UNIRAD v 3.0

Trapped Radiation Software

User Manual

D. Heynderickx, M. Kruglanski, J. Lemaire

Belgian Institute for Space Aeronomy
Ringlaan 3, B-1180 Brussel, Belgium

Errata for Version 3.0

- Table 2.1 (page 17): parameter `IATTI`, value 2, change to: z axis parallel to velocity vector, x axis in orbital plane pointing away from Earth, y axis perpendicular to orbital plane pointing south.

- Page 26: The Mead & Fairfield model has one input parameter, i.e. K_{p}. The corresponding `NAMELIST` parameter is `VALUE_KP`.

- Page 85: Add a line with `$END` before the line `$SHIELDOSE`.

Printing history

Version 2.0:	March 1996
Version 3.0:	November 1996

Contents

List of Figures

List of Tables

Preface

UNIRAD is a software package developed by and for ESA for evaluation of the radiation fluences and doses expected in a spacecraft from a definition of the mission characteristics.

The UNIRAD package consists of a series of programs: SAPRE, BLXTRA, TREP, TREP-POS, ANISO, ANISOPOS, TREPAVE, SHIELDOSE, EQFRUX, and EQFRUXGA. There also is a plotting program, UNIRAD.PRO, to produce graphs of the results. Both the individual programs in UNIRAD and the interfaces between them have been modified substantially in the course of the TREND projects (ESTEC contracts nrs. 9011/88/NL/MAC and 9828/92/NL/FM). This document is intended as a user manual to the present version of the software (v 3.0).

The installation guide is contained in Chapter 1.

Chapter 2 contains the descriptions of the individual programs in UNIRAD. The purpose of each software package is stated, and their interaction is documented. A detailed description is given of the input files necessary for running the software and the output files that are produced.

In Chapter 3, we present the results of a sample run of the complete UNIRAD suite, in the form of a series of plots.

Chapter 4 describes the plotting routine UNIRAD.PRO.

Appendix A contains the listing of a sample NAMELIST file with all NAMELIST parameters and their default values. Appendix B contains the format description of the common interface file.

Support

If you have a question about or problems with UNIRAD first look in this documentation and make sure you followed the proper installation instructions. If you cannot find the answer:

- consult the WWW FAQ page http://magnet.oma.be/unirad/faq.html;

- contact D. Heynderickx by telephone (+32-2-3730417), fax (+32-2-3748423) or E-mail (d.heynderickx@oma.be).

Customized on-site training for end users and trainers can be organised.

Disclaimer

The UNIRAD software provides a means of conveniently running the various models described in this document. No guarantee is given concerning the capability of the models to represent actual conditions in the near Earth environment.

No liability for consequential damages: in no event shall BISA or ESA be liable for any damages whatsoever (including without limitation, damages for loss of business profits, business interruption, loss of business information, or any other pecuniary loss) arising out of the use of or inability to use the UNIRAD software, even if BISA and/or ESA have been advised of the possibility of such damages.

Copyright statements

You may not rent, lease or sell the software or the accompanying written materials.

The UNIRAD v 3.0 software was developed under ESA contract by the Belgian Institute for Space Aeronomy (BISA), Brussels, Belgium. The software was completely reorganised and restructured into its present form by BISA.

The trapped particle models AP-8 and AE-8 were developed by NASA and are distributed by NSSDC (Vette 1991b).

The SHIELDOSE program was developed by NBS (Seltzer 1979, 1980) and is available from RSIC, NEA or NSSDC. EQFRUX and EQFRUXGA were developed by and are available from JPL (Tada et al. 1982). SHIELDOSE, EQFRUX and EQFRUXGA have been modified substantially in order to incorporate them into UNIRAD. However, the main algorithms have not been altered.

PostScript is a registered trademark of Adobe Systems, Inc..

MS, MS-DOS and MS-Windows are registered trademarks of Microsoft Corporation.

VMS is a registered trademark of Digital Equipment Corporation.

Acknowledgements

We are indebted to H.D.R. Evans and E.J. Daly for their contributions to the debugging of and improvements to the software. We thank L. Desorgher for carefully reading the manuscript and extensive testing of the software.

List of acronyms and abbreviations

AE-8 NASA empirical models for the omnidirectional flux of trapped electrons (Vette 1991)

AP-8 NASA empirical models for the omnidirectional and unidirectional flux of trapped protons (Sawyer & Vette 1976)

AL Anomalously Large

ASCII American Standard Code for Information Interchange

BISA Belgian Institute for Space Aeronomy

CIRA COSPAR International Reference Atmosphere

COSPAR COmmittee for Space Research

CRRES Combined Release and Radiation Effects Satellite

DEC Digital Equipment Corporation

DGRF Definitive Geomagnetic Reference Field

DOS Disk Operating System

ESA European Space Agency

GEI Geographic Equatorial Inertial

IBM International Business Machines

IDL Interactive Data Language

IGRF International Geomagnetic Reference Field

JPL Jet Propulsion Laboratory

LANL Los Alamos National Laboratory

MS MicroSoft

NASA National Aeronautics and Space Administration

NEA Nuclear Energy Agency

NBS National Bureau of Standards

NSSDC National Space Science Data Center

PC Personal Computer

PV Precision Visuals

RAM Random Access Memory

RSIC Radiation Shielding Information Center

SAA South Atlantic Anomaly

TREND Trapped Radiation ENvironment Development

VAX Virtual Address eXtension

VMS Virtual Memory System

List of symbols

α Pitch angle

B Geomagnetic field intensity

B_{m} Mirror point geomagnetic field intensity

B_0 Equatorial geomagnetic field intensity

E Particle energy

j Differential particle flux

J Integral particle flux

L McIlwain's (1961) parameter

Changes from previous versions

The `UNIRAD` software has been modified substantially by BISA. The modifications fall into two categories:

- internal modifications in the code which are transparent to the user and are not described here;

- external modifications including:

    - changes in `NAMELIST` parameters;
    - changes in the format of the input and output files;
    - changes in the user interface;
    - extensions of the programs;
    - additions of programs.

Chapter 2 contains listings of all `NAMELIST` parameters and descriptions of all output files. The extensions of the file names of all input and output files have been shortened to three characters.

Changes in NAMELIST parameters

Changes from v 2.0

The `NAMELIST`s `ANISO` and `ANISOPOS` were added. The modifications to the existing `NAMELIST`s are listed below.

SAPRE   The following parameters have been modified: `IAE`, `DT1`, `DT2`, and `DH2`. The parameters `PP` and `XP` were deleted. `ALT`, `PB`, `PM`, `XB`, `XM`, `IATTI`, and `PA` were added.

BLXTRA   The parameter `CUTOFF` has been modified.

TREP   The default values of the following parameters have been modified: `FLAMOD` and `SAAROT`.

Changes from older versions (prior to v 2.0)

The `NAMELIST`s for all programs have been modified. The parameters which have been changed, deleted or added are listed below. Chapter 2 contains detailed descriptions of all `NAMELIST` parameters. Modification of a parameter may mean a change in its effect, a change in its name, a change in its default value, or a change in its data type.

The `NAMELIST` parameters are now reset to their default values for each trajectory in the `NAMELIST` file.

The `NAMELIST` `TEXT` has been deleted. The parameter `TITLE` has been moved to the `NAMELIST` `SAPRE`.

SAPRE   The following parameters have been modified: `HAPO`, `HPER`, `INCL`, `TANO`, `OEYEAR`, `OEMON`, `OEDAY`, `ORBNUM`, and `PRINT`. The parameters `TAPE`, `OGEN` and `EPHMR` were deleted. `TITLE`, `TYPE`, `AMJD`, `IAE`, `IG50`, `A`, `E`, `LONGAPO`, `LONGPER`, `OELONG`, `SOLTIM`, and `IORBEL` were added.

BLXTRA   The following parameters have been modified: `PRINTC`, `BLTIME`, `MODEL`. The parameters `IPLOT` and `DEBUG` were deleted. `IASCII`, `CUTOFF`, `OUTER`, `VALUE_KP`, `DEN`, `VEL`, and `DST` were added.

TREP   The following parameters have been modified: `KPRINT`, `ILTV`, `ENERFL`. The parameters `IPLOT` and `IBUG` were deleted. `TREMOD`, `FLAMOD`, `FLSTAR`, `RAU`, `IFORM`, `SAAROT`, `OMNI`, `NENERP`, `NENERE`, `NENERS`, `ISPEC`, and `ISPSHT` were added.

TREPAVE TREPAVE now requires the presence of the NAMELIST TREPAVE in the NAMELIST file PROJECT.NML.

SHIELDOSE The following parameters have been modified: IPRNT, IPLOT. ISHLD was added.

EQFRUX and EQFRUXGA The parameter IPRINT has been modified.

Changes in input files

In older versions (before v 2.0), SAPRE, BLXTRA, TREP, and SHIELDOSE read data files with model parameters. These data have now been linked in with the programs so that the only input now needed for UNIRAD consists of a single user written NAMELIST file.

Changes in output files

Changes from v 2.0

The format of the common interface file PROJECT.INT generated by SAPRE and used as input by BLXTRA and TREP has changed: see Appendix B for a description of the file format. The interface file now is an ASCII file. A utility program INT2ASCI is provided to rewrite v 2.0 interface files to the new format. The extension of the TREPAVE output file has been changed from .TAI to TRA.

Changes from older versions (prior to v 2.0)

The format of the common interface file PROJECT.INT generated by SAPRE and used as input by BLXTRA and TREP has changed. The format of the interface file PROJECT.TRI, generated by TREP, TREPPOS, and TREPAVE, and used as input by SHIELDOSE, EQFRUX, and EQFRUXGA, also has been modified. The new format is described in Table 2.9.

   The layout of most printable report files has been improved. The format of the plot files has changed completely to a more compact and structured form.

   New output files have been added to SAPRE, BLXTRA, and TREP.

Changes in the user interface

The user interface has been completely reworked for v 2.0. The programs now are run by typing the program name and, optionally, a project name as command line argument. See Chapter 1 for more details. From v 3.0 on, the user is no longer warned when an output file will be overwritten in order to limit the number of user prompts.

Extensions of programs

Changes from v 2.0

SAPRE   SAPRE will now generate a file with attitude information for each orbital point when the NAMELIST parameter IATTI is set to 1, 2, or 3 (the values select between three pre-defined satellite coordinate systems).

Changes from older versions (prior to v 2.0)

BLXTRA   In older versions, the calculation of (B, L) coordinates was performed by a program called SHELLG. This program has been replaced by BLXTRA, which has a more effective and accurate field line tracing algorithm and has several external magnetic field models built in.

TREP   The trapped proton models AP-8 MIC and AP-8 MAC in TREP have been replaced by AP-8 MIN and AP-8 MAX, respectively. An electron model based on geostationary LANL data has been added. The Feynman model (Feynman & Gabriel 1990) for solar flare proton fluences is now available in TREP.

TREP now calculates (B, L) coordinates internally with the appropriate magnetic field models and parameters. It is no longer necessary to run BLXTRA prior to TREP. The (B, L) values calculated by TREP are written to the commmon interface file PROJECT.INT, overwriting (B, L) values already present if BLXTRA has been called prior to TREP.

New programs

Changes from v 2.0

The programs ANISO and ANISOPOS have been added to calculate the trapped proton anisotropy.

