parameterized model routine USUMODEL.

7. The semi-empirical f_oE model has been replaced by the CCIR recommendation. Previously, when f_oE normalization was requested, f_oE was calculated by a semi-empirical f_oE model from IRI. Now, it is calculated by the CCIR empirical f_oE model. This change was suggested by Leo McNamara.

8. Routines in module ENVIRON.FOR have been replaced to accommodate the CCIR empirical f_oE model.

9. Several routines have been modified to accommodate the new version of routine SOLDEC.

10. Several minor changes have been made for compatibility with Microsoft FORTRAN. They do not impact the results.

Changes to PIM 1.3 for PIM 1.4
5-February-1996

The changes to PIM 1.3 for PIM 1.4 focus on replacing the LLF coefficient set again, fixing a bug in the slant TEC calculation of azimuth/elevation grid output, and making PIM consistent with the forthcoming Visual PIM interface. The changes are summarized as follows:

1. BUG FIX: The coefficient set for the LLF parameterized model has been replaced again. Because of bugs in the version of LOWLAT that we used to generate the LLF coefficients for PIM 1.3, the LLF parameterized model in PIM 1.3 did not exhibit an $F_{10.7}$ dependence. The LOWLAT model has been repaired and the new LLF parameterized model does exhibit an $F_{10.7}$ dependence. The bottomside density from the model has also been improved, eliminating potential false structure introduced by spline interpolation. Note that this fix requires no change to the PIM source code.

2. BUG FIX: The slant TEC calculation for azimuth/elevation grid output has been corrected. Previously, vertical TEC was calculated instead of TEC along the line-of-sight.

3. In preparation for the forthcoming release of Visual PIM, the following changes have been made to PIM for consistency with Visual PIM:

    - The “Satellite Track” output grid type has been renamed to “Latitude/Longitude Pairs”.
    - The “Radar Azimuth/Elevation” output grid type has been renamed to “Azimuth/Elevation (ground-based)”.
    - Some of the variables in common block GRID have been renamed to reflect the new names of the output grid types.
    - Some of the routines in module OUTPUT.FOR have been renamed to reflect the new names of the output grid types.
    - The Year input parameter is now restricted to the range [1800,2100] and is validated based on this range. Previously, its range was unbounded and it was not validated.
    - Leap years are now accounted for when validating the Day of the Year input parameter. Previously, the maximum allowed value of the Day of the Year was 366 regardless of the Year input parameter.
    - The validation of the Universal Time input parameter has been improved by separately validating its hours and minutes. Previously, the Universal Time input parameter was validated as a single number (hhmm) in the range [0000,2359], which would allow an invalid number of minutes in some circumstances.
    - The $F_{10.7}$ input parameter is now validated unconditionally. Previously, it was validated only if the Sunspot Number Treatment Flag input parameter was 0 or 1.

- The Sunspot Number input parameter is now validated unconditionally. Previously, it was validated only if the Sunspot Number Treatment Flag input parameter was 0 or 2.
- The Latitude Increment input parameter for the rectangular latitude/longitude output grid is now restricted to the range [-180,180; ≠0] and is validated based on this range. Previously, its range was unbounded and it was not validated.
- The Number of Longitudes input parameter for the rectangular latitude/longitude output grid is now restricted to a maximum value of 3601. Previously, it had no upper limit.
- The Starting Longitude input parameter for the rectangular latitude/longitude output grid is now restricted to the range [-360,360] and is validated based on this range. Previously, its range was unbounded and it was not validated.
- The Longitude Increment input parameter for the rectangular latitude/longitude output grid is now restricted to the range [-360,360; ≠0] and is validated based on this range. Previously, its range was unbounded and it was not validated.
- The Longitude of each latitude/longitude pair in the latitude/longitude pairs grid is now restricted to the range [-360,360] and is validated based on this range. Previously, its range was unbounded and it was not validated.
- The Observer Longitude input parameter for the azimuth/elevation (ground-based) output grid is now restricted to the range [-360,360] and is validated based on this range. Previously, its range was unbounded and it was not validated.
- The Number of Azimuths input parameter for the azimuth/elevation (ground-based) output grid is now restricted to a maximum value of 3601. Previously, it had no upper limit.
- The Starting Azimuth input parameter for the for the azimuth/elevation (ground-based) output grid is now restricted to the range [-360,360] and is validated based on this range. Previously, its range was unbounded and it was not validated.
- The Azimuth Increment input parameter for the azimuth/elevation (ground-based) output grid is now restricted to the range [-360,360; ≠0] and is validated based on this range. Previously, its range was unbounded and it was not validated.
- The Elevation Increment input parameter for the azimuth/elevation (ground-based) output grid is now restricted to the range [-90,90; ≠0] and is validated based on this range. Previously, its range was unbounded and it was not validated.
- The output formats of some parameters in the output file have been widened to support the full ranges of the parameters.

Changes to PIM 1.2 for PIM 1.3
25-September-1995

The changes to PIM 1.2 for PIM 1.3 focus on making PIM as consistent with PRISM as possible. The changes are summarized as follows:

1. The coefficient set for the LLF parameterized model has been replaced, with the following benefits:
   - The topside O^+ density has been corrected. Previously, under certain circumstances the topside O^+ density from the LLF parameterized model decreased too rapidly due to a problem in interpolating the output of the LOWLAT physical model.

- The new LLF parameterization is in terms of magnetic local time instead of Universal Time, eliminating the need for an iterative root-finding scheme that determines Universal Time from magnetic local time and magnetic longitude.
- The magnetic latitude range of new LLF parameterization is ±34°, as was the original design intent. Previously, the magnetic latitude range was forced back to ±32° due to problems in interpolating the output of the LOWLAT physical model

2. The merging of peak density and density profiles across $F_{10.7}$, K_p, and regional model now uses exclusively the actual density values. Previously, critical frequency and the logarithm of the density profile were used in most cases, defeating the intent that the merging be done in a linear fashion.
3. In the calculation of TEC, the logarithm of electron density is now interpolated linearly instead of quadratically to avoid overestimation of TEC for pathological electron density profiles.
4. Logical flag ONLYOP is now a PARAMETER in subroutine REGMOD instead of a variable in common block INDIRECT. The value of PARAMETER ONLYOP should normally be FALSE, directing PIM to output total ion density instead of O^+ density.
5. To be consistent with PRISM, a header line now appears before each critical parameters line in the output file for output type 0 (critical frequencies and heights and TEC).
6. Unnecessary exponentiation of logarithmic density profiles has been eliminated in order to reduce execution time.
7. The parameterized USU model algorithm has been optimized for speed.
8. The reporting of OPEN errors in subroutine OPEN_FILE has been improved.
9. The case of all file paths is now preserved.
10. Leading spaces are now removed from all file paths.
11. Portability problems regarding character string concatenation have been resolved.
12. WRITE statements to unit 6 have been replaced by PRINT statements.
13. Unnecessary diagnostic PRINT statements have been removed.
14. Routines that are no longer used have been removed.
15. INCLUDE files that are no longer used have been removed.
16. Unused coding has been removed.
17. Variables that are no longer used have been removed.
18. All blank lines have been replaced by comment lines.

Changes to PIM 1.1 for PIM 1.2 29-April-1995

The changes to PIM 1.1 for PIM 1.2 focus on the elimination of the unused monthly-average F10.7 input parameter from PIM, the addition of a new “radar” output grid type, and a major improvement in size and speed, as well as miscellaneous bug fixes and minor improvements.

The changes are summarized as follows:

1. The monthly-average F10.7 input parameter has been eliminated from the PIM input stream and source code since it is not used by PIM.

2. A third output grid type has been added. As an alternative to the "rectangular" latitude/longitude or "satellite" latitude/longitude output grids, output may be specified on a "radar" azimuth/elevation grid. The radar output grid parameters are specified in the PIM input stream between the satellite output grid parameters and the altitude grid parameters.

3. Profiles from the LME, MLF, and USU parameterized models are now interpolated from 2×2 ($F_{10.7}$×K_p) reconstructed profile matrices instead of 3×3 reconstructed profile matrices, and profiles from the LLF parameterized model are now linearly interpolated from a 2×4 ($F_{10.7}$×Longitude Sector) reconstructed profile matrix instead of a 3×4 reconstructed profile matrix. Previously, the larger reconstructed profile matrices were calculated but not all of their information used since linear interpolation merged the profiles. Reducing the profile matrices cuts the size and time requirements of PIM approximately in half without significant (or any) change in the results. Under Lahey F77L-EM/32 5.20, the size of an unpacked PIM 1.1 executable is 4,219,561 bytes; the size of an unpacked PIM 1.2 executable is 2,439,630 bytes. The table below shows PIM 1.1 and 1.2 execution times for the standard test cases. The executables were created under Lahey F77L-EM/32 5.20 and run under Windows NT 3.5.

4. The initialization of common block LOW_E1 array element KPLE(1) has been corrected from 2. to 1. in BLOCK DATA LOWE. Previously, the incorrect value resulted in erroneous K_p interpolation of the profiles in the LME parameterized model for low input K_p values.