Changes from older versions (prior to v 2.0)

The program EQFRUXGA has been added. It has the same function as EQFRUX, but is intended for GaAs solar cells.

A plotting routine, UNIRAD.PRO, has been added. It is written in the IDL language.

Chapter 1

Overview and installation guide

The `UNIRAD` suite of programs provides information about the radiation environment in an arbitrary Earth orbit, predicting satellite exposures to particle fluxes, the resulting radiation dose, and the resulting damage-equivalent fluences for solar cell degradation calculations.

This chapter outlines the system and its constituent programs, its use, and the installation of the software. The programs outlined here are upgrades and extensions of the programs in previous versions, as well as new programs.

1.1 Overview of UNIRAD

An orbit analysis generally consists of running one or more of the `UNIRAD` component programs, with communication between the programs via interface files. Both graphical and tabular output is provided.

The user input required by the system consists of a single FORTRAN `NAMELIST` file containing the project header, the orbital parameters, solar activity conditions, plotting and printing options, ... It is possible to run a series of projects within the same `NAMELIST` file. All parameters in the `NAMELIST` file are assigned default values when not specified, and are reset for each project.

From the orbit parameters, the system will generate a detailed trajectory, magnetic coordinates, integral and differential proton and electron fluences, doses for three shield geometries in four detector materials, and solar cell degradation information, in both printed and graphical form.

The `UNIRAD` package consists of the following programs:

- `SAPRE`: orbit generator which produces a data file used by the next two programs in the package;

- `BLXTRA`: calculates the geomagnetic coordinates (B, L) from the geographic coordinates generated by `SAPRE`;

- `TREP`: determines the radiation flux for the geographic coordinates generated by `SAPRE` from the NASA trapped radiation models AP-8 and AE-8 and determines the solar proton flux over the mission. It produces a data file with the energy spectra of trapped protons and electrons and of solar protons;

- `TREPPOS`: calculates the trapped radiation flux for pairs of (B, L) or $(B/B_0, L)$ coordinates interactively input by the user. It produces a data file with the energy spectra of trapped protons and electrons;

- `TREPAVE`: averages the spectra generated by `TREP` or `TREPPOS` for different orbits;

- `ANISO`: transforms the trapped proton omnidirectional integral flux produced by TREP into unidirectional integral and differential fluxes, taking into account pitch angle and azimuthal dependence. The user can define a set of look directions with respect to a satellite reference frame. The resulting fluxes are averaged over the orbit.

- `ANISOPOS`: provides the angular distribution (i.e. pitch angle and azimuthal dependence) of the unidirectional integral or differential flux at a given geographic location.

- `SHIELDOSE`: reads the energy spectra resulting from `TREP`, `TREPAVE`, or `TREPPOS` and converts them to radiation dose-depth curves for different detector materials and simple shielding geometries;

- `EQFRUX`: determines 1 MeV electron damage equivalent fluences from the `TREP` spectra to evaluate degradation of Si solar cells;

- `EQFRUXGA`: idem as `EQFRUX`, but for GaAs solar cells.

- `UNIRAD.PRO`: a set of IDL routines to produce graphical output.

The flow diagram of the `UNIRAD` package is represented in Fig. 1.1. This diagram illustrates the interdependence of the various programs making up `UNIRAD`. Except for the interface files, the output files are not shown in Fig. 1.1.

1.2   Inputs and outputs

A complete `UNIRAD` radiation analysis requires only one user input file: the `NAMELIST` file `PROJECT.NML`, where `PROJECT` represents the project name to be chosen by the user. If the programs are run in the proper order, all successively needed input files are generated by `UNIRAD`. Alternatively, the user may supply his own input files according to the specifications given in Chapter 2.

All programs in `UNIRAD` (except `TREPPOS` and `ANISOPOS`) require input from the `NAMELIST` input file `PROJECT.NML`. In Sect. 1.2.1 a general description of `NAMELIST` files and syntax is given. The `NAMELIST` parameters for the different programs in `UNIRAD`
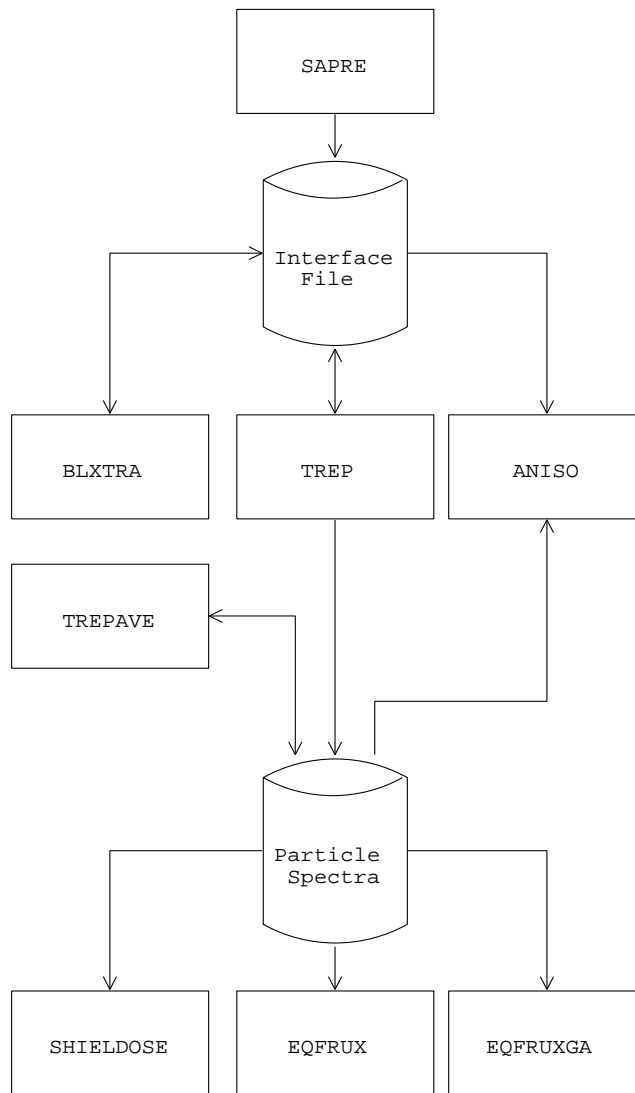
Figure 1.1. Flow diagram of UNIRAD

are listed and discussed in Chapter 2. The input files necessary for the programs in UNIRAD are listed in Table 1.1 together with the programs that generate them, and their format.

The output generated by UNIRAD consists of ASCII files with file names of the form PROJECT.XXX, where XXX identifies the program generating the file and the type of information in the file. PROJECT must be chosen so that PROJECT.XXX represents a valid file name. A list of all output files is presented in Sect. 1.2.2.

1.2.1   Namelist file input

UNIRAD requires a composite NAMELIST file named PROJECT.NML, where PROJECT

Table 1.1. List of input files for the programs in UNIRAD. Every program (except TREPPOS and ANISOPOS) also requires the NAMELIST file PROJECT.NML

Program name	File name
SAPRE	None
BLXTRA	PROJECT.INT
TREP	PROJECT.INT
TREPPOS	None
TREPAVE	PROJECT.TRI
ANISO	PROJECT.INT, PROJECT.ATT, PROJECT.SPP
ANISOPOS	PROJECT.TRI (optional)
SHIELDOSE	PROJECT.TRI
EQFRUX	PROJECT.TRI
EQFRUXGA	PROJECT.TRI

represents the project name to be chosen by the user. The NAMELIST file should contain a NAMELIST for each program to be executed, except for TREPPOS. NAMELISTs do not have to contain all input variables and may even be empty, in which cases default values will be chosen, but must be present in the file. NAMELISTs for more than one project may be put in the same file. The general syntax rules for NAMELIST input follow.

Namelist file input consists of a record delimited by the dollar sign $ (except on PC, where an ampersand & should be used) which starts in the second column (the first column must be blank).

Generally, NAMELIST input has the form:

 $NAME PARAMETER=VALUE [, PARAMETER=VALUE, ...] $[END]

where

- $ is the special dollar sign symbol that indicates the beginning and end of input and the start of a namelist section. Optionally an ampersand (&) may be used instead of a dollar sign (mandatory on PC).

- NAME is the name of the namelist file section. This name may not contain any spaces or tabs.

- PARAMETER is the name of one of the input parameters of the program for which the namelist file provides the data. The parameter list does not have to be exhaustive, i.e. not all parameters have to be given. A PARAMETER name that does not correspond to any parameter defined by the NAMELIST in the calling program will generate an error.

- VALUE is a constant or list of constants.

- `END` is an optional part of the last delimiter. On PCs, a `NAMELIST` should be terminated with a slash (/).

The following syntax rules for namelist file input must be obeyed.

1. The `NAMELIST` name `NAME` may not contain spaces or tabs.

2. The `PARAMETER` names appearing on the left-hand side of the equals sign in a value assignment may not contain spaces or tabs.

3. Each constant that appears in a `VALUE` assignment must be one of the following types:

   - integer, in the form of a whole number with no decimal point. An integer may have a leading sign (+ or −). Leading zeros are ignored;

   - real, in the form of a number with a decimal point or an exponent (or both). An exponent letter (`E` or `D`) must be followed by an exponent;

   - character, consisting of at least one printable character or string of characters delimited by apostrophes (`'`). The value does not include delimiting apostrophes but does include all spaces and tabs. Apostrophes in the input string are represented by two consecutive apostrophes (`''`). Quotation marks (`"`) cannot be used in place of apostrophes.

4. Valid separators in lists of values are spaces, tabs and commas. Except within character strings (enclosed within apostrophes) any number of consecutive spaces or tabs is equivalent to one space.

5. A null value is specified by two consecutive commas. Spaces before and after commas are ignored. A null value may be entered for a complete value only, not for part of it.

6. The equals sign in a value assignment may be preceded or followed by a space or tab. Multiple spaces or tabs are treated as one space.

7. Vector elements can be specified in the form `ARRAY(n) = VALUE`.

A sample namelist file `DEFAULT.NML` containing the default values for the namelist parameters for all `UNIRAD` programs is provided with the distribution and listed in Appendix A.

Each program in `UNIRAD` will process all `NAMELIST`s intended for its use. It is the user's responsability to ensure that each `NAMELIST` corresponds to the intended program run. When a `NAMELIST` is missing, the program will terminate but will not generate an error.

Table 1.2. List of output files generated or modified by the programs in UNIRAD. The first column identifies the output file. The second column lists the NAMELIST parameters affecting the creation or format of the file. "Always" means that the file is always created or modified. The third column describes the contents of the files.