5. In the iterative scheme in subroutine MAG_TIME that determines UT from magnetic longitude and magnetic local time, logic has been corrected to avoid a stoppage in subroutine RTBIS.

6. In subroutines GRID_OUTPUT and SAT_OUTPUT, the dimensions of local arrays DATA and CDATA have been corrected to avoid out-of-bounds errors for large altitude grids.

7. In subroutine FNDMAX, handling of the special case C2=0 has been added to avoid a divide-by-zero error.

8. In subroutine SAT_OUTPUT, the format of the grid information written to the output file has been improved.

9. In record 6 of the output file, the coordinate system flag and output grid type flag have been combined in such a way that compatibility with the PRISM output format is maintained, while at the same time the output grid type can be identified.

10. All tabs have been replaced by spaces.

11. The version number and version date are now displayed to the default output device upon startup of PIM.

	Execution Time (seconds)		Ratio (1.2/1.1)
	PIM 1.1	PIM 1.2	
Test Case 1	69.28	23.85	0.34
Test Case 2	64.46	24.08	0.37
Test Case 3	28.21	13.38	0.47
Test Case 4	1072.04	546.03	0.51
Test Case 5	65.58	28.64	0.44
Test Case 6	n/a	1148.97	n/a

Changes to PIM 1.0.9 for PIM 1.1
13-October-1994

The changes to PIM 1.0.9 for PIM 1.1 are summarized as follows:

1. We have determined that unexpected periodic structure observed in maps of PIM 1.0.9 (and below) f_oF_2, h_mF_2, f_oE, and h_mE in the low- and mid- latitudes is introduced by the Fourier fitting of the parameterization process rather than the physical models (ECSD, LOWLAT, and MIDLAT) whose output is parameterized for PIM. After looking at the diurnal variation of the result of the first stage of the parameterization (the EOF coefficients), we have concluded that a Fourier series does not well represent the diurnal variation. We could not find a substitute basis set that satisfactorily reproduced the diurnal variations observed in the EOF coefficients, so we have eliminated the Fourier fitting from the low- and mid- latitude parameterization altogether. The orthogonal polynomial fitting in magnetic latitude is now done directly on the EOF coefficients instead of on Fourier coefficients. I've reparameterized the physical model output for the low- and mid- latitudes and modified PIM to use the new coefficient set.

2. In the low-latitude F-region parameterized model (LLF), the user now has the option of using either all longitude sectors or a single longitude sector via a new line in the input stream following the f_oE normalization input line. The new line contains an integer flag with valid values between 0 and 4: 0 if all longitude sectors are to be used in the LLF parameterized model, 1 if only the Brazilian sector is to be used, 2 if only the Indian sector is to be used, 3 if only the Pacific sector is to be used, or 4 if only the USA sector is to be used.

3. Grid output modes 0 and 2 have been redefined. Grid output mode 0 now gives TEC in addition to critical frequencies and heights and grid output mode 2 now gives EDPs, critical frequencies and heights, and TEC. , Grid output mode 1 gives EDPs as before. This was done to match PRISM's output specification. Grid output option 3 (no grid output) is no longer a valid choice.

4. The format of the grid output in all grid output modes has been changed to match on a record basis PRISM's output specification.

Changes to PIM 1.0.8 for PIM 1.0.9
22-September-1994

Over the last few months, I've been working with PIM 1.0.8 to speed it up. I've completed making changes and have produced a new version 1.0.9. The changes that I've made result in a 41% reduction in run-time for a default global grid run made on my optimized 486DX2-66 PC.

Support Services

Support for PIM is provided by Computational Physics, Inc. (CPI). Technical issues, such as installation and usage, should be addressed to Lincoln Brown. Administrative matters, such as user registration and CD-ROM purchase, should be directed to Laura Koziol.

Contacts: Lincoln Brown (technical)
Laura Koziol (administrative)

Mail: Computational Physics, Inc.
240 Bear Hill Road
Suite 202A
Waltham, Massachusetts 02154
USA

Telephone: 1-(781)-487-2250

Fax: 1-(781)-487-2290

E-mail: brown@cpiboston.com (Lincoln Brown)
koziol@cpiboston.com (Laura Koziol)

WWW: http://www.shore.net/~cpibos (CPI Boston)
http://www.cpi.com (CPI Headquarters)

FTP: ftp://andersun.plh.af.mil/pub/pim (AFRL)

Glossary

A visual cue to refer to other parts of the User Guide.

°	Degree of angle.
°E	Degree East.
°N	Degree North.
27-Day Running Mean	An average of daily values for the past 27 days.
ANSI	American National Standards Institute. The governing body for the FORTRAN 77 standard (ANSI X3.9-1978).
AFRL	Air Force Research Laboratory.
b	A binary unit (0 or 1).
B	Byte = 8 b.
CCIR	Comité Consultatif International des Radiocommunications.
CCIR f_oE model	The CCIR semi-empirical model of f_oE.
CGM	Corrected Geomagnetic.
cm	Centimeter = 10^{-2} m.
Console	Default input/output device. Typically keyboard for input and monitor for output.
Corrected Geomagnetic	A geomagnetic coordinate system that attempts to accurately represent the Earth's magnetic field.
CPI	Computational Physics, Inc.
Critical Frequency	The lowest frequency that is transmitted through an ionospheric region at vertical incidence. It is commonly converted to number density using the formula

	$n = 1.24 \times 10^4 \cdot f^2$ where n is the number density in cm^{-3} and f is the critical frequency in MHz.
Critical Height	The altitude of the peak density of an ionospheric region.
E	Ionospheric *E* region. An altitude regime of the ionosphere, extending from approximately 90 to 200 km altitude, that is generally dominated by NO^+ and O_2^+ and generally controlled by local molecular chemistry.
ECSD	The theoretical ionospheric model that is the basis of the LME parameterized model.
EDP	Electron Density Profile. Electron density as a function of altitude.
Empirical Orthonormal Function	A member of a basis set of functions derived from data.
EOF	Empirical Orthonormal Function.
EUV	Extreme Ultra-Violet. The region of the solar emission spectrum chiefly responsible for photoionization of the Earth's atmosphere.
F_1	Ionospheric F_1 region. An often ill-defined altitude regime of the ionosphere, at approximately 200 km altitude, that is generally a transition region between the ionospheric *E* and F_2 regions.
$F_{10.7}$	The measured solar radio flux at 2800 MHz (10.7 cm wavelength). A solar activity index. A 27-day running mean of $F_{10.7}$ correlates better with the ionosphere than an instantaneous or daily value.
F_2	Ionospheric F_2 region. An altitude regime of the ionosphere, extending from approximately 200 to 600 km altitude, that is generally dominated by O^+ and generally controlled by O^+ chemistry and O^+ diffusion.
f_oE	The critical frequency of the ionospheric *E* region.

f_oF_1	The critical frequency of the ionospheric F_1 region.
f_oF_2	The critical frequency of the ionospheric F_2 region.
Gallagher Plasmaspheric Model	A fast empirical model of plasmaspheric H^+ density.
Geographic	A geocentric coordinate system aligned with the Earth's axis of rotation.
GZIP	A freely-distributed lossless file compression utility.
h_mE	The critical height of the ionospheric E region.
h_mF_1	The critical height of the ionospheric F_1 region.
h_mF_2	The critical height of the ionospheric F_2 region.
Hz	Hertz = s^{-1}.
IDL	Interactive Data Language. A commercial software package for data analysis and visualization.
IMF	Interplanetary Magnetic Field. The magnetic field created by the Sun.
IMF B_y	The *y*-component of the IMF.
IMF B_z	The *z*-component of the IMF.
Input Stream	Default or console input.
km	Kilometer = 10^3 m.
K_p	Planetary K index. A geomagnetic activity index.
LLF	The parameterized model in PIM that provides vertical O^+ number density altitude profiles in the geomagnetic low-latitude region.
LME	The parameterized model in PIM that provides vertical NO^+ and O_2^+ number density altitude profiles in the geomagnetic low- and mid- latitude regions.

LOWLAT	The theoretical ionospheric model that is the basis of the LLF parameterized model.
m	Meter.
MB	MegaByte = 1,048,576 B
MHz	MegaHertz = 10^6 Hz.
MIDLAT	The theoretical ionospheric model that is the basis of the MLF parameterized model.
MLF	The parameterized model in PIM that provides vertical O^+ number density altitude profiles in the geomagnetic mid-latitude regions.
OPC	Orthogonal Polynomial Coefficient.
Orthogonal Polynomial Coefficient	A coefficient for an orthogonal polynomial used to describe the latitude variation of a parameter.
Output Stream	Default or console output.
PIM	Parameterized Ionospheric Model.
s	Second.
Slant	Referring to an arbitrary line-of-sight defined by an azimuth and an elevation.
SSN	Sunspot Number. A solar activity index. A 27-day running mean of *SSN* correlates better with ionosphere than an instantaneous or daily value.
TAR	A freely-distributable utility that archives file in a directory structure by combining them into a single file, while preserving the names and content of the files and the directory structure.
TEC	Total Electron Content. Electron density integrated along a line-of-sight.
TEC Unit	10^{12} cm^{-2}.
Universal Time	Greenwich Mean Time.

URSI	Union Radio-Scientifique Internationale.
URSI-88	A set of semi-empirical coefficients describing monthly median f_oF_2 and h_mF_2. Only the f_oF_2 coefficients are used by PIM.
USU	The parameterized model in PIM that provides vertical NO^+, O_2^+, and O^+ number density altitude profiles in the geomagnetic high-latitude regions. Also the theoretical model that is the basis for the parameterized USU model.
Vertical	Referring to a line-of-sight with an elevation of 90°.
W	Watt.

References

Daniell, Jr., R. E., L. D. Brown, D. N. Anderson, M. W. Fox, P. H. Doherty, D. T. Decker, J. J. Sojka, and R. W. Schunk, Parameterized ionospheric model: A global ionospheric parameterization based on first principles models, *Radio Sci., 30*, 1499-1510, 1995.

Gallagher, D. L., P. D. Craven, and R. H. Comfort, An empirical model of the earth's plasmasphere, *Adv. Space Res., 8*, 15-24, 1988.

More references are listed at the end of Appendix A.

Tables

Table 3.1 The PIM file set and directory structure restored from the TAR file.

Directory	Files	Description
cgmdb		Directory containing the corrected geomagnetic coordinate conversion database
	cglalo.dat	Corrected geomagnetic coordinate conversion database (a text file)
llfdb/form		Directory containing the text version of the parameterized low-latitude F-region (LLF) model database
	fac*.oe	Text LLF O^+ empirical orthonormal function (EOF) files for the Pacific sector
	fac*.op	Text LLF O^+ orthogonal polynomial coefficient (OPC) files for the Pacific sector
	fnd*.oe	Text LLF O^+ EOF coefficient files for the Indian sector
	fnd*.op	Text LLF O^+ OPC files for the Indian sector
	frz*.oe	Text LLF O^+ EOF coefficient files for the Brazilian sector
	frz*.op	Text LLF O^+ OPC files for the Brazilian sector
	fsa*.oe	Text LLF O^+ EOF coefficient files for the USA sector
	fsa*.op	Text LLF O^+ OPC files for the USA sector
llfdb/utils		Directory containing utilities for the parameterized LLF model
	environ.f	FORTRAN library of geophysical environment routines required by llf.f
	environ.pro	IDL library of geophysical environment routines required by llf.pro
	llf.f	Callable FORTRAN module that produces O^+ density altitude profiles from the LLF database
	llf.pro	Callable IDL module that produces O^+ density altitude profiles from the LLF database
	llftofrm.f	FORTRAN utility program for converting the LLF database from binary to text
	llftofrm.in	Example input stream to llftofrm.f that converts the entire LLF database from binary to text
	llftounf.f	FORTRAN utility program for converting the LLF database from text to binary
	llftounf.in	Example input stream to llftounf.f that converts the entire LLF database from text to binary
	strnglib.f	FORTRAN library of string manipulation routines required by llf.f, llftofrm.f, and llftounf.f
lmedb/form		Directory containing the text version of the parameterized low- and mid- latitude E-region (LME) model database
	flm*.noe	Text LME NO^+ EOF coefficient files

	flm*.nop	Text LME NO^+ OPC files
	flm*.o2e	Text LME O_2^+ EOF coefficient files
	flm*.o2p	Text LME O_2^+ OPC files
lmedb/utils		Directory containing utilities for the parameterized LME model
	environ.f	FORTRAN library of geophysical environment routines required by lme.f
	environ.pro	IDL library of geophysical environment routines required by lme.pro
	lme.f	Callable FORTRAN module that produces NO^+ and O_2^+ density altitude profiles from the LME database
	lme.pro	Callable IDL module that produces NO^+ and O_2^+ density altitude profiles from the LME database
	lmetofrm.f	FORTRAN utility program for converting the LME database from binary to text
	lmetofrm.in	Example input stream to lmetofrm.f that converts the entire LME database from binary to text
	lmetounf.f	FORTRAN utility program for converting the LME database from text to binary
	lmetounf.in	Example input stream to lmetounf.f that converts the entire LME database from text to binary
	strnglib.f	FORTRAN library of string manipulation routines required by lme.f, lmetofrm.f, and lmetounf.f
mlfdb/form		Directory containing the text version of the parameterized mid-latitude F-region (MLF) model database
	fnm*.oe	Text MLF O^+ EOF coefficient files for the northern geomagnetic hemisphere
	fnm*.op	Text MLF O^+ OPC files for the northern geomagnetic hemisphere
	fsm*.oe	Text MLF O^+ EOF coefficient files for the southern geomagnetic hemisphere
	fsm*.op	Text MLF O^+ OPC files for the southern geomagnetic hemisphere
mlfdb/utils		Directory containing utilities for the parameterized MLF model
	environ.f	FORTRAN library of geophysical environment routines required by mlf.f
	environ.pro	IDL library of geophysical environment routines required by mlf.pro
	mlf.f	Callable FORTRAN module that produces O^+ density altitude profiles from the MLF database
	mlf.pro	Callable IDL module that produces O^+ density altitude profiles from the MLF database
	mlftofrm.f	FORTRAN utility program for converting the MLF database from binary to text
	mlftofrm.in	Example input stream to mlftofrm.f that converts the entire MLF database from binary to text
	mlftounf.f	FORTRAN utility program for converting the MLF database from text to binary
	mlftounf.in	Example input stream to mlftounf.f that converts the entire MLF database from text to binary
	strnglib.f	FORTRAN library of string manipulation routines required by mlf.f, mlftofrm.f, and mlftounf.f

pim/help		Directory containing informational files
	pim17ug.*	PIM 1.7 User Guide in multiple formats
pim/idlprocs		Directory containing IDL procedures for plotting PIM output
	aeccrit.pro	Plots contours of PIM critical frequencies and heights and TEC vs. azimuth and elevation for azimuth/elevation grid output
	aeccut.pro	Plots contours of PIM electron density and plasma frequency for cuts in azimuth, elevation, or altitude for azimuth/elevation grid output
	aepcrit.pro	Plots profiles of PIM critical frequencies and heights and TEC vs. azimuth or elevation for azimuth/elevation grid output
	aepcut.pro	Plots profiles of PIM electron density and plasma frequency for cuts in azimuth, elevation, or altitude for azimuth/elevation grid output
	llpccut.pro	Plots contours of PIM electron density and plasma frequency for latitude/longitude pairs grid output
	llppcrit.pro	Plots profiles of PIM critical frequencies and heights and TEC for latitude/longitude pairs grid output
	llppcut.pro	Plots profiles of PIM electron density and plasma frequency for latitude/longitude pairs grid output
	recccrit.pro	Plots contours of PIM critical frequencies and heights and TEC vs. latitude and longitude for rectangular grid output
	recccut.pro	Plots contours of PIM electron density and plasma frequency for cuts in latitude, longitude, or altitude for rectangular grid output
	recpcrit.pro	Plots profiles of PIM critical frequencies and heights and TEC vs. latitude or longitude for rectangular grid output
	recpcut.pro	Plots profiles of PIM electron density and plasma frequency for cuts in latitude, longitude, or altitude for rectangular grid output
pim/source		Directory containing PIM FORTRAN source code and example PIM MAKE file
	*.f	PIM FORTRAN source code
	*.inc	PIM FORTRAN INCLUDE files
	makefile	Example PIM MAKE file
pim/testcase		Directory containing PIM test case input and output
	testcas*.in	Input streams for PIM test case runs

	testcas*.log	Output streams from PIM test case runs
	testcas*.out	Output files from PIM test case runs
ursidb/form		Directory containing the text version of the URSI-88 coefficients database
	ursi88.dat	Text URSI-88 coefficients database
ursidb/utils		Directory containing utilities for the URSI-88 coefficients database
	ursistod.f	FORTRAN utility program for converting the URSI-88 coefficients database from text to binary
usudb/form		Directory containing the text version of the parameterized USU (high-latitude E- and F- region) model database
	fnh*.noe	Text USU NO^+ EOF coefficient files for the northern geomagnetic hemisphere
	fnh*.nop	Text USU NO^+ OPC files for the northern geomagnetic hemisphere
	fnh*.o2e	Text USU O_2^+ EOF coefficient files for the northern geomagnetic hemisphere
	fnh*.o2p	Text USU O_2^+ OPC files for the northern geomagnetic hemisphere
	fnh*.oe	Text USU O^+ EOF coefficient files for the northern geomagnetic hemisphere
	fnh*.op	Text USU O^+ OPC files for the northern geomagnetic hemisphere
	fsh*.noe	Text USU NO^+ EOF coefficient files for the southern geomagnetic hemisphere
	fsh*.nop	Text USU NO^+ OPC files for the southern geomagnetic hemisphere
	fsh*.o2e	Text USU O_2^+ EOF coefficient files for the southern geomagnetic hemisphere
	fsh*.o2p	Text USU O_2^+ OPC files for the southern geomagnetic hemisphere
	fsh*.oe	Text USU O^+ EOF coefficient files for the southern geomagnetic hemisphere
	fsh*.op	Text USU O^+ OPC files for the southern geomagnetic hemisphere
usudb/utils		Directory containing utilities for the parameterized USU model
	environ.f	FORTRAN library of geophysical environment routines required by usu.f
	environ.pro	IDL library of geophysical environment routines required by usu.pro
	strnglib.f	FORTRAN library of string manipulation routines required by usu.f, usutofrm.f, and usutounf.f
	usu.f	Callable FORTRAN module that produces NO^+, O_2^+, and O^+ density altitude profiles from the USU database
	usu.pro	Callable IDL module that produces NO^+, O_2^+, and O^+ density altitude profiles from the USU database
	usutofrm.f	FORTRAN utility program for converting the USU database from binary to text
	usutofrm.in	Example input stream to usutofrm.f that converts the entire USU database from binary to text
	usutounf.f	FORTRAN utility program for converting the USU database from text to binary

	usutounf.in	Example input stream to usutounf.f that converts the entire USU database from text to binary

Table 3.2 The PIM file set and directory structure after installation.