File name	NAMELIST parameter	Description
	SAPRE	
PROJECT.INT	Always	Common interface file
PROJECT.SAP	Always, IPRINT, IBUG	Ephemeris
PROJECT.ATT	IATTI	Attitude interface file for ANISO
PROJECT.ORB	IPRINT, IORBEL	Orbital Elements
	BLXTRA	
PROJECT.INT	Always	Common interface file
PROJECT.BLP	Always, IPRINT	Report file
PROJECT.BLA	IASCII	Ephemeris and magnetic field components
	TREP	
PROJECT.INT	Always	Common interface file
PROJECT.TRI	INTERF	Interface file for TREPAVE, SHIELDOSE, EQFRUX, EQFRUXGA
PROJECT.TRP	Always, IPRINT	Report file
PROJECT.SPP	ISPEC	Ephemeris and full proton spectrum (serves as interface file for ANISO)
PROJECT.SPE	ISPEC	Ephemeris and full electron spectrum
PROJECT.TRW	ISPSHT	Fluence spectrum in spreadsheet format
	TREPPOS	
PROJECT.TRI	Menu option	Interface file for TREPAVE, ANISO, ANISO-POS, SHIELDOSE, EQFRUX, EQFRUXGA
PROJECT.TRP	Always	Report file
	TREPAVE	
PROJECT.TRA	Always	Averaged fluence spectra
	ANISO	
PROJECT.TRD	Always	Orbit-averaged directional proton spectra
PROJECT.SPD	IFULL	Full directional proton spectra

Table 1.2. (continued)

File name	NAMELIST parameter	Description
ANISOPOS		
PROJECT.TRD	Always	Directional proton spectra
PROJECT.TRI	Always	Omnidirectional proton spectra
SHIELDOSE		
PROJECT.SHP	Always, IPRINT, ISHLD	Report file
PROJECT.SHG	IPLOT, ISHLD	Input file for plotting routines
EQFRUX		
PROJECT.EQP	Always, IPRINT	Report file
PROJECT.EQG	IPLOT	Input file for plotting routines
EQFRUXGA		
PROJECT.EGP	Always, IPRINT	Report file
PROJECT.EGG	IPLOT	Input file for plotting routines

1.2.2   Output files generated by UNIRAD

UNIRAD produces different types of output files. SAPRE creates the common interface file PROJECT.INT, which is used as input by BLXTRA and TREP, and the attitude interface file PROJECT.ATT (optional) for ANISO. Each program (except TREPAVE, ANISO, and ANISOPOS) creates an ASCII file named PROJECT.XXP, which contains a printable overview of the main input parameters and the results. In addition, ASCII files named PROJECT.XXG are optionally produced by SHIELDOSE, EQFRUX, and EQFRUXGA. These latter files can be used by plotting software to produce plots of the results. Finally, TREP and TREPPOS optionally write the interface file PROJECT.TRI which serves as the main input for SHIELDOSE, EQFRUX, and EQFRUXGA.

Table 1.2 lists all output files together with the program that generates them, their format, and the NAMELIST parameters that control their creation or format, if applicable.

In Chapter 2 a description is presented of the functioning of the UNIRAD programs, of their inputs and outputs, and of the NAMELISTs that drive them.

1.3   Installation

This section contains the installation instructions for the platforms and operating systems that are supported.

1.3.1   Distribution format

The software distribution consists of:

- a set of executables for the platform and operating system specified in the purchase order;

- a sample `NAMELIST` file which contains the default values for all input parameters (`DEFAULT.NML`);

- a sample `NAMELIST` file which contains the values used to produce the graphics in Chapter 4 (`TEST.NML`);

- the plotting program `UNIRAD.PRO`;

- a printed copy of this manual.

- a PostScript version of this manual.

The programs are run by changing to a user directory (current directory) and typing the program name, optionally followed by the project name `PROJECT`. The project name may contain a path name, in which case it is not necessary to set the current directory to the specified path. If `PROJECT` was not given on the command line, the program will prompt for it to be input. The current directory (or the path in the project name) should contain the `NAMELIST` file of the current project. The output files generated by `UNIRAD` will be placed in the same directory. No other data files are needed since all model data have been linked into the executables. In this respect, the `UNIRAD` component programs are standalone.

1.3.2   Installation procedure

The product is delivered in the form of a TAR file, except for the MS-DOS version. The following sections contain platform specific instructions.

VMS

To install the software, proceed as follows:

1. Create a directory and make it the current directory, e.g.:
   `CREATE/DIRECTORY MY$DISK:[MY_ID.UNIRAD]` `Return`
   `SET DEFAULT MY$DISK:[MY_ID.UNIRAD]` `Return`

2. Copy the TAR file from the distribution medium and extract it:
   `TAR -XVF UNIRAD.TAR` `Return`
   This will extract the executables and other files in the current directory and subdirectories.

3. Modify the file `MY$DISK:[MY_ID.UNIRAD]UNIRAD.COM` so that it reflects your installation. Add the modified file to your `LOGIN.COM` file. Alternatively, add a line to your `LOGIN.COM` file that executes `UNIRAD.COM`, or execute `UNIRAD.COM` before a `UNIRAD` run.

4. Add the `UNIRAD` directory to the IDL input directory path.

To run the programs:

1. Change directory to a user directory, e.g.:
   `SET DEFAULT MY$DISK:[MY_ID.PROJECT]` `Return`
   (if necessary, create this directory first).

2. Using a text editor, create a `NAMELIST` file (e.g. `PROJECT.NML`) and enter the `NAMELIST` parameters you will need for the run.

3. Run a `UNIRAD` program by typing its name and (optionally) the project name `PROJECT` (with the full path if the `NAMELIST` file is not in the current directory) and pressing the `Return` key. Run the programs in the proper order so that all necessary files are created.

4. Run `UNIRAD.PRO` in an IDL or PV-Wave session by typing 'UNIRAD', optionally followed by a comma and the project name in single quotes.

UNIX

The UNIX version is shipped as a TAR file `UNIRAD.TAR`. To install the software, proceed as follows:

1. Create a directory and make it the current directory, e.g.:
   `mkdir unirad` `Return`
   `cd unirad` `Return`

2. Copy the TAR file from the distribution medium and extract it:
   `tar -xvf unirad.tar` `Return`
   This will extract the executables and other files in the current directory and subdirectories.

3. Change the mode of the files to executable:
   `chmod +x *` `Return`

4. Add the `UNIRAD` directory to your path.

5. Add the `UNIRAD` directory to the IDL input directory path.

To run the programs:

1. Change directory to a user directory, e.g.:
   `cd project` `Return`
   (if necessary, create this directory first).

2. Using a text editor, create a `NAMELIST` file (e.g. `project.nml`) and enter the `NAMELIST` parameters you will need for the run.

3. Run a `UNIRAD` program by typing its name and (optionally) the project name `project` (with the full path if the `NAMELIST` file is not in the current directory) and pressing the `Return` key. Run the programs in the proper order so that all necessary files are created.

4. Run `UNIRAD.PRO` in an IDL or PV-Wave session by typing 'unirad', optionally followed by a comma and the project name in single quotes.

All file extensions used and generated by `UNIRAD` are lowercase.

MS-DOS

The MS-DOS version is shipped as self-extracting compressed files on two or more disks, depending on the disk format specified in the purchase order. Each disk contains a self-extracting executable.

To install the software, proceed as follows:

1. Make backup copies of the distribution disk(s) using the `COPY` or `DISKCOPY` command in MS-DOS.

2. Create a directory on the hard disk and make it the current directory, e.g.:
   `md c:\unirad` `Return`
   `cd c:\unirad` `Return`

3. Insert the disks in the drive one by one and type
   `a:\install` `Return`
   for each disk (assuming your floppy drive is "a", otherwise substitute the appropriate drive letter). This will extract the executables and other files into the current directory.

4. Add the installation directory to the `PATH` statement in the `AUTOEXEC.BAT` file.

5. Add the `UNIRAD` directory to the IDL input directory path.

To run the programs:

1. Using a text editor, create a `NAMELIST` file (e.g. `PROJECT.NML`) in a user directory (if necessary, create this directory first) and enter the `NAMELIST` parameters you will need for the run.

2. Change directory to the user directory, e.g.:
   `cd c:\project` `Return`

3. Run a `UNIRAD` program by typing its name and (optionally) the project name `PROJECT` (with the full path if the `NAMELIST` file is not in the current directory) and pressing the `Return` key. Run the programs in the proper order so that all necessary files are created.

4. Run `UNIRAD.PRO` in an IDL or PV-Wave session by typing 'UNIRAD', optionally followed by a comma and the project name in single quotes.

   The programs were compiled for MS-DOS with MS-Fortran. The executables require an additional program `DOSXMSF.EXE` (from Phar Lap), which is included in the distribution. You should check whether another version of this utility is already present on your hard disk. If this is the case, you may delete the `DOSXMSF.EXE` file included with the `UNIRAD` distribution.

MS-Windows 3.1

To install the software, proceed as for the MS-DOS installation from an MS-DOS window. The MS-Windows distribution contains additional files: a program group file `UNIRAD.GRP` and one `.PIF` file per program. Add the following line to the `PROGMAN.INI` file (this file resides in your main MS-Windows directory) under the heading `[Groups]`: `Groupnn=C:\UNIRAD\UNIRAD.GRP` (or the directory you choose if not `\UNIRAD`), where `nn` refers to the first group number not in use. If you did not install the software in directory `md c:\unirad`, you have to modify the working directory of each `.PIF` file with the MS-Windows `PIF File Editor`.

   The programs can be run by double-clicking on the icons or in an MS-DOS window. Any Windows editor can be used to edit `NAMELIST` files. `UNIRAD.PRO` is run in an IDL or PV-Wave session by typing 'UNIRAD', optionally followed by a comma and the project name in single quotes.

Chapter 2

Description of programs in UNIRAD

The `UNIRAD` package consists of an orbit generator, geomagnetic field models, trapped particle models, proton anisotropy models, solar flare models, a code for computing doses in simple geometries and codes for computing solar cell damage-equivalent fluences.

This chapter contains detailed descriptions of the models and programs and of all input files and output files.

2.1   Overview

The orbit generator provides geodetic coordinates on a user defined orbit and, optionally, attitude vectors. The geomagnetic field model program computes the corresponding (B, L) coordinates. The geodetic coordinates also serve as input for the trapped particle model program, which (independently from the geomagnetic field model program) computes $(B/B_0, L)$ coordinates and accesses the standard NSSDC trapped particle models (Vette 1991b).

The trapped particle models used are AP-8 for protons (Sawyer & Vette 1976) and AE-8 for electrons (Vette 1991a). These models provide integral omnidirectional proton and electron fluxes as a function of particle energy E, L and B/B_0. In addition, directional proton fluxes can be computed. The program uses these flux data to compute integral and differential fluxes as a function of orbit time and generates orbit-averaged spectra.

Solar flare particle events are treated statistically. Models of mean, worst-case and anomalous solar flare proton spectra based on King (1974) are used together with the Burrell distribution. Depending on the mission duration and timing, a number of flares are predicted and the resulting total fluence spectrum is produced. The model of Feynman & Gabriel (1990) is also implemented.

Dose in simple aluminium geometries is calculated using the resulting fluence spectra and a data set containing dose/unit fluence generated by runs of a Monte Carlo transport code (Seltzer 1979). The Monte Carlo method follows the transport of populations of particles explicitly, including energy loss, scattering and the generation of secondary particles. Doses due to electrons, electron induced Bremsstrahlung, and protons are included in the data sets.

Finally, damage-equivalent fluences for solar cell degradation are computed for Si and GaAs.

Facilities are available for averaging orbital results to produce mission averages and for producing spectra at single (B, L) or $(B/B_0, L)$ locations.