Directory	Files	Description
cgmdb		Directory containing the corrected geomagnetic coordinate conversion database
	cglalo.dat	Corrected geomagnetic coordinate conversion database (a text file)
llfdb/unform		Directory containing the binary version of the parameterized low-latitude F-region (LLF) model database
	brz*.oe	Binary LLF O^+ EOF coefficient files for the Brazilian sector
	brz*.op	Binary LLF O^+ OPC files for the Brazilian sector
	pac*.oe	Binary LLF O^+ EOF files for the Pacific sector
	pac*.op	Binary LLF O^+ OPC files for the Pacific sector
	ind*.oe	Binary LLF O^+ EOF coefficient files for the Indian sector
	ind*.op	Binary LLF O^+ OPC files for the Indian sector
	usa*.oe	Binary LLF O^+ EOF coefficient files for the USA sector
	usa*.op	Binary LLF O^+ OPC files for the USA sector
llfdb/utils		Directory containing utilities for the parameterized LLF model
	environ.f	FORTRAN library of geophysical environment routines required by llf.f
	environ.pro	IDL library of geophysical environment routines required by llf.pro
	llf.f	Callable FORTRAN module that produces O^+ density altitude profiles from the LLF database
	llf.pro	Callable IDL module that produces O^+ density altitude profiles from the LLF database
	llftofrm.f	FORTRAN utility program for converting the LLF database from binary to text
	llftofrm.in	Input stream to llftofrm.f that converts the entire LLF database from binary to text, for your platform
	llftounf.f	FORTRAN utility program for converting the LLF database from text to binary
	llftounf.in	Input stream to llftounf.f that converts the entire LLF database from text to binary, for your platform
	strnglib.f	FORTRAN library of string manipulation routines required by llf.f, llftofrm.f, and llftounf.f
lmedb/unform		Directory containing the binary version of the parameterized low- and mid- latitude E-region (LME) model database

	lm*.noe	Binary LME NO^+ EOF coefficient files
	lm*.nop	Binary LME NO^+ OPC files
	lm*.o2e	Binary LME O_2^+ EOF coefficient files
	lm*.o2p	Binary LME O_2^+ OPC files
lmedb/utils		Directory containing utilities for the parameterized LME model
	environ.f	FORTRAN library of geophysical environment routines required by lme.f
	environ.pro	IDL library of geophysical environment routines required by lme.pro
	lme.f	Callable FORTRAN module that produces NO^+ and O_2^+ density altitude profiles from the LME database
	lme.pro	Callable IDL module that produces NO^+ and O_2^+ density altitude profiles from the LME database
	lmetofrm.f	FORTRAN utility program for converting the LME database from binary to text
	lmetofrm.in	Input stream to lmetofrm.f that converts the entire LME database from binary to text, for your platform
	lmetounf.f	FORTRAN utility program for converting the LME database from text to binary
	lmetounf.in	Input stream to lmetounf.f that converts the entire LME database from text to binary, for your platform
	strnglib.f	FORTRAN library of string manipulation routines required by lme.f, lmetofrm.f, and lmetounf.f
mlfdb/unform		Directory containing the binary version of the parameterized mid-latitude F-region (MLF) model database
	nm*.oe	Binary MLF O^+ EOF coefficient files for the northern geomagnetic hemisphere
	nm*.op	Binary MLF O^+ OPC files for the northern geomagnetic hemisphere
	sm*.oe	Binary MLF O^+ EOF coefficient files for the southern geomagnetic hemisphere
	sm*.op	Binary MLF O^+ OPC files for the southern geomagnetic hemisphere
mlfdb/utils		Directory containing utilities for the parameterized MLF model
	environ.f	FORTRAN library of geophysical environment routines required by mlf.f
	environ.pro	IDL library of geophysical environment routines required by mlf.pro
	mlf.f	Callable FORTRAN module that produces O^+ density altitude profiles from the MLF database
	mlf.pro	Callable IDL module that produces O^+ density altitude profiles from the MLF database
	mlftofrm.f	FORTRAN utility program for converting the MLF database from binary to text
	mlftofrm.in	Input stream to mlftofrm.f that converts the entire MLF database from binary to text, for your

		platform
	mlftounf.f	FORTRAN utility program for converting the MLF database from text to binary
	mlftounf.in	Input stream to mlftounf.f that converts the entire MLF database from text to binary, for your platform
	strnglib.f	FORTRAN library of string manipulation routines required by mlf.f, mlftofrm.f, and mlftounf.f
pim/bin		The PIM working directory
	path_nam.txt	The PIM path names input file for your installation
	pim	The PIM executable for your installation (the name may vary)
pim/help		Directory containing informational files
	pim17ug.*	PIM 1.7 User Guide in multiple formats
pim/idlprocs		Directory containing IDL procedures for plotting PIM output
	aeccrit.pro	Plots contours of PIM critical frequencies and heights and TEC vs. azimuth and elevation for azimuth/elevation grid output
	aeccut.pro	Plots contours of PIM electron density and plasma frequency for cuts in azimuth, elevation, or altitude for azimuth/elevation grid output
	aepcrit.pro	Plots profiles of PIM critical frequencies and heights and TEC vs. azimuth or elevation for azimuth/elevation grid output
	aepcut.pro	Plots profiles of PIM electron density and plasma frequency for cuts in azimuth, elevation, or altitude for azimuth/elevation grid output
	llpccut.pro	Plots contours of PIM electron density and plasma frequency for latitude/longitude pairs grid output
	llppcrit.pro	Plots profiles of PIM critical frequencies and heights and TEC for latitude/longitude pairs grid output
	llppcut.pro	Plots profiles of PIM electron density and plasma frequency for latitude/longitude pairs grid output
	recccrit.pro	Plots contours of PIM critical frequencies and heights and TEC vs. latitude and longitude for rectangular grid output
	recccut.pro	Plots contours of PIM electron density and plasma frequency for cuts in latitude, longitude, or altitude for rectangular grid output
	recpcrit.pro	Plots profiles of PIM critical frequencies and heights and TEC vs. latitude or longitude for rectangular grid output

	recpcut.pro	Plots profiles of PIM electron density and plasma frequency for cuts in latitude, longitude, or altitude for rectangular grid output
pim/source		Directory containing PIM FORTRAN source code and MAKE file for your platform
	*.f	PIM FORTRAN source code
	*.inc	PIM FORTRAN INCLUDE files
	makefile	The PIM MAKE file for your platform (if MAKE is supported)
pim/testcase		Directory containing PIM test case input and output
	testcas*.in	Input streams for PIM test case runs
	testcas*.log	Output streams from PIM test case runs
	testcas*.out	Output files from PIM test case runs
ursidb/unform		Directory containing the binary version of the URSI-88 coefficients database
	ursi88da.dat	Binary URSI-88 coefficients database
ursidb/utils		Directory containing utilities for the URSI-88 coefficients database
	ursistod.f	FORTRAN utility program for converting the URSI-88 coefficients database from text to binary, for your platform
usudb/unform		Directory containing the binary version of the parameterized USU (high-latitude E- and F-region) model database
	nh*.noe	Binary USU NO^+ EOF coefficient files for the northern geomagnetic hemisphere
	nh*.nop	Binary USU NO^+ OPC files for the northern geomagnetic hemisphere
	nh*.o2e	Binary USU O_2^+ EOF coefficient files for the northern geomagnetic hemisphere
	nh*.o2p	Binary USU O_2^+ OPC files for the northern geomagnetic hemisphere
	nh*.oe	Binary USU O^+ EOF coefficient files for the northern geomagnetic hemisphere
	nh*.op	Binary USU O^+ OPC files for the northern geomagnetic hemisphere
	sh*.noe	Binary USU NO^+ EOF coefficient files for the southern geomagnetic hemisphere
	sh*.nop	Binary USU NO^+ OPC files for the southern geomagnetic hemisphere
	sh*.o2e	Binary USU O_2^+ EOF coefficient files for the southern geomagnetic hemisphere
	sh*.o2p	Binary USU O_2^+ OPC files for the southern geomagnetic hemisphere
	sh*.oe	Binary USU O^+ EOF coefficient files for the southern geomagnetic hemisphere
	sh*.op	Binary USU O^+ OPC files for the southern geomagnetic hemisphere
usudb/utils		Directory containing utilities for the parameterized USU model
	environ.f	FORTRAN library of geophysical environment routines required by usu.f

environ.pro	IDL library of geophysical environment routines required by usu.pro
strnglib.f	FORTRAN library of string manipulation routines required by usu.f, usutofrm.f, and usutounf.f
usu.f	Callable FORTRAN module that produces NO^+, O_2^+, and O^+ density altitude profiles from the USU database
usu.pro	Callable IDL module that produces NO^+, O_2^+, and O^+ density altitude profiles from the USU database
usutofrm.f	FORTRAN utility program for converting the USU database from binary to text
usutofrm.in	Input stream to usutofrm.f that converts the entire USU database from binary to text, for your platform
usutounf.f	FORTRAN utility program for converting the USU database from text to binary
usutounf.in	Input stream to usutounf.f that converts the entire USU database from text to binary, for your platform

Table 4.1 The PIM input file **path_nam.txt**.

Record Number	Element Name	Data Type	Valid Range	Description
1	DATYPE	Integer	1 or 2	A flag that instructs PIM how to open direct-access files: if record-lengths of direct-access files in FORTRAN OPEN statements are specified in bytes, then DATYPE should be **1**; if record-lengths of direct-access files in FORTRAN OPEN statements are specified in longwords (4-byte units), then DATYPE should be **2**. Most FORTRAN implementations in DOS, Windows NT, and UNIX platforms use bytes; the FORTRAN implementation in the VMS platform uses longwords.
2	CGMPATH	Character*80	Valid path	The location of the corrected geomagnetic (CGM) coordinate conversion database. Because the name of the CGM coordinate conversion database file is appended to CGMPATH, it must terminate with a valid path separator such as \ in DOS, / in UNIX, or **]** in VMS. For example, a valid value for CGMPATH in UNIX might be **../cgmdb/**.

3	USUPATH	Character*80	Valid path	The location of the binary parameterized USU model database. Because the names of binary parameterized USU model database files are appended to USUPATH, it must terminate with a valid path separator such as \ in DOS, / under UNIX, or **]** in VMS. For example, a valid value for USUPATH in UNIX might be **../usudb/unform/**.
4	MLFPATH	Character*80	Valid path	The location of the binary parameterized MLF model database. Because the names of binary parameterized MLF model database files are appended to MLFPATH, it must terminate with a valid path separator such as \ in DOS, / under UNIX, or **]** in VMS. For example, a valid value for MLFPATH in UNIX might be **../mlfdb/unform/**.
5	LLFPATH	Character*80	Valid path	The location of the binary parameterized LLF model database. Because the names of binary parameterized LLF model database files are appended to LLFPATH, it must terminate with a valid path separator such as \ in DOS, / under UNIX, or **]** in VMS. For example, a valid value for LLFPATH in UNIX might be **../llfdb/unform/**.
6	LMEPATH	Character*80	Valid path	The location of the binary parameterized LME model database. Because the names of binary parameterized LME model database files are appended to LMEPATH, it must terminate with a valid path separator such as \ in DOS, / under UNIX, or **]** in VMS. For example, a valid value for LMEPATH in UNIX might be **../lmedb/unform/**.
7	URSIPATH	Character*80	Valid path	The location of the binary URSI-88 coefficients database. Because the name of the binary URSI-88 coefficients database file is appended to URSIPATH, it must terminate with a valid path separator such as \ in DOS, / in UNIX, or **]** in VMS. For example, a valid value for URSIPATH in UNIX might be **../ursidb/unform/**.

Table 4.2 The PIM input stream.

Record Number	Element Name	Data Type	Units	Valid Range	Description
1	YEAR	Integer	Years	1800 to 2100	The calendar year.
2	DOY	Integer	Days	1 to 365(366)	The day of the year. Its maximum allowed value depends on YEAR: if YEAR is a leap year, then the maximum allowed value of DOY is **366**; if YEAR is not a leap year, then the maximum allowed value of DOY is **365**.
3	UT	Integer	hhmm	0000 to 2359	The Universal Time.
4	OUTTYPE	Integer	n/a	0, 1, or 2	A flag that instructs PIM what quantities to output: **0** for critical frequencies and heights for the ionospheric *E* and F_2 regions, and *TEC*; **1** for *EDP*s; or **2** for *EDP*s, critical frequencies and heights for the ionospheric *E* and F_2 regions, and *TEC*.
5	GRIDTYPE	Integer	n/a	0, 1, or 2	A flag that instructs PIM what type of grid to use for output: **0** for a rectangular latitude/longitude grid (see elements NLAT, SLAT, DLAT, NLON, SLON, and DLON); **1** for latitude/longitude pairs (see elements NPR, LATPR, and LONPR); or **2** for an azimuth/elevation grid with a ground-based observer (see elements OBSLAT, OBSLON, NAZ, SAZ, DAZ, NEL, SEL, and DEL).
6	OUTFILE	Character*32	n/a	Valid file name	The name of the output file.
7	IMFBY	Character*1	n/a	- or +	The orientation of *IMF* B_y: **-** for negative or **+** for positive.
8	IMFBZ	Character*1	n/a	-, 0, or +	The orientation of *IMF* B_z: **-** for negative, **0** for zero, or **+** for positive. It affects the value of K_p used by PIM (see element KP).
9	SSNTREAT	Integer	n/a	0, 1, or 2	A flag that instructs PIM how to handle $F_{10.7}$

					and *SSN*: **0** if $F_{10.7}$ and *SSN* are independent of each other; **1** if *SSN* is calculated from $F_{10.7}$; or **2** if $F_{10.7}$ is calculated from *SSN*. See elements F10P7 and SSN.
10	F10P7	Real	10^{22} W m^{-2} Hz^{-1}	0. to 300.	The solar radio flux at 10.7 cm (2800 MHz), $F_{10.7}$. If the element SSNTREAT is **2**, then F10P7 is read and validated but thereafter ignored, and $F_{10.7}$ is calculated from *SSN*. A 27-day running mean of $F_{10.7}$ is recommended instead of an instantaneous or daily $F_{10.7}$ because the ionosphere correlates better with the mean value.
11	SSN	Real	n/a	0. to 300.	The sunspot number, *SSN*. If element SSNTREAT is **1**, then SSN is read and validated but thereafter ignored, and *SSN* is calculated from $F_{10.7}$. A 27-day running mean of *SSN* is recommended instead of an instantaneous or daily *SSN* because the ionosphere correlates better with the mean value.
12	KP	Real	n/a	0. to 9.	The geomagnetic activity index, K_p. If element IMFBZ is **0** or **+**, then KP is read and validated but thereafter ignored, and a K_p of **1.** is used by PIM.
13	FOF2NORM	Integer	n/a	0 or 1	A flag that instructs PIM how to normalize f_oF_2: **0** if f_oF_2 is normalized to the URSI-88 coefficients or **1** if f_oF_2 is not normalized to the URSI-88 coefficients. A value of **1** is recommended.
14	FOENORM	Integer	n/a	0 or 1	A flag that instructs PIM how to normalize f_oE: **0** if f_oE is normalized to the CCIR f_oE model or **1** if f_oE is not normalized to the CCIR f_oE

					model. A value of **1** is recommended.
15	LLFSECT	Integer	n/a	0, 1, 2, 3, or 4	A flag that instructs PIM which longitude sectors to use in the LLF parameterized model: **0** if models for all four longitude sectors are used; **1** if only the model for the Brazilian longitude sector is used; **2** if only the model for the Indian longitude sector is used; **3** if only the model for the Pacific longitude sector is used; or **4** if only the model for the USA longitude sector is used. If PIM is being used to make global runs, then a value of **0** is recommended; PIM will merge the models for the four longitude sectors with a longitude-dependent weighting. If PIM is being used to study a particular longitude region, then LLFSECT can be set so that PIM uses the model for a single longitude sector.
16	CRDTYPE	Character*1	n/a	G or M	The output latitude/longitude coordinate system: **G** for geographic or **M** for corrected geomagnetic. See elements SLAT, SLON, LATPR, LONPR, OBSLAT, OBSLON, and SAZ.
17	NLAT	Integer	n/a	≥ 1	The number of latitudes in the rectangular latitude/longitude output grid. If element GRIDTYPE is not **0**, then NLAT is read and validated but thereafter ignored.
18	SLAT	Real	°N	-90. to 90.	The starting latitude of the rectangular latitude/longitude output grid. Its coordinate system is defined by element CRDTYPE. If element GRIDTYPE is not **0**, then SLAT is read and validated but thereafter ignored.
19	DLAT	Real	°N	-180. to 180.; ≠ 0.	The latitude increment of the rectangular

					latitude/longitude output grid. If element GRIDTYPE is not **0**, then DLAT is read and validated but thereafter ignored.
20	NLON	Integer	n/a	1 to 3601	The number of longitudes in the rectangular latitude/longitude output grid. If element GRIDTYPE is not **0**, then NLON is read and validated but thereafter ignored.
21	SLON	Real	°E	-360. to 360.	The starting longitude of the rectangular latitude/longitude output grid. Its coordinate system is defined by element CRDTYPE. If element GRIDTYPE is not **0**, then SLON is read and validated but thereafter ignored.
22	DLON	Real	°E	-360. to 360.; ≠ 0.	The longitude increment of the rectangular latitude/longitude output grid. If element GRIDTYPE is not **0**, then DLON is read and validated but thereafter ignored.
23	NPR	Integer	n/a	1 to 1000	The number of latitude/longitude pairs in the latitude/longitude pairs output grid. If element GRIDTYPE is not **1**, then NPR is read and validated but thereafter ignored.
24 (repeat NPR times)	$LATPR_i$	Real	°N	-90. to 90.	The latitude of the i^{th} latitude/longitude pair in the latitude/longitude pairs output grid. Its coordinate system is defined by element CRDTYPE. If element GRIDTYPE is not **1**, then $LATPR_i$ is read and validated but thereafter ignored.
	$LONPR_i$	Real	°E	-360. to 360.	The longitude of the i^{th} latitude/longitude pair in the latitude/longitude pairs output grid. Its coordinate system is defined by element CRDTYPE. If element GRIDTYPE is not **1**, then $LONPR_i$ is read and validated but thereafter ignored.