2.2   SAPRE

2.2.1   Function and structure

SAPRE computes trajectory osculatory orbital elements using a numerical Runge-Kutta integration method. The numerical Runge-Kutta orbit generator can be used for low altitude orbits, geostationary orbits, and highly eccentric orbits. It takes into account the oblateness of the Earth, the gravitational attraction of Sun and Moon, air drag (by means of the CIRA atmospheric model) and solar radiation pressure. The independent variable is the eccentric anomaly. Osculatory elements are computed at constant equidistant eccentric anomaly steps.

2.2.2   Input to SAPRE

The input to SAPRE is a description of an orbital trajectory of a space system. Each trajectory is divided into one or more orbital arcs, where each orbital arc consists of a series of orbital points.

The description of the trajectory is contained in the NAMELIST file PROJECT.NML. This file consists of one or more $SAPRE sections. Each $SAPRE section specifies the project title with the NAMELIST parameter TITLE (80 characters including blank spaces). If the title contains blank spaces, it must be enclosed in quotation marks ('). $SAPRE further contains the apogee and perigee altitudes, inclination, ascending node (ω, υ), orbit start time and date, perturbations, and calculation step (in seconds). Other (optional) parameters may be specified, but usually the default values for these parameters are sufficient.

The SAPRE NAMELIST parameters are listed in Table 2.1 with their data type and default value, and a brief description (more detailed descriptions of the main parameters are given below). A sample NAMELIST file containing all parameters and their default values is listed in Appendix A. The parameters not specified in the NAMELIST file are given their default values. A NAMELIST with no parameters will generate one orbit of a trajectory with perigee 300 km, apogee 36,000 km, inclination $0°$.

IPRINT is a flag for writing the ephemeris to the report file PROJECT.SAP. The ephemeris and world map data are generated for a specified constant time step (DT). If IPRINT $= 0$, each ephemeris step is listed. If IPRINT > 0, each 60th step is listed. The default value of IPRINT is 1. Orbital elements are printed to a separate file PROJECT.ORB if IORBEL=1.

The parameters OEYEAR, OEMON, OEDAY, OEHRS, OEMIN, OESEC define the trajectory

Table 2.1. `NAMELIST` parameters for `SAPRE`

Parameter	Data Type	Default	Function
`TITLE`	`CHARACTER*80`	`80*' '`	Project title
`TYPE`	`CHARACTER*3`	GEN	Orbit type GEN: General GEO: Geostationary HEL: Heliocentric GTO: Geostationary Transfer (Ariane)
`OEYEAR`	`INTEGER`	1995	Orbit epoch year
`OEMON`	`INTEGER`	1	Orbit epoch month
`OEDAY`	`INTEGER`	1	Orbit epoch day
`OEHRS`	`INTEGER`	0	Orbit epoch hour
`OEMIN`	`INTEGER`	0	Orbit epoch minute
`OESEC`	`REAL*8`	0	Orbit epoch second
`SEYEAR`	`INTEGER`	-1	Orbit start year
`SEMON`	`INTEGER`	-1	Orbit start month
`SEDAY`	`INTEGER`	-1	Orbit start day
`SEHRS`	`INTEGER`	-1	Orbit start hour
`SEMIN`	`INTEGER`	-1	Orbit start minute
`SESEC`	`REAL*8`	-1	Orbit start second
`AMJD`	`REAL*8`	-1	Orbit epoch in Modified Julian Day from 1950.0. Used instead of `OEYEAR`... when ≥ 0.
`ORBITS`	`REAL*8`	1	Number of orbits, used if `EPDUR`=0
`EPDUR`	`REAL*8`	0	Duration of trajectory (days). If 0, `ORBITS` is used.
`IAE`	`INTEGER`	0	Input control: 0: Perigee and apogee 1: Semi-major axis, eccentricity 2: Circular orbit with altitude `ALT`
`HAPO`	`REAL*8`	36000	Apogee height (km)
`HPER`	`REAL*8`	300	Perigee height (km)
`A`	`REAL*8`	6671	Semi-major axis (km)
`E`	`REAL*8`	0	Eccentricity
`ALT`	`REAL*8`	300	Altitude (km) for circular orbits

Table 2.1. (continued)

Parameter	Data Type	Default	Function
RINCL	REAL*8	0	Orbit inclination (deg)
RAAN	REAL*8	0	Right ascension (deg) of ascending node
IG50	INTEGER	1	Choice of origin for RAAN (for GEN and HEL orbits only): 0: Greenwich 1: γ_{50}
ARGPER	REAL*8	0	Argument of perigee (deg)
TRANO	REAL*8	0	True anomaly (deg)
LONGAPO	REAL*8	999.9	Apogee longitude (deg)
LONGPER	REAL*8	999.9	Perigee longitude (deg)
OELONG	REAL*8	0	East longitude (deg) for geostationary orbit
SOLTIM	REAL*8	-9999.999	Local solar time (hrs) for heliosynchronous orbit
STEP	REAL*8	2	Eccentric anomaly integration step (deg)
DT1	REAL*8	120	Time step for output (s). Used if altitude $<$ DH2.
DT2	REAL*8	240	Time step for output (s). Used if DH2 $<$ altitude $<$ DH3.
DT3	REAL*8	1200	Time step for output (s). Used if altitude $>$ DH3.
DH2	REAL*8	1.0D4	Use DT1 if height $<$ DH2.
DH3	REAL*8	1.0D6	Use DT2 if height $<$ DH3.
WIBAIR	REAL*8	0	Air drag parameter. Used if altitude $<$ 700 km.
WIBSPR	REAL*8	0	Solar radiation pressure
NCIRA	INTEGER	5	CIRA atmosphere density model number (1–10). Used if WIBAIR$\neq$ 0.
SUN	INTEGER	1	Flag for gravitational perturbations from Sun
MOON	INTEGER	1	Flag for gravitational perturbations from Moon
KZONAL	INTEGER	6	Number of zonal harmonic gravity coefficients (0–8). Ignore oblateness of Earth if KZONAL and KTESS $\leq$ 1.
KTESS	INTEGER	3	Number of tesseral harmonic gravity coefficients (0–4). Ignore oblateness of Earth if KZONAL and KTESS $\leq$ 1.

Table 2.1. (continued)

Parameter	Data Type	Default	Function
XB	REAL*8	15.25	Bow shock distance (R_E)
XM	REAL*8	10	Magnetopause distance (R_E)
PB	REAL*8	22.164	Bow shock curvature (R_E)
PM	REAL*8	15	Magnetopause curvature (R_E)
PA	REAL*8	90	Pitch angle for interface file
IPRINT	INTEGER	1	Output table control: 0: Lists each ephemeris step 1: Lists each 60th step
IORBEL	INTEGER	0	0: Orbital elements are not printed 1: Orbital elements are printed
IBUG	INTEGER	0	1: Extra output in report file
RUN	INTEGER	1	Run number (for output annotation)
ORBNUM	REAL*8	1	Orbit identification number (for output annotation)
IATTI	INTEGER	0	Control of attitude interface file: 0: No attitude interface file 1: z axis to zenith, x axis to North, y axis to West 2: z axis parallel to velocity vector, y axis in orbital plane pointing Southwards 3: x, y, z parallel to GEI system

epoch, i.e. the epoch of the first perigee. Alternatively, the trajectory epoch can be given in Modified Julian Day from 1950.0 with the parameter AMJD. AMJD is used when it is positive or zero. The start time of the orbit can be chosen by setting SEYEAR, SEMON, SEDAY, SEHRS, SEMIN, SESEC to positive values.

The orbit can be specified in three ways, depending on the value of the parameter IAE:

1. IAE=0: HPER and HAPO specify the altitude (in km) of the perigee and apogee of the trajectory, respectively, above the mean radius of the Earth (6378 km);

2. IAE=1: A specifies the length of the semi-major axis (in km) and E the eccentricity;

3. IAE=2: a circular orbit with altitude specified by ALT (in km).

The default value of IAE is 0. In addition, the parameter TYPE selects among a number of pre-defined orbit types:

1. `TYPE=GEN`: general orbit type, to be defined in the `NAMELIST`;

2. `TYPE=GEO`: geostationary orbit with longitude `OELONG`;

3. `TYPE=HEL`: heliosynchronous orbit with right ascension of ascending node given by `RAAN` or calculated from local solar time `SOLTIM`, the inclination is calculated by `SAPRE`;

4. `TYPE=GTO`: geostationary transfer orbit for Ariane.

The air drag parameter `WIBAIR` is defined as

$$\texttt{WIBAIR} = \frac{1}{2}\, C_\mathrm{D}\, \frac{A}{M}\, 10^6 \,,$$

where C_D is the coefficient of drag, A is the cross-sectional area in square meters, and M is the mass in kg. This parameter is ignored for orbit altitudes greater than 700 km. The default value is 0.

The solar radiation pressure parameter `WIBSPR` is defined as

$$\texttt{WIBSPR} = 0.451 \times 10^{-8} K\, \frac{A}{M} \,,$$

where

- K is the material parameter:

    $= 1 - \tau + \rho$ for a plate,
    $= 1 - \tau$ for a sphere,
    $= 2$ for a plate with perfect specular reflection,
    $= 1$ for a sphere with perfect specular reflection of perfect absorption,
    $= 1.44$ for a sphere with perfect diffuse reflection;

- A is the total reflective area in square meters;

- M is the mass in kg;

- τ and ρ are the transmissivity and reflectivity (constants).

The default value is 0.

`KZONAL` and `KTESS` represent the number of zonal and tesseral harmonic gravity coefficients, respectively. If `KZONAL` ≤ 1 and `KTESS` ≤ 1, the oblateness of the Earth is ignored. The default values are 6 and 3, respectively.

`NCIRA` is the number of the CIRA atmospheric density model, ranging between 1 and 10. The CIRA models are for:

   1: extremely low solar activity
   2: very low solar activity
   3: low solar activity
   4: between low and mean solar activity
   5: mean solar activity
   6: between mean and high solar activity
   7: lower level of high solar activity
   8: upper level of high solar activity
   9: very high solar activity
   10: extremely high solar activity

The default value is 5. The upper atmospheric density is available as a function of the local time for each two hour step of the day and as a function of the altitude in the range from 120 km to 700 km above the surface of the Earth. If the altitude is greater than 700 km, the density is set to 0. If the altitude is less than 120 km, the program will abort and an error message will be issued.

The bow shock and magnetopause paraboloid parameters are defined by the pairs (XB,PB) and (XM,PM), respectively, in units of R_{E}. The first elements are for the bow shock, the second ones for the magnetopause. The general form for the paraboloids is given by the equation

$$Y^2 + Z^2 = -2\mathtt{P}\left(X - \mathtt{X}\right).$$

Finally, a fixed pitch angle value can be specified for the interface file with the parameter PA.

2.2.3   Output from SAPRE

The output of SAPRE consists of four files: the common interface file PROJECT.INT, the report file PROJECT.SAP, the optional orbital element file PROJECT.ORB, and the optional attitude interface file PROJECT.ATT. SAPRE does not produce separate plot files.

Interface and report files

The common interface file contains the orbital trajectory data and serves as input for BLXTRA and TREP. The format of this file is described in Appendix B.

The report file PROJECT.SAP is a printable file containing the input data, the orbit parameters and summary tables. The world map coordinates, geocentric inertial ephemeris and Earth centred ecliptic coordinates are printed as well as the distance between the satellite and the bow shock paraboloid, magnetopause paraboloid, and northern neutral point. The summary tables include details of eclipses, crossings of the bow shock and magnetopause, and the nearest approach to the northern neutral point.