25	OBSLAT	Real	°N	-90. to 90.	The latitude of the ground observer for the azimuth/elevation output grid. Its coordinate system is defined by element CRDTYPE. If element GRIDTYPE is not **2**, then OBSLAT is read and validated but thereafter ignored.
26	OBSLON	Real	°E	-360. to 360.	The longitude of the ground observer for the azimuth/elevation output grid. Its coordinate system is defined by element CRDTYPE. If element GRIDTYPE is not **2**, then OBSLON is read and validated but thereafter ignored.
27	NAZ	Integer	n/a	1 to 3601	The number of azimuths in the azimuth/elevation output grid. If element GRIDTYPE is not **2**, then NAZ is read and validated but thereafter ignored.
28	SAZ	Real	°	-360. to 360.	The starting azimuth of the azimuth/elevation output grid. The orientation of north (*AZ*=0°) depends on the coordinate system defined by element CRDTYPE. If element GRIDTYPE is not **2**, then SAZ is read and validated but thereafter ignored.
29	DAZ	Real	°	-360. to 360.; ≠ 0.	The azimuth increment of the azimuth/elevation output grid. If element GRIDTYPE is not **2**, then DAZ is read and validated but thereafter ignored.
30	NEL	Integer	n/a	≥ 1	The number of elevations in the azimuth/elevation output grid. If element GRIDTYPE is not **2**, then NEL is read and validated but thereafter ignored.
31	SEL	Real	°	0. to 90.	The starting elevation of the azimuth/elevation output grid. If element GRIDTYPE is not **2**, then SEL is read and validated but thereafter ignored.

32	DEL	Real	°	-90. to 90.; ≠ 0.	The elevation increment of the azimuth/elevation output grid. If element GRIDTYPE is not **2**, then DEL is read and validated but thereafter ignored.
33	NALT	Integer	n/a	1 to 100	The number of altitudes in the output altitude grid.
34 (Repeat NALT times)	ALT_i	Real	km	90. to 25000.	The i^{th} altitude in the output altitude grid.
35	PLASPH	Character*1	n/a	Y or N	A flag that instructs PIM whether to include a plasmasphere: **Y** for yes (use the Gallagher plasmaspheric model) or **N** for no.

Table 4.3 File name encoding in the parameterized USU model database.

Character Position	Range	Description
1	n or s	The geomagnetic hemisphere: **n** for northern or **s** for southern.
2	h	The geomagnetic latitude region: **h** for high latitude.
3	e, s, or w	The season: **e** for equinox (day of the year **82**); **s** for summer in the northern geomagnetic hemisphere (day of the year **173**), corresponding to winter in the southern geomagnetic hemisphere; or **w** for winter in the northern geomagnetic hemisphere (day of the year **357**), corresponding to summer in the southern geomagnetic hemisphere.
4	l, m, or h	The geomagnetic activity level: **l** for low (K_p=**1.**); **m** for moderate (K_p=**3.5**); or **h** for high (K_p=**6.**).
5	l, m, or h	The solar activity level: **l** for low ($F_{10.7}$=**70.** and year **1987**); **m** for moderate ($F_{10.7}$=**130.** and year **1984**); or **h** for high ($F_{10.7}$=**210.** and year **1981**).
6-7	bc or de	The convection pattern: **bc** for the Heppner-Maynard BC convection pattern (*IMF* B_y **+**) or **de** for the Heppner-Maynard DE convection pattern (*IMF* B_y **-**).
8	.	The separator between the root of the file name and its extension.

9(-10)	no, o2, or o	The ion species: **no** for NO^+; **o2** for O_2^+; or **o** for O^+.
10(11)	e or p	The contents of the file: **e** for EOFs or **p** for OPCs.

Table 4.4 File name encoding in the parameterized MLF model database.

Character Position	Range	Description
1	n or s	The geomagnetic hemisphere: **n** for northern or **s** for southern.
2	m	The geomagnetic latitude region: **m** for midlatitude.
3	e, s, or w	The season: **e** for equinox (day of the year **82**); **s** for summer in the northern geomagnetic hemisphere (day of the year **173**), corresponding to winter in the southern geomagnetic hemisphere; or **w** for winter in the northern geomagnetic hemisphere (day of the year **357**), corresponding to summer in the southern geomagnetic hemisphere.
4	l, m, or h	The geomagnetic activity level: **l** for low (K_p=**1.**); **m** for moderate (K_p=**3.5**); or **h** for high (K_p=**6.**).
5	l, m, or h	The solar activity level: **l** for low ($F_{10.7}$=**70.** and year **1987**); **m** for moderate ($F_{10.7}$=**130.** and year **1984**); or **h** for high ($F_{10.7}$=**210.** and year **1981**).
6	.	The separator between the root of the file name and its extension.
7	o	The ion species: **o** for O^+.
8	e or p	The contents of the file: **e** for EOFs or **p** for OPCs.

Table 4.5 File name encoding in the parameterized LLF model database.

Character Position	Range	Description
1-3	brz, ind, pac, or usa	The geomagnetic longitude sector: **brz** for Brazilian; **ind** for Indian; **pac** for Pacific; or **usa** for USA.
4-5	03, 06, or 12	The month: **03** for March (day of the year **82**); **06** for June (day of the year **173**); or **12** for December (day of the year **357**).
6-8	min, mod, or max	The solar activity level: **min** for low ($F_{10.7}$=**70.** and year **1987**); **mod** for moderate ($F_{10.7}$=**130.** and year **1984**); or **max** for high ($F_{10.7}$=**210.** and year **1981**).

9	.	The separator between the root of the file name and its extension.
10	o	The ion species: **o** for O^+.
11	e or p	The contents of the file: **e** for EOFs or **p** for OPCs.

Table 4.6 File name encoding in the parameterized LME model database.

Character Position	Range	Description
1-2	lm	The geomagnetic latitude region: **lm** for low- and mid- latitude.
3-4	03, 06, or 12	The month: **03** for March (day of the year **82**); **06** for June (day of the year **173**); or **12** for December (day of the year **357**).
5	l, m, or h	The geomagnetic activity level: **l** for low (K_p=**1.**); **m** for moderate (K_p=**3.5**); or **h** for high (K_p=**6.**).
6	l, m, or h	The solar activity level: **l** for low ($F_{10.7}$=**70.** and year **1987**); **m** for moderate ($F_{10.7}$=**130.** and year **1984**); or **h** for high ($F_{10.7}$=**210.** and year **1981**).
7	.	The separator between the root of the file name and its extension.
8-9	no or o2	The ion species: **no** for NO^+ or **o2** for O_2^+.
10	e or p	The contents of the file: **e** for EOFs or **p** for OPCs.

Table 5.1 The PIM output file.

Record Number	Element Name	Data Type	Units	Valid Range	Description
1	HEADER1	Character	n/a	n/a	Column labels for record 3.
2	HEADER2	Character	n/a	n/a	Column labels for record 3.
3	YEAR	Integer	Years	1800 to 2100	The calendar year.
	DOY	Integer	Days	1 to 365(366)	The day of the year.
	UT	Real	Seconds	0. to 86399.999…	The Universal Time.
	F10P7	Real	10^{22} W m^{-2} Hz^{-1}	0. to 300.	The instantaneous solar radio flux at 10.7 cm (2800 MHz), $F_{10.7}$.
	KP	Real	n/a	0. to 9.	The geomagnetic activity index, K_p.