Table 2.2. Record structure of the attitude interface file. N indicates the number of points on the orbit. The whole structure is repeated for additional orbits.

Record	Format	Description
		Header Records
1	1X,A80	Project descriptor
2	1X,5I6,F8.4, 2E14.6,F10.5	Year, month, day, hour, minute, second of trajectory epoch, height (km) of apogee and perigee, orbit inclination (deg)
3	2(1X,A32),2I6	Description labels and model numbers for two internal magnetic field models
4	2(1X,A32),2I6	Description labels and model numbers for two external magnetic field models
5	1X,2F10.4,F9.6, 4E14.6,2(2X,A8), 2I3	Epochs for two internal magnetic field models, geomagnetic moment, four energy thresholds (two each for protons and electrons), trapped particle model headers, solar flare model number, attitude model number (IATTI)
6	1X,A80	Format string for data records
7		Blank line
		Data Records
1–N	given in header record 6	Rotation matrix describing the satellite attitude relative to the geographic spherical coordinate system $(\bar{1}_R, \bar{1}_\theta, \bar{1}_\phi)$ at the current location of the satellite, and the angle (deg) between the zenith direction and the velocity vector
$N+1$	given in header record 6	10*-1.0

Attitude interface file

The attitude interface file PROJECT.ATT is an input for ANISO which contains the attitude of the satellite along its orbit. For the three cases listed below, this file can be produced by SAPRE. For other specific cases, the file PROJECT.ATT must be provided by the user.

The attitude interface file PROJECT.ATT is an ASCII file with fixed record length of 112 bytes, in direct access. Each record corresponds to an orbital point in the common interface file PROJECT.INT. The format of the PROJECT.ATT is given in Table 2.2. The rotation matrix between the coordinate system $(\bar{1}_x, \bar{1}_y, \bar{1}_z)$ attached to the satellite and the geocentric

spherical coordinate system $(\bar{1}_R, \bar{1}_\theta, \bar{1}_\phi)$ is given by

$$
[A(i,j)] = \left[\begin{array}{ccc}
\bar{1}_R \cdot \bar{1}_x & \bar{1}_R \cdot \bar{1}_y & \bar{1}_R \cdot \bar{1}_z \\
\bar{1}_\theta \cdot \bar{1}_x & \bar{1}_\theta \cdot \bar{1}_y & \bar{1}_\theta \cdot \bar{1}_z \\
\bar{1}_\phi \cdot \bar{1}_x & \bar{1}_\phi \cdot \bar{1}_y & \bar{1}_\phi \cdot \bar{1}_z
\end{array} \right] .
\tag{2.1}
$$

The unit vector $\bar{1}_R$ points to the zenith. The unit vectors $\bar{1}_\theta$ and $\bar{1}_\phi$ lie in the horizontal plane, pointing respectively to the geographic South and East directions.

SAPRE can produce three types of attitude interface files, under the control of the NAMEL-IST parameter IATTI. The default value is set to 0 for which no attitude interface file is produced. When IATTI is set to a positive value, a PROJECT.ATT file is generated by SAPRE. The file contains the attitude of the satellite which corresponds to one of the three following stabilized orientations:

1. IATTI=1: The z-axis of the satellite points to the zenith. The x-axis and y-axis lie in the horizontal plane and point to the geographic North and West, respectively.

2. IATTI=2: The z-axis is parallel to the velocity vector of the satellite. The x-axis lies in the orbital plane pointing away from the Earth. The y-axis is perpendicular to the orbital plane and points to the South direction.

3. IATTI=3: The $(\bar{1}_x, \bar{1}_y, \bar{1}_z)$ coordinate system is parallel to the GEI coordinate system.

When different SAPRE NAMELISTs are present in the same NAMELIST file PROJECT.NML, IATTI has to be either equal to zero in every NAMELIST, or always greater than zero. In other words, trajectories with and without attitude interface file generation should not be mixed.

2.2.4   Using the Runge-Kutta orbit generator

Appropriate values are needed for the eccentric anomaly step ΔE (STEP) and for the time increment Δt for the ephemeris output. To achieve a stable integration process the figures shown in Table 2.3 should be used as guides.

Table 2.3. Suggested values for the Runge-Kutta generator

Orbit eccentricity	STEP (decimal degrees)
$e < 0.5$	5
$e \geq 0.5$	2

A value for Δt must be chosen so that the Van Allen belts are sampled securely, primarily at perigee. This is achieved by determining the minimum time step corresponding to a ΔE step with the equation

$$\Delta t_{\min} = T\,(1-e)\,\frac{\Delta E}{360°}\,,$$

where T is the orbital period in minutes.

As an example, suppose that $T = 2\,\mathrm{days}$, $e = 0.8$, and $E = 1°$:

$$\Delta t_{\min} = 2 \times 1440\,(1-0.8)\,\frac{1°}{360°} = 1.6\,.$$

In this case, since `SAPRE` requires that the values must be entered as multiples of $60\,\mathrm{s}$, specify `DT1 = DT2 = DT3 = 120`. The values for `DT2` and `DT3` can be increased in combination with the altitudes `DH1` and `DH2`, which determine the maximum altitudes (in km) for which `DT1` and `DT2`, respectively, will be used (for altitudes greater than `DH2`, `DT3` is used).

2.3   BLXTRA

2.3.1   Function and structure

`BLXTRA` converts geographical positions given in geodetic longitude, latitude, and height into the (B, L) coordinate system (McIlwain 1961) using an internal geomagnetic field representation and optionally an external magnetic field. The field line tracing in `BLXTRA` uses an algorithm developed by Pfitzer (1991), which was extended and refined by BISA.

`BLXTRA` First reads and updates the header information in the common interface file `PROJECT.INT`. For each orbital point, the modified Julian day is read, which is then used to derive universal time. The geodetic coordinates are transformed into geocentric coordinates prior to the calculation [except when the geomagnetic field model of Jensen & Cain (1962) is used]. If an external field is specified, the necessary coordinate transformations are executed. `BLXTRA` then calculates (B, L) using the pitch angle read from the interface file, and adds the (B, L) pair to the common interface file. When `TREP` is run on the same project, the (B, L) pairs will be overwritten by the values calculated by `TREP`. It is not necessary to run `BLXTRA` before `TREP`. However, one can run `BLXTRA` after `TREP` without destroying the fluxes written on the interface file.

2.3.2   Pitch angle dependence

The (B, L) coordinates are associated with a particle moving through a point P with given geographical coordinates, and having a given pitch angle α. In the special case $\alpha = 90°$, P is a mirror point for the particle considered (the conjugate mirror points will be denoted by M and M').

The standard software routines used for calculating (B, L) coordinates for a point in space implicitly make the assumption that this point is the mirror point of a particle on the geomagnetic field line passing through it. This approach was adopted for use with older trapped particle models, since these generally are given in terms of omnidirectional fluxes. More modern measurements, however, include flux measurements with high directional resolution. Therefore, in order to make use of this additional information, pitch angle dependence was implemented in the calculation of B and L.

The mirror point M of a particle with pitch angle $\alpha \neq 90°$, measured at P, does not coincide with P. The magnetic field intensity B_{m} at M is given by

$$B_{\mathrm{m}} = \frac{B}{\sin^2\alpha}, \tag{2.2}$$

where B is the value of the magnetic field intensity at P. The field line tracing procedure used to calculate the second adiabatic invariant I has to be extended beyond P to the mirror point M (and to the conjugate mirror point M').

The pitch angle dependent coordinates $(B_{\mathrm{m}}, L_{\mathrm{m}})$ have been implemented in the latest version of BLXTRA (from here on we drop the subscript "m"). For each orbital point, one pitch angle value is given in the common interface file. When the common interface file is created by SAPRE, the value $\alpha =$ PA is written for all orbital points. If other values are needed, a separate program has to be used to alter the values of α. Such a program is not provided in UNIRAD.

2.3.3 Input and output

BLXTRA takes as input the NAMELIST BLXTRA, and reads geodetic coordinates, modified Julian day, and pitch angle from the common interface file PROJECT.INT. The header of the common interface file is updated. The calculated (B, L) values are appended to the data records in the common interface file. The B value written by BLXTRA is the magnetic field intensity at the point specified by the geodetic coordinates, not that at the mirror point. The NAMELIST parameters for BLXTRA are listed in Table 2.4 with their data type and default value, and a brief description.

When the field line tracing in the calculation of the second adiabatic invariant breaks down, a negative L value is returned. These negative values serve as flags, indicating the problem that occurred. The values of the error flags are listed in Table 2.5.

BLXTRA produces a report file PROJECT.BLP. In addition, when IASCII=1, an ASCII file PROJECT.BLA is created. This latter file contains the information written to the common interface file plus the vector components of the magnetic field vector in geocentric coordinates. The format of the file PROJECT.BLA is given in Table 2.6. BLXTRA does not produce separate plot files.

Table 2.4. NAMELIST parameters for BLXTRA

Parameter	Data Type	Default	Function
BLTIME	REAL*8	1995	Epoch for internal magnetic field model
MODEL	INTEGER	0	Choice of internal magnetic field model: 0: DGRF/IGRF for epoch BLTIME 1: Jensen & Cain (1962) for epoch 1960 2: GSFC 12/66 (Cain et al. 1967) for epoch BLTIME
OUTER	INTEGER	0	Choice of external magnetic field model: 0: No external field 1: Mead & Fairfield (1975) 2: Tsyganenko (1987) short 3: Tsyganenko (1987) long 4: Tsyganenko (1989) 5: Olson & Pfitzer (1977) quiet 6: Olson & Pfitzer (1988) dynamic
VALUE_KP	REAL*8	0	Geomagnetic activity index K_{p}
DEN	REAL*8	25	Density ρ of solar wind
VEL	REAL*8	300	Velocity V of solar wind
DST	REAL*8	-30	Geomagnetic activity index D_{st}
IPRINT	INTEGER	60	Frequency of points in listing
IASCII	INTEGER	0	ASCII file with geographic and (B, L) coordinates and components of **B** is created when not 0
CUTOFF	REAL*8	10^6	Maximum orbit time (hours)

2.3.4   Magnetic field models in BLXTRA

By default BLXTRA uses the DGRF/IGRF internal magnetic field models and no external model. The inner and outer models available are described below.

Internal magnetic field models

The user has a choice of three internal geomagnetic field models:

1. DGRF/IGRF (1945–1995),

2. Jensen & Cain (1962),

Table 2.5. Possible negative L values indicating a problem encountered during field line tracing

Value	Error
-1	$\|$Geomagnetic latitude$\| > 80°$
-2	Field line crosses the magnetopause
-3	Tracing is beyond valid field limits ($B < 0.00004$)
-4	Maximum step number has been reached
-6	Less than four points in field line tracing
-7	The magnetic field has a local maximum
-8	An error occurred in the interpolation routine

Table 2.6. Format of the ASCII file `PROJECT.BLA` produced by `BLXTRA`. N signifies the number of orbital points. If more than one trajectory were specified, the whole structure is repeated.

Record	Format	Description
1	`1X,A80`	Project header
2	`2(1X,A32)`	Header of internal and external magnetic field model
3	`2I3,F7.1`	`MODEL, OUTER, BLTIME`
4		Table headers
5–$(N+4)$	`F12.5,F8.2,F7.2,` `F10.2,F6.1,4F10.6,` `F8.4`	Modified Julian Day, geographic coordinates, pitch angle, geocentric components of $\boldsymbol{B}$, (B, L)
$N+5$	`F12.5,F8.2,F7.2,` `F10.2,F6.1,4F10.6,` `F8.4`	`10*-1.0`: sentinel record to signify end of trajectory

   3. GSFC12/66 (Cain et al. 1967).

The internal models are selected with the parameter `MODEL`. The expansion coefficients for the internal geomagnetic field models are linked in with `BLXTRA`.

The coefficients of the Jensen & Cain model do not depend on time and have epoch 1960. The other models are time dependent and are extrapolated to epoch `BLTIME`. When the DGRF/IGRF models are chosen, the program selects the model whose epoch is closest to `BLTIME`. The Jensen & Cain and GSFC 12/66 models have been included since they were used to build AP-8 and AE-8 (Heynderickx et al. 1996).

External magnetic field models

Six external magnetic field models are available:

1. Mead & Fairfield (1975),

2. Tsyganenko (1987) short,

3. Tsyganenko (1987) long,

4. Tsyganenko (1989),

5. Olson & Pfitzer (1977) quiet,

6. Olson & Pfitzer (1988) dynamic.

The models are selected with the parameter OUTER (see Table 2.4).

The coordinate transformations necessary for the use of the external models are performed by BLXTRA. The dipole tilt angle also is calculated internally when necessary and thus is not an input parameter.

The Mead & Fairfield (1975) model is included for historical reasons. There are no input parameters for this model.

The Tsyganenko models   The Tsyganenko models are sets of models each representing a different K_{p} level. The models were created as fits to data from eight IMP and two HEOS satellites covering the period 1966–1980, with the data binned according to activity. The Tsyganenko (1989) model is tilt dependent and was developed primarily as a tail model.

The Tsyganenko models are claimed to be valid up to $X = -70\,R_{\mathrm{E}}$ but may differ nevertheless from the observed field in this region for several reasons (Stern & Tsyganenko 1992):

- The models only give average values of the field whereas actual values vary greatly around such averages. This is true for the other external models as well.

- The high latitude coverage of the data used to build the models is sparse.

- The models use only one magnetospheric index, the planetary geomagnetic activity index K_{p}.

The Olson-Pfitzer models   The tilt dependent model of Olson & Pfitzer (1977) is an average models fit to quiet conditions ($K_{\mathrm{p}} = 0, 1$) using data from OGO 3 and 5 (inner magnetosphere) and Explorer 33 and 35 (tail). This model is valid from $2.5\,R_{\mathrm{E}}$ to $15.0\,R_{\mathrm{E}}$. There are no input parameters for this model.

The Olson & Pfitzer (1988) dynamic model is a scalable model depending upon the activity level. It uses the current systems in the tilt dependent Olson & Pfitzer (1977) model,

but scale factors are introduced to separately adjust the strength of all current systems and the geometry of the magnetopause and tail currents. The input parameters that determine the scale factors are the standoff distance of the magnetopause and the activity index D_{st}. The spatial validity in the tail has been extended out to $60\,R_E$ for this model.

The Olson & Pfitzer (1988) dynamic model does not have a tilt dependence. The input parameters are:

1. DEN: density (cm^{-3}) of the solar wind (default: 25),

2. VEL: velocity ($\mathrm{km\,s}^{-1}$) of the solar wind (default: 300),

3. DST: geomagnetic activity index (nT) D_{st} (default: -30.0).

2.4 TREP

2.4.1 Function and structure

TREP calculates orbital radiation environment fluxes for arbitrary spacecraft trajectories. It estimates fluxes of trapped particles in function of (B, L) coordinates, and computes solar proton event probabilities to estimate solar proton fluences. The output consists of:

1. the time dependent fluxes of trapped protons, solar protons, and trapped electrons;

2. trajectory-averaged spectra for trapped protons and electrons;

3. solar flare fluence spectra, taking into account geomagnetic shielding.

2.4.2 Trapped particle models

TREP accesses the NASA trapped particle models AP-8 (Sawyer & Vette 1976) and AE-8 (Vette 1991a) at each orbital point contained in the common interface file. The NASA trapped radiation models AP-8 and AE-8 are distributed as tables of omnidirectional fluxes in function of particle energy E and $(B/B_0, L)$, where

$$B_0 = \frac{0.311653}{L^3}\,. \tag{2.3}$$

The NASA models AP-8 and AE-8 are static models built with data obtained in the sixties and seventies. There are two sets of models: AP-8 MAX and AE-8 MAX for solar maximum conditions, and AP-8 MIN and AE-8 MIN for solar minimum conditions. Since these models were built with specific magnetic field models, the same magnetic field models should be used to calculate the $(B/B_0, L)$ values that serve as input to AP-8 and AE-8. The appropriate magnetic field models are: Jensen & Cain (1962) for AE-8 MIN, AP-8 MIN, and AE-8 MAX,

and GSFC 12/66 (Cain et al. 1967) updated to epoch 1970 for AP-8 MAX (Heynderickx et al. 1996). Two coordinate pairs (B, L) are written to the common interface file `PROJECT.INT`, corresponding to the field models used with the proton and electron model, respectively.

The position of the SAA in the old magnetic field models to be used with AP-8 and AE-8 is different from its actual position due to the secular variation on the geomagnetic field (Fraser-Smith 1987). This may have important effects on the prediction of trapped particle fluxes for low altitude orbits. Therefore, a correction has been built into `TREP`: when the `NAMELIST` parameter `SAAROT` is set to 1 (the default value), a correction term $\Delta\phi(t)$ is added to the geodetic longitude before the calculation of (B, L) (the actual value of the longitude on the interface file is not changed):

$$\delta\phi(t) = 0.3° \, (t - \texttt{EPOCH}) \,. \tag{2.4}$$

where t is the orbit epoch as written in the header of the common interface, and `EPOCH` is the epoch of the geomagnetic field model.

Originally, `TREP` read (B, L) values, calculated by `BLXTRA`, from the common interface file. `TREP` has since been modified so that it calculates (B, L) independently, with the appropriate magnetic field models. Consequently, one can pass directly from `SAPRE` to `TREP`. (B, L) are calculated (twice, if necessary, for the two geomagnetic field models) and written on the interface file. Therefore, it is not necessary to run `BLXTRA` prior to `TREP`.

The energy spectrum for trapped protons, trapped electrons, and solar flare protons is specified with the `NAMELIST` parameters `PROEN(30)`, `ELEEN(30)`, and `ENERFL(30)`, respectively. The values are threshold energies (in MeV) for integral flux. The maximum number of energies is 30 for each of the three types of particle, the minimum is 10. The number of energies is specified with the `NAMELIST` parameters `NENERP`, `NENERE`, and `NENERS`, which default to 30.

The AP-8 and AE-8 models are distributed with a set of routines called `TRARA` which perform interpolations of the flux maps in $(E, B/B_0, L)$ space. In regions of strong gradients of B/B_0 and L the `TRARA` interpolation routines do not perform very well, resulting in a possible underestimate of the trapped proton flux in the SAA by as much as 50%. Daly & Evans (1993) have developed an interpolation routine which is better suited for the regions of steep flux gradients and which does not display the anomalies obtained with the `TRARA` routines. This routine has been implemented in `TREP` and replaces the `TRARA` routines.

Trapped protons

In addition to the omnidirectional models AP-8 there also exists a version of the AP-8 models in terms of perpendicular fluxes (Vette, unpublished report), which has been added to `TREP`. See the section on pitch angle dependence for instructions on how to use this model. Please note that `SHIELDOSE`, `EQFRUX`, and `EQFRUXGA` expect omnidirectional fluences as input, which means that the unidirectional AP-8 model should not be used for computations of doses or damage equivalent electron fluences.

Table 2.7. Confidence levels corresponding to `ISIG`

`ISIG`	Confidence level (%)
-3	0.135
-2	2.275
-1	15.865
0	50.000
1	84.135
2	97.725
3	99.865

Trapped electrons

The influence of local time on electron fluxes can be included. The model used is an extension of the AE-4 local time model of Singley & Vette (1972a). It is called by setting `ILTV=1` (the default value is 0). Electron local time variation parameters then are initialized and the local time correction to the logarithm of the flux is computed.

A statistical model describing the distribution of observed electron flux levels has been introduced. Electron fluxes are modified according to an extension of the AE-4 statistical model of Singley & Vette (1972a). The model is activated by setting `ISIG` to one of the values -3, -2, -1, 1, 2, 3 (the default value is 0), which results in the calculation of standard deviations from the mean flux. `ISIG` gives the confidence level that $\log_{10}(\text{flux})$ will not be exceeded. The confidence levels are listed in Table 2.7.

Instead of the electron model AE-8, a model developed for the geostationary environment on the basis of LANL omnidirectional measurements (Lemaire et al. 1991) may be used. The model is activated by setting `TREMOD=2` (`TREMOD=1`, the default value, selects AE-8). This model is valid only for geostationary orbits and gives zero fluxes everywhere else.

Pitch angle dependence

Since the dependence on pitch angle has been introduced in `TREP`, the magnetic coordinates (B, L) written to the interface file are the magnetic field intensity at the geodetic coordinates and the L value obtained by tracing the field line to the mirror point.

The user can choose between the calculation of omnidirectional or unidirectional proton fluxes by means of the `NAMELIST` parameter `OMNI`. With `OMNI=1` (the default), omnidirectional proton fluxes are calculated, while setting `OMNI=0` results in unidirectional proton fluxes.

For the calculation of unidirectional proton fluxes, B_m is derived with Eq. (2.2), since the flux in direction α at the point with coordinates (B, L) is equal to the perpendicular flux at

the point with coordinates (B_{m}, L). The coordinates $(B_{\mathrm{m}}/B_0, L)$ then are used as input to the unidirectional proton model.

The user should be aware that if $\alpha \neq 90°$, the omnidirectional flux is calculated for the mirror point of the particle, which in that case does not coincide with the geographic coordinates!

2.4.3 Solar flare protons

Solar flare particle events, because of their unpredictability and large variability in magnitude, duration and spectral characteristics, have to be treated statistically. In TREP solar flare proton spectra are generated with the models of King (1974) or Feynman & Gabriel (1990), which are described below, by means of the NAMELIST parameter FLAMOD (default value is 0, meaning no solar flare model). For both models, geomagnetic shielding has to be evaluated.

The mission duration (in years) is specified with the NAMELIST parameter TFLARE (default is one year). TREP computes the mission time during solar active years from TFLARE and the orbit epoch year and month OEYEAR and OEMON which are written in the header of the common interface file PROJECT.INT. The starting offset (in years) in the solar active cycle can also be given explicitly with the NAMELIST parameter FLSTAR. When FLSTAR≥ 0 this value is used to calculate the time during solar active years instead of the orbit epoch.

The King model

King (1974) used Burrell's modified Poisson distribution for predicting the number of events expected and the spectra. Burrell's distribution gives the probability p of a number of events n occurring during a time t, based on a previously observed frequency N during time T:

$$p(n, t; N, T) = \frac{(n + N)! \, (t/T)^n}{n! \, N! \, (1 + t/T)^{N+n+1}} \, .$$
(2.5)

Most planning is based on worst case confidence levels (95%) and a model of the August 1972 anomalously large (AL) event. This anomalously large flare dominates all others in terms of peak flux, integrated flux, and duration.