	SSN	Real	n/a	0. to 300.	The sunspot number, *SSN*.
4	BLANK1	Character	n/a	n/a	A blank line.
5	BLANK2	Character	n/a	n/a	A blank line.
6	CRDTYPE + GRIDTYPE *10 + PLASPH *100	Integer	n/a	0, 1, 10, 11, 20, 21, 100, 101, 110, 111, 120, or 121	CRDTYPE is a flag identifying the output latitude/longitude coordinate system: **0** for geographic or **1** for corrected geomagnetic. GRIDTYPE is a flag identifying the type of output grid: **0** for a rectangular latitude/longitude grid; **1** for latitude/longitude pairs; or **2** for an azimuth/elevation grid with a ground-based observer. PLASPH is a flag identifying the plasmaspheric model used: **0** for none or **1** for the Gallagher plasmaspheric model. The three flags have been combined into a single value to maintain output-format compatibility with PRISM.
7	HEADER3	Character	n/a	n/a	Column labels for record 9. The column labels depend on element GRIDTYPE.
8	HEADER4	Character	n/a	n/a	Column labels for record 9. The column labels depend on element GRIDTYPE.
For GRIDTYPE=0 only					
9	SLAT	Real	°N	-90. to 90.	The starting latitude of the rectangular latitude/longitude output grid. Its coordinate system is defined by element CRDTYPE.
	SLON	Real	°E	-360. to 360.	The starting longitude of the rectangular latitude/longitude output grid. Its coordinate system is defined by element CRDTYPE.
	ELAT	Real	°N	-90. to 90.	The ending latitude of the rectangular latitude/longitude output grid. Its coordinate system is defined by element CRDTYPE.
	ELON	Real	°E	-360. to 360.	The ending longitude of the rectangular

					latitude/longitude output grid. Its coordinate system is defined by element CRDTYPE.
	NLAT	Integer	n/a	≥ 1	The number of latitudes in the rectangular latitude/longitude output grid.
	NLON	Integer	n/a	1 to 3601	The number of longitudes in the rectangular latitude/longitude output grid.
	DLAT	Real	°N	-180. to 180.; ≠ 0.	The latitude increment of the rectangular latitude/longitude output grid.
	DLON	Real	°E	-360. to 360.; ≠ 0.	The longitude increment of the rectangular latitude/longitude output grid.
For GRIDTYPE=1 only					
9	$LATPR_1$	Real	°N	-90. to 90.	The latitude of the first latitude/longitude pair in the latitude/longitude pairs output grid. Its coordinate system is defined by element CRDTYPE.
	$LATPR_{NPR}$	Real	°N	-90. to 90.	The latitude of the last latitude/longitude pair in the latitude/longitude pairs output grid. Its coordinate system is defined by element CRDTYPE.
	$LONPR_1$	Real	°E	-360. to 360.	The longitude of the first latitude/longitude pair in the latitude/longitude pairs output grid. Its coordinate system is defined by element CRDTYPE.
	$LONPR_{NPR}$	Real	°E	-360. to 360.	The longitude of the last latitude/longitude pair in the latitude/longitude pairs output grid. Its coordinate system is defined by element CRDTYPE.
	NPR	Integer	n/a	1 to 1000	The number of latitude/longitude pairs in the latitude/longitude pairs output grid.
	0	Integer	n/a	0	A dummy placeholder.
	0.	Real	n/a	0.	A dummy placeholder.
	0.	Real	n/a	0.	A dummy placeholder.

For GRIDTYPE=2 only					
9	SAZ	Real	°	-360. to 360.	The starting azimuth of the azimuth/elevation output grid. The orientation of north (*AZ*=0°) depends on the coordinate system defined by element CRDTYPE.
	EAZ	Real	°	-360. to 360.	The ending azimuth of the azimuth/elevation output grid. The orientation of north (*AZ*=0°) depends on the coordinate system defined by element CRDTYPE.
	SEL	Real	°	0. to 90.	The starting elevation of the azimuth/elevation output grid.
	EEL	Real	°	0. to 90.	The ending elevation of the azimuth/elevation output grid.
	NAZ	Integer	n/a	1 to 3601	The number of azimuths in the azimuth/elevation output grid.
	NEL	Integer	n/a	≥ 1	The number of elevations in the azimuth/elevation output grid.
	DAZ	Real	°	-360. to 360.; ≠ 0.	The azimuth increment of the azimuth/elevation output grid.
	DEL	Real	°	-90. to 90.; ≠ 0.	The elevation increment of the azimuth/elevation output grid.
10	OUTTYPE	Integer	n/a	0, 1, or 2	A flag that identifies what quantities are output: **0** for critical frequencies and heights for the ionospheric *E* and F_2 regions, and *TEC*; **1** for *EDP*s; or **2** for *EDP*s, critical frequencies and heights for the ionospheric *E* and F_2 regions, and *TEC*.
For OUTTYPE=0 only					
11	BLANK3	Character	n/a	n/a	A blank line.
For GRIDTYPE=0 and GRIDTYPE=1 only					

12	GLAT	Real	°N	-90. to 90.	The geographic latitude of an output grid point.
	GLON	Real	°E	-360. to 360.	The geographic longitude of an output grid point.
	MLAT	Real	°N	-90. to 90.	The corrected geomagnetic latitude of an output grid point.
	MLON	Real	°E	-360. to 360.	The corrected geomagnetic longitude of an output grid point.
	MLT	Real	Hours	0. to 23.999…	The corrected geomagnetic local time of an output grid point.
13	HEADER5	Character	n/a	n/a	Column labels for record 14.
14	FOF2	Real	MHz	≥ 0.	The critical frequency of the ionospheric F_2 region, f_oF_2, for an output grid point.
	HMF2	Real	km	≥ 0.	The altitude of the peak of the ionospheric F_2 region, h_mF_2, for an output grid point.
	FOF1	Real	MHz	0.	The critical frequency of the ionospheric F_1 region, f_oF_1, for an output grid point. It is always zero because PIM currently cannot calculate ionospheric F_1 region parameters.
	HMF1	Real	km	0.	The altitude of the peak of the ionospheric F_1 region, h_mF_1, for an output grid point. It is always zero because PIM currently cannot calculate ionospheric F_1 region parameters.
	FOE	Real	MHz	≥ 0.	The critical frequency of the ionospheric E region, f_oE, for an output grid point.
	HME	Real	km	≥ 0.	The altitude of the peak of the ionospheric E region, h_mE, for an output grid point.
	VERTTEC	Real	TEC Units	≥ 0.	Vertical *TEC* in the altitude range 90 to 25000 km, for an output grid point.
For GRIDTYPE=2 only					
12	AZ	Real	°	-360. to 360.	The azimuth of an output grid point. The orientation of north (AZ=0°) depends on the coordinate system defined by element

					CRDTYPE.
	EL	Real	°	0. to 90.	The elevation of an output grid point.
	OBSLAT	Real	°N	-90. to 90.	The latitude of the ground observer. Its coordinate system is defined by element CRDTYPE.
	OBSLON	Real	°E	-360. to 360.	The longitude of the ground observer. Its coordinate system is defined by element CRDTYPE.
	0.	Real	n/a	0.	A dummy placeholder.
13	HEADER6	Character	n/a	n/a	Column labels for record 14.
14	FPMAX	Real	MHz	≥ 0.	The plasma frequency of the peak of the slant *EDP*, for an output grid point.
	HMAX	Real	km	≥ 0.	The altitude of the peak of the slant *EDP*, for an output grid point.
	0.	Real	n/a	0.	A dummy placeholder.
	0.	Real	n/a	0.	A dummy placeholder.
	0.	Real	n/a	0.	A dummy placeholder.
	0.	Real	n/a	0.	A dummy placeholder.
	SLANTTEC	Real	TEC Units	≥ 0.	Slant *TEC* in the altitude range defined by the output altitude grid, for an output grid point.
Repeat records 11-14 for each output grid point					
For OUTTYPE=1 only					
11	LABEL1	Character	n/a	Number of altitude points =	A label for element NALT.
	NALT	Integer	n/a	1 to 100	The number of altitudes in the output altitude grid.
12	HEADER7	Character	n/a	n/a	A header for record 13.
13 (Repeat NALT	ALT_i	Real	km	90. to 25000.	The i^{th} altitude in the output altitude grid.

times)					
14	BLANK4	Character	n/a	n/a	A blank line.
For GRIDTYPE=0 and GRIDTYPE=1 only					
15	GLAT	Real	°N	-90. to 90.	The geographic latitude of an output grid point.
	GLON	Real	°E	-360. to 360.	The geographic longitude of an output grid point.
	MLAT	Real	°N	-90. to 90.	The corrected geomagnetic latitude of an output grid point.
	MLON	Real	°E	-360. to 360.	The corrected geomagnetic longitude of an output grid point.
	MLT	Real	Hours	0. to 23.999…	The corrected geomagnetic local time of an output grid point.
For GRIDTYPE=2 only					
15	AZ	Real	°	-360. to 360.	The azimuth of an output grid point. The orientation of north (*AZ*=0°) depends on the coordinate system defined by element CRDTYPE.
	EL	Real	°	0. to 90.	The elevation of an output grid point.
	OBSLAT	Real	°N	-90. to 90.	The latitude of the ground observer. Its coordinate system is defined by element CRDTYPE.
	OBSLON	Real	°E	-360. to 360.	The longitude of the ground observer. Its coordinate system is defined by element CRDTYPE.
	0.	Real	n/a	0.	A dummy placeholder.
16	HEADER8	Character	n/a	n/a	A header for record 17.
17 (Repeat NALT times)	$EDEN_i$	Real	cm^{-3}	≥ 0.	The electron density at the i^{th} altitude in the output altitude grid.