King (1974) examined IMP energetic proton data for a number of flare events and, treating the 1972 AL flare separately, produced a log normal distribution of ordinary flare fluences as a function of proton energy and an exponential fit to the 1972 AL fluence spectrum:

$$J(> E) = 7.9^9 \, \exp\left(\frac{30 - E}{26.5}\right) \, ,$$
(2.6)

with E in MeV and J in cm^{-2}.

Models of mean, worst case and anomalous solar flare proton spectra based on King's work are used in TREP, together with the Burrell distribution. Depending on the mission duration and timing, a number of flares is predicted and the resulting total fluence spectrum is produced.

The number of anomalously large solar flares and of ordinary solar flares can be input with the NAMELIST parameters NAL and NOR, respectively. Note that if NOR is input as, or predicted to be, 1, and the probability level is 90% or greater, a worst case ordinary flare spectrum is produced.

The Feynman & Gabriel model

The predictive model of Feynman & Gabriel (1990) for the $> 10\,\mathrm{MeV}$ and $> 30\,\mathrm{MeV}$ solar proton environment is based on observations made from 1956 through 1985. In this data set the distinction between "ordinary events" and "anomalously large events" disappeared, which permitted the use of of the statistical analysis methods that have been developed for "ordinary events" to be used on the entire data set. The $> 10\,\mathrm{MeV}$ fluences expected with the new model are about twice those expected on the basis of earlier models. At energies $> 30\,\mathrm{MeV}$ the old and new models agree.

The results obtained with the Feynman & Gabriel (1990) model do not depend critically on the fluence from any one event and are independent of the size of the sunspot cycle.

With the NAMELIST parameter IFORM, the user can choose between a solar proton spectral form that is exponential in energy (IFORM=0, the default) or exponential in rigidity (IFORM=1).

Geomagnetic shielding

Geomagnetic shielding of flare protons is computed on the basis of the trajectory in (B, L) space. Stassinopoulos and King (1973) reported a model which has total cutoff at $L = 5$: it is assumed that no protons can penetrate to lower values. Computing the geomagnetic cutoff for vertically arriving protons, one finds that this model corresponds to a quiet magnetosphere vertical cutoff model excluding protons with energy below $200\,\mathrm{MeV}$ from $L < 5\,R_\mathrm{E}$. This model is adequate for most cases and is included in TREP and is selected by setting the NAMELIST parameter IFLATT=0 (this is the default value).

Protons of lower energy can penetrate below $L = 5$ with other arrival directions, especially in a disturbed magnetosphere when the geomagnetic shielding is weakened. For westward arrival at the $L = 5$ geomagnetic equator in a disturbed magnetosphere, the energy cutoff could be as low as $30\,\mathrm{MeV}$. A model for flare proton cutoff that depends on the arrival direction is available in TREP. It is selected by setting the NAMELIST parameter IFLATT=1.

For missions that do not remain at a heliocentric radial distance r of 1 AU a correction should be applied to the solar flare fluence: the fluence should be multiplied by a factor which depends on the actual trajectory, i.e. the integration of the radial dependence of the fluence over the mission trajectory. The radial dependence used in TREP is an inverse r

cubed dependence for $r \leq 1\,\mathrm{AU}$ and an inverse r squared dependence for $r > 1\,\mathrm{AU}$. These radial dependencies are worst case choices. r can be input with the `NAMELIST` parameter `RAU`, whose default value is 1.

2.4.4   Input and output

Input files

`TREP` takes as input the `NAMELIST TREP`, and reads modified Julian day, geodetic coordinates, and pitch angle from the common interface file. The `NAMELIST` parameters for `TREP` are listed in Table 2.8 with their data type and default value, and a brief description.

Output files

The header in the common interface file is updated. The (B, L) coordinates and the integral proton and electron fluxes for the energy thresholds specified in the `NAMELIST` parameters `ETHP` and `ETHE` are appended to the data records in the common interface file.

TREP produces the report file `PROJECT.TRP`, but no separate plot files. An additional output file `PROJECT.TRI` is created when the `NAMELIST` parameter `INTERF=1` (this is the default). This file contains the orbit-averaged spectra of trapped protons and electrons, and of solar protons if a solar flare model was specified. The file serves as input to `SHIELDOSE`, `EQFRUX`, and `EQFRUXGA`. The format of this file is described in Table 2.9. The orbit-averaged spectra can also be output in spreadsheet format (i.e. as comma separated columns with one header line) on the file `PROJECT.TRW`, which is created when `ISPSHT` is set to 1 (the default value is 0). The format of the file `PROJECT.TRW` is given in Table 2.10.

When `ISPEC=1`, `TREP` produces two ASCII files, `PROJECT.SPP` and `PROJECT.SPE` for protons and electrons, respectively. These files contain, for each orbital point, modified Julian day, geodetic coordinates, pitch angle, geocentric components of the magnetic field vector, (B, L), and the full particle flux spectrum. The format of the files `PROJECT.SP*` is given in Table 2.11. The file `PROJECT.SPP` is needed with `ANISO`.

2.5   TREPPOS

`TREPPOS` works interactively and allows calculation of trapped particle fluxes at particular positions in the Earth's radiation belts. The user has to supply a project name, a title, the proton and electron energies (which default to the values in Table 2.8) and a series of (B, L) or $(B/B_0, L)$ coordinate pairs.

TREPPOS calculates omnidirectional trapped proton and electron fluxes with the NASA models AP-8 and AE-8. The local time variation model and the statistical model (both described in Sect. 2.4.2) can be applied to the electron fluxes.

Table 2.8. `NAMELIST` parameters for `TREP`

Parameter	Data Type	Default	Function
SOLACT	CHARACTER*3	MAX	Solar activity level for NASA trapped radiation models (MAX, MIN)
SAAROT	INTEGER	1	0: No correction for drift of SAA 1: Westward rotation of SAA of $0.3°$/yr
OMNI	INTEGER	1	0: Unidirectional AP-8 proton fluxes 1: Omnidirectional AP-8 proton fluxes
TREMOD	INTEGER	1	1: Electron model AE-8 2: Electron model LANL
NENERP	INTEGER	30	Number of energies in trapped proton spectrum ($10 \leq$NENERP≤ 30).
PROEN	REAL*4(30)		Trapped proton threshold energies (MeV). Default: 0.1, 0.5, 1, 2, 3, 4, 5, 6, 8, 10, 12, 15, 17, 20, 25, 30, 35, 40, 45, 50, 60, 70, 80, 90, 100, 125, 150, 175, 200, 300
NENERE	INTEGER	30	Number of energies in trapped electron spectrum ($10 \leq$NENERE≤ 30).
ELEEN	REAL*4(30)		Trapped electron threshold energies (MeV). Default: 0.04, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 1.0, 1.25, 1.5, 1.75, 2.0, 2.25, 2.5, 2.75, 3.0, 3.25, 3.5, 3.75, 4.0, 4.25, 4.5, 4.75, 5.0, 5.5, 6.0, 6.5, 7.0
NENERS	INTEGER	30	Number of energies in solar flare proton spectrum ($10 \leq$NENERS≤ 30).
ENERFL	REAL*4(30)		Threshold energies (MeV) for solar flare protons. Default: 0.1, 0.5, 1, 2, 3, 4, 5, 6, 8, 10, 12, 15, 17, 20, 25, 30, 35, 40, 45, 50, 60, 70, 80, 90, 100, 120, 140, 160, 180, 200
ETHP	REAL*4(2)	1, 10	Trapped proton energy thresholds (MeV) for prints, plots, and interface file
ETHE	REAL*4(2)	0.1, 1	Trapped electron energy thresholds (MeV) for prints, plots, and interface file
FLUXTH	REAL*4(2)	1, 1	Threshold fluxes ($\mathrm{cm}^{-2}\mathrm{s}^{-1}$) for protons and electrons for flux-free time tables
ISIG	INTEGER	0	Extension of AE-4 statistical model. Values: $-3, -2, -1, 0, 1, 2, 3$

Table 2.8. (continued)

Parameter	Data Type	Default	Function
ILTV	INTEGER	0	Local time variation (LTV) flag for AE-8 electron fluxes: 0: no LTV of electron fluxes 1: include LTV of electron fluxes
FLAMOD	INTEGER	0	Select solar flare model. 0: No solar protons 1: Feynman & Gabriel's (1990) flare model 2: King's (1974) flare model
TFLARE	REAL*4	1	Mission duration (years)
FLSTAR	REAL*4	-1	Starting offset in solar activity cycle (years)
FLPROB	REAL*4	95	Probability (in %) of not exceeding solar proton fluence
RAU	REAL*4	1	Integrated heliocentric radial distance
IFORM	INTEGER	0	Solar proton spectral form for Feynman & Gabriel model 0: Exponential in energy 1: Exponential in rigidity
NAL	INTEGER	0	Number of anomalously large solar flares for King (1974) model. If 0, computed from Burrell distribution.
NOR	INTEGER	0	Number of ordinary solar flares for King model. If 0, computed from Burrell distribution.
IFLATT	INTEGER	0	Flare particle attenuation flag 0: Stassinopoulos-King model 1: Model of non-vertical proton arrival
IPRINT	INTEGER	60	Frequency of points in listing
INTERF	INTEGER	1	1: Produce interface file `PROJECT.TRI`. Not produced if 0.
ISPEC	INTEGER	0	1: Produce ASCII files `PROJECT.SPP` and `PROJECT.SPE`. Not produced if 0.
ISPSHT	INTEGER	0	1: Produce spreadsheet file `PROJECT.TRW`. Not produced if 0.
PERIOD	REAL*4	0	0: Orbital period (hrs). If 0, the period is computed from the apogee and perigee.

Table 2.9. Format of the interface file PROJECT.TRI. NENER* signifies the number of spectral energies. If more than one trajectory was specified, the whole structure is repeated.

Record	Format	Description
1	1X,A80	Project header
2	1X,A3, 2(1X,A8)	Solar activity level, trapped particle model headers
3	2(1X,A32)	Headers for internal magnetic field models
4	2(1X,A32)	Headers for external magnetic field models
5	4I3,2F8.1	Numbers of internal and external magnetic field models, BLTIME(2)
6	F6.1,2F8.1, E11.4	Orbit inclination (deg), perigee height (km), apogee height (km), period (hrs)
7	F7.3,F7.2, 2I4,2E11.4, I3	Mission duration (years), probability of not exceeding solar flare proton fluence (%), number of anomalously large flares, number of ordinary flares, fraction of orbit time in solar active period, total orbit time (hrs), number of solar flare model
8	3E11.4	Flux threshold FLUXTH(1) and energy thresholds ETHP for protons
9	2E11.4	Time (hrs) spent in regions where proton flux is above threshold FLUXTH(1) for energies ETHP
10	3E11.4	Flux threshold FLUXTH(2) and energy thresholds ETHE for electrons
11	2E11.4	Time (hrs) spent in regions where electron flux is above threshold FLUXTH(2) for energies ETHE
12	3I4	Number of energies in spectra of trapped protons, trapped electrons, solar flare protons
13 ⋮	3E11.4	Proton energy, integral and differential fluence
12+NENERP	3E11.4	Proton energy, integral and differential fluence
13+NENERP		Blank line
14+NENERP ⋮	3E11.4	Electron energy, integral and differential fluence
14+NENERP+ NENERE	3E11.4	Electron energy, integral and differential fluence

Table 2.9. (continued)

Record nr.	Format	Description
15+NENERP+ NENERE		Blank line
16+NENERP+ NENERE ⋮	3E11.4	Solar proton energy, integral and differential fluence
16+NENERP+ NENERE+NENERS	3E11.4	Solar proton energy, integral and differential fluence
17+NENERP+ NENERE+NENERS		Blank line

Table 2.10. Format of the spreadsheet file PROJECT.TRW. If more than one trajectory was specified, the whole structure is repeated.