Repeat records 14-17 for each output grid point					
For OUTTYPE=2 only					
11	LABEL2	Character	n/a	Number of altitude points =	A label for element NALT.
	NALT	Integer	n/a	1 to 100	The number of altitudes in the output altitude grid.
12	HEADER9	Character	n/a	n/a	A header for record 13.
13 (Repeat NALT times)	ALT_i	Real	km	90. to 25000.	The i^{th} altitude in the output altitude grid.
14	BLANK5	Character	n/a	n/a	A blank line.
For GRIDTYPE=0 and GRIDTYPE=1 only					
15	GLAT	Real	°N	-90. to 90.	The geographic latitude of an output grid point.
	GLON	Real	°E	-360. to 360.	The geographic longitude of an output grid point.
	MLAT	Real	°N	-90. to 90.	The corrected geomagnetic latitude of an output grid point.
	MLON	Real	°E	-360. to 360.	The corrected geomagnetic longitude of an output grid point.
	MLT	Real	Hours	0. to 23.999…	The corrected geomagnetic local time of an output grid point.
For GRIDTYPE=2 only					
15	AZ	Real	°	-360. to 360.	The azimuth of an output grid point. The orientation of north (*AZ*=0°) depends on the coordinate system defined by element CRDTYPE.
	EL	Real	°	0. to 90.	The elevation of an output grid point.
	OBSLAT	Real	°N	-90. to 90.	The latitude of the ground observer. Its coordinate system is defined by element CRDTYPE.

	OBSLON	Real	°E	-360. to 360.	The longitude of the ground observer. Its coordinate system is defined by element CRDTYPE.
	0.	Real	n/a	0.	A dummy placeholder.
16	HEADER10	Character	n/a	n/a	A header for record 17.
17 (Repeat NALT times)	$EDEN_i$	Real	cm^{-3}	≥ 0.	The electron density at the i^{th} altitude in the output altitude grid.
18	HEADER11	Character	n/a	n/a	A header for record 19.
For GRIDTYPE=0 and GRIDTYPE=1 only					
19	FOF2	Real	MHz	≥ 0.	f_oF_2 for an output grid point.
	HMF2	Real	km	≥ 0.	h_mF_2 for an output grid point.
	FOF1	Real	MHz	0.	f_oF_1 for an output grid point. It is always zero because PIM currently cannot calculate ionospheric F_1 region parameters.
	HMF1	Real	km	0.	h_mF_1 for an output grid point. It is always zero because PIM currently cannot calculate ionospheric F_1 region parameters.
	FOE	Real	MHz	≥ 0.	f_oE for an output grid point.
	HME	Real	km	≥ 0.	h_mE for an output grid point.
	VERTTEC	Real	TEC Units	≥ 0.	Vertical *TEC* in the altitude range 90 to 25000 km, for an output grid point.
For GRIDTYPE=2 only					
19	FPMAX	Real	MHz	≥ 0.	The plasma frequency of the peak of the slant *EDP*, for an output grid point.
	HMAX	Real	km	≥ 0.	The altitude of the peak of the slant *EDP*, for an output grid point.
	0.	Real	n/a	0.	A dummy placeholder.
	0.	Real	n/a	0.	A dummy placeholder.

	0.	Real	n/a	0.	A dummy placeholder.
	0.	Real	n/a	0.	A dummy placeholder.
	SLANTTEC	Real	TEC Units	≥ 0.	Slant *TEC* in the altitude range defined by the output altitude grid, for an output grid point.
Repeat records 14-19 for each output grid point					

Figures

Figure 6.1 An example **path_nam.txt** input file for DOS/Windows/NT platforms.

```
1
e:\pim\cgmdb\
e:\pim\usudb\unform\
e:\pim\mlfdb\unform\
e:\pim\llfdb\unform\
e:\pim\lmedb\unform\
e:\pim\ursidb\unform\
```

Figure 6.2 An example input stream **testcas1.in**.

```
1992
270
1200
0
0
TESTCAS1.out
+
-
1
142
100
3.0
0
1
0
G
17
-80.0
10.0
1
0.0
1.0
1
0.0,0.0
0.0
0.0
24
0.0
15.0
10
0.0
10.0
50
   90   95  100  105  110  115  120  125  130  135  140  145  150  160  170  180
  190  200  210  220  230  240  250  260  270  280  290  300  320  340  360  380
  400  450  500  550  600  650  700  750  800  850  900 1000 1100 1200 1300 1400
 1500 1600
N
```

Figure 6.3 An example output stream **testcas1.log**.

```
*****************************
 = PIM 1.7   13-January-1998 =
 *****************************
Corrected geomagnetic coordinate database location:
e:\pim\cgmdb\
Parameterized USU database location:
e:\pim\usudb\unform\
Parameterized mid-latitude F-region database location:
e:\pim\mlfdb\unform\
Parameterized low-latitude F-region database location:
e:\pim\llfdb\unform\
Parameterized low- and mid- latitude E-region database location:
e:\pim\lmedb\unform\
URSI coefficients database location:
e:\pim\ursidb\unform\
Reading parameterized low- and mid- latitude E-region model database...
Reading parameterized low-latitude F-region model database...
Reading parameterized mid-latitude F-region model database...
Reading parameterized USU model database...
Reading URSI-88 coefficients database...
SSN = 100: IREC =         4217
Running...
Output status:[Lat,Lon]=[-80.00,   0.00]
Output status:[Lat,Lon]=[-70.00,   0.00]
Output status:[Lat,Lon]=[-60.00,   0.00]
Output status:[Lat,Lon]=[-50.00,   0.00]
Output status:[Lat,Lon]=[-40.00,   0.00]
Output status:[Lat,Lon]=[-30.00,   0.00]
Output status:[Lat,Lon]=[-20.00,   0.00]
Output status:[Lat,Lon]=[-10.00,   0.00]
Output status:[Lat,Lon]=[  0.00,   0.00]
Output status:[Lat,Lon]=[ 10.00,   0.00]
Output status:[Lat,Lon]=[ 20.00,   0.00]
Output status:[Lat,Lon]=[ 30.00,   0.00]
Output status:[Lat,Lon]=[ 40.00,   0.00]
Output status:[Lat,Lon]=[ 50.00,   0.00]
Output status:[Lat,Lon]=[ 60.00,   0.00]
Output status:[Lat,Lon]=[ 70.00,   0.00]
Output status:[Lat,Lon]=[ 80.00,   0.00]
PIM successfully completed.
```

Figure 6.4 An example output file **testcas1.out**.

```
YEAR    DAY   UT (sec)   F10.7   Kp    Solar Sunspot Number
-------------------------------------------------------------
 1992    270   43200.0    142.0   3.0     96.20


    0
      Latitude       Longitude        Latitude  Longitude Latitude Longitude
  Starting Ending   Starting Ending       Step   Step        Delta   Delta
  -80.00    80.00     0.00     0.00       17       1         10.00    1.00
  0

   -80.00    0.00  -67.00   36.20    9.71
   FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
   6.62  300.39    0.00    0.00    3.21  121.01   18.77

   -70.00    0.00  -60.10   45.00   10.29
   FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
   8.69  286.86    0.00    0.00    1.14  110.00   24.61

   -60.00    0.00  -53.90   50.50   10.66
   FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
   9.98  285.57    0.00    0.00    2.56  110.00   41.65

   -50.00    0.00  -48.20   54.60   10.93
   FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
  10.65  280.45    0.00    0.00    3.26  110.00   51.81

   -40.00    0.00  -42.40   58.30   11.18
   FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
  11.01  276.68    0.00    0.00    3.43  110.00   53.97

   -30.00    0.00  -35.90   62.10   11.43
   FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
  11.06  265.92    0.00    0.00    3.53  110.00   51.70

   -20.00    0.00  -28.80   65.80   11.68
   FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
  10.96  261.12    0.00    0.00    3.63  105.00   50.30

   -10.00    0.00  -21.10   69.00   11.89
   FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
  11.20  288.27    0.00    0.00    3.74  105.00   53.70

     0.00    0.00  -12.70   71.10   12.03
   FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
  11.45  359.84    0.00    0.00    3.83  105.00   65.18

    10.00    0.00   -3.30   72.40   12.12
   FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
  11.07  457.92    0.00    0.00    3.88  105.00   86.59

    20.00    0.00    7.40   73.40   12.19
   FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
  13.19  435.52    0.00    0.00    3.89  105.00  106.11

    30.00    0.00   19.70   74.60   12.27
   FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
  11.02  340.25    0.00    0.00    3.85  105.00   55.90
```

Figure 6.4 An example output file **testcas1.out** (continued).

```
 40.00    0.00   33.60   76.40   12.39
FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
9.13  290.30    0.00    0.00    3.70  105.00   34.00

 50.00    0.00   46.50   79.10   12.57
FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
7.99  259.28    0.00    0.00    3.49  105.00   23.63

 60.00    0.00   57.90   83.40   12.85
FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
7.11  293.88    0.00    0.00    1.68  105.00   18.12

 70.00    0.00   68.40   91.10   13.37
FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
6.51  310.94    0.00    0.00    1.47  110.00   16.23

 80.00    0.00   77.70  110.30   14.65
FoF2,  HmF2,    FoF1,  HmF1,    FoE,  HmE    TEC
6.04  334.38    0.00    0.00    1.14  120.00   18.20
```