Record	Format	Description
1	1X,A80	Project header
2	2(3(E11.4,',')), 2(E11.4,','), E11.4	Energy, integral flux and differential flux for trapped protons and electrons, and solar flare proton fluence
⋮		
31	2(3(E11.4,',')), 2(E11.4,','), E11.4	Energy, integral flux and differential flux for trapped protons and electrons, and solar flare protons

2.5.1   Inputs and outputs

TREPPOS takes its input interactively from the keyboard. Two output files are produced:

1. PROJECT.TRI: the creation of this file is an option in the user menu. It can serve as input for TREPAVE, SHIELDOSE, EQFRUX, and EQFRUXGA when a proper NAMEL-IST file is created.

2. The report file PROJECT.TRP.

If the project name PROJECT is not given on the command line, TREPPOS prompts for it.

Table 2.11. Format of the ASCII file `PROJECT.SP*` produced by `TREP`. N signifies the number of orbital points. If more than one trajectory was specified, the whole structure is repeated. `NENER` signifies the number of energies in the spectrum.

Record	Format	Description
1	`1X,A80`	Project header
2	`2(1X,A32)`	Headers of internal and external magnetic field models
3	`2I3,F7.1,I3`	`MODEL`, `OUTER`, `BLTIME`, number of energies
4	`1X,A80`	Data record format string
5		Table headers
6	see record 4	Modified Julian Day, geographic coordinates, pitch angle, geocentric components of $\boldsymbol{B}$, (B, L), `NENER` flux values
$\vdots$		
$N + 4$	see record 4	`(10+NENER)*-1.0`: sentinel record to signify end of trajectory

2.6 TREPAVE

`TREPAVE` takes the results from the `TREP` interface file `PROJECT.TRI` for each orbit and produces average proton and electron fluxes using appropriate weightings, contained in the `NAMELIST` parameter `WEIGHTS`, which is the sole parameter in the `NAMELIST TREPAVE`. `WEIGHTS` is a `REAL*4` vector of 200 elements, with default values 1.0. Solar proton fluences are not averaged but summed as `TREP` provides predictions of flares for a certain period.

2.6.1 Inputs and outputs

`TREPAVE` reads the `NAMELIST` file `PROJECT.NML` and the file `PROJECT.TRI` produced by `TREP`. The averaged spectra are written to the file `PROJECT.TRA`, which has the same format as `PROJECT.TRI` (see Table 2.9). The file `PROJECT.TRA` can be used as input for `SHIELDOSE`, `EQFRUX`, and `EQFRUXGA`, but should be renamed to a file with extension `.TRI`. An appropriate `NAMELIST` file should also be set up.

2.7   ANISO

At low altitude, the high-energy trapped proton fluxes are strongly anisotropic and may induce anisotropies in the radiation exposure of a spacecraft with stabilized attitude. The flux anisotropy is understood theoretically by the interaction of the trapped protons with the Earth's atmosphere and by the finite length of the proton gyroradius. The main purpose of modelling this flux anisotropy is to deduce angular-dependent proton flux spectra from standard omnidirectional flux databases which were, until recently, the only ones available. Such a model has been developed analytically by Watts et al. (1989) and has been applied to the evaluation of radiation shielding for manned spacecraft. A new model based on the Badhwar & Konradi (1990) pitch angle distribution has been developed by Kruglanski & Lemaire (1996).

ANISO transforms trapped proton omnidirectional integral fluxes produced by TREP into orbit-averaged unidirectional integral and differential fluences. The program can also provide the unidirectional integral and differential fluxes along the orbit of the satellite.

2.7.1   Anisotropy models

A trapped proton anisotropy model is intended to predict the trapped proton unidirectional flux at any point of observation as well as the unidirectional fluence obtained by the average of the flux along the trajectory of a satellite. The model is a combination of different ingredients:

1. a model of the omnidirectional flux j_0, such as the NASA trapped proton models AP-8 MIN and AP-8 MAX (Sawyer and Vette, 1976);

2. a pitch angle distribution caused by the atmospheric loss cone;

3. an East-West asymmetry effect due to the finite size of the gyration radius;

4. a normalisation factor which has to be determined so that the omnidirectional flux computed from the unidirectional flux remains equal to j_0.

Heckman & Nakano (1969) assumed that the flux along a magnetic field line is inversely proportional to the atmospheric density and proposed a pitch angle distribution based on a dipolar magnetic field and an exponential atmosphere. They obtained a pitch-angle distribution in the form of a gaussian function centred around the pitch angle $\alpha = 90°$ with the square of the standard deviation given by

$$\sigma^2 = \frac{3}{4}\frac{H}{R}\left(2 + \cos^2 I\right),\tag{2.7}$$

where I is the magnetic dip angle, H is the atmospheric scale height and R is the distance from the Earth's centre. With similar assumptions, Lenchek & Singer (1962) proposed an azimuthal correction function for trapped particles at low altitude. This function takes into

account that, for a given point of observation, a proton coming from the West has its guiding centre above the guiding centre of one coming from the East. The function of Lenchek & Singer (1962) depends on the ratio between the gyroradius of the trapped particles and the atmospheric scale height at their mirror point.

Watts et al. (1989) combined the Heckman & Nakano (1969) pitch angle distribution with the Lenchek & Singer (1962) East-West asymmetry function to derive a two-dimensional distribution. They obtained the following anisotropy conversion factor to transform omni-directional differential flux into unidirectional differential flux:

$$W_{\mathrm{VF}} = \frac{\exp\left[-\dfrac{(\alpha - \pi/2)^2}{2\sigma^2}\right] \, \exp\left(\dfrac{r_{\mathrm{g}} \cos I \sin \alpha \sin \phi}{H}\right)}{(2\pi)^{3/2} \sin \alpha \, \sigma \, \mathrm{erf}\left(\dfrac{\pi}{\sigma\sqrt{8}}\right) \, \mathrm{I}_0\left(\dfrac{r_{\mathrm{g}} \cos I \sin \alpha}{H}\right)} \, , \tag{2.8}$$

where α is the local pitch angle, ϕ is the azimuthal angle measured to the East in the plane perpendicular to the magnetic field vector from the local vertical plane, r_{g} is the gyroradius of a locally mirroring particle, and I_0 is the modified Bessel function.

Armstrong et al. (1990) used this anisotropy conversion factor with an atmospheric scale height H deduced from the Johnson & Smith (1985) atmospheric model. They applied it to the NASA models AP-8 MIN and AP-8 MAX to produce models of trapped proton unidirectional fluxes at low altitude. The models were called VF1MIN and VF1MAX, for solar minimum and solar maximum respectively, by Colborn et al. (1990).

Due to approximations, especially in the evaluation of the atmospheric scale height, Armstrong et al. (1990) restrict their model to the altitude range between 250 and 500 km. To extend the altitude range of validity, we developed a model based on a pitch angle distribution proposed by Badhwar & Konradi (1990). The Badhwar & Konradi (1990) pitch angle distribution has to be fitted on experimental data (see Heynderickx & Lemaire 1993 for fits to AP-8). It depends on a loss cone angle parameter α_{L} and a shape parameter β and is given by

$$j(\alpha) = \begin{cases} K\,\xi_\alpha \exp(-\beta\xi_\alpha) & \text{when } \alpha_{\mathrm{L}} < \alpha < \pi - \alpha_{\mathrm{L}} \\ 0 & \text{otherwise} \end{cases} \tag{2.9}$$

where K is a normalisation constant,

$$\xi_\alpha = \frac{\sin \alpha - \sin \alpha_{\mathrm{L}}}{\sqrt{B}} \, , \tag{2.10}$$

and B is the local magnetic field intensity.

We combined this pitch angle distribution with the Lenchek & Singer (1962) East-West asymmetry function to deduce a new anisotropy conversion factor (Kruglanski & Lemaire

1996). For $\alpha_{\mathrm{L}} < \alpha < \pi - \alpha_{\mathrm{L}}$, the conversion factor is given by

$$W_{\mathrm{BK}} = \frac{\xi_\alpha \, \exp(-\beta\xi_\alpha) \, \exp\left(\dfrac{r_{\mathrm{g}}\cos I \sin\alpha \sin\phi}{H}\right)}{4\pi \displaystyle\int_{\alpha_{\mathrm{L}}}^{\pi/2} \left[\mathrm{I}_0\left(\dfrac{r_{\mathrm{g}}\cos I \sin\alpha'}{H}\right) \xi_{\alpha'} \exp(-\beta\xi_{\alpha'}) \, d\cos\alpha'\right]} \,, \qquad (2.11)$$

outside this range it is 0. We combined Eq. (2.11) with the NASA models AP-8 MIN and AP-8 MAX to produce new models called BK-MIN and BK-MAX, respectively. The parameters α_{L} and β are functions of B and L. These functions are obtained from a fit of the pitch angle distribution to the AP-8 unidirectional proton fluxes at 20 MeV (Heynderickx & Lemaire 1993). The scale height H is fixed to 100 km.

By combining Eqs. (2.8) and (2.11) the trapped proton unidirectional differential fluxes at any position in space can be evaluated. ANISO applies these expressions are applied along a satellite orbit to deduce a unidirectional fluence for a set of directions specified by the user (see Sect. 2.2.3). The program provides differential as well as integral fluences.

2.7.2   Input and output

Input files

ANISO takes as input the namelist ANISO, reads the geodetic and (B, L) coordinates from the common interface file PROJECT.INT, the satellite attitude from the attitude interface file PROJECT.ATT, and the magnetic field vector components and the full proton flux spectrum from the file PROJECT.SPP.

The namelist parameters controlling the program are listed in Table 2.12. The polar and azimuthal angles DALPH and DBETA are defined in the coordinate system $(\bar{1}_x, \bar{1}_y, \bar{1}_z)$ attached to the satellite.

Note that the trapped proton anisotropy models have to be used with the appropriate omnidirectional spectra. VF1MIN and BK-MIN are defined with the NASA model AP-8 MIN for solar minimum, while VF1MAX and BK-MAX are defined for the solar maximum model AP-8 MAX. ANISO produces a warning when the omnidirectional model and the anisotropy model are not consistent.

Output files

The file PROJECT.TRD produced by ANISO contains the orbit-averaged integral and differential spectra of trapped protons for the different look directions given in (DALPH = ζ, DBETA = η). The format of the file is described in Table 2.13. For each look direction, the

Table 2.12. NAMELIST parameters for ANISO

Parameter	Data Type	Default	Function
JANIS	INTEGER	1	Anisotropy model number 1: VF1MIN model 2: VF1MAX model 3: BK-MIN model 4: BK-MAX model
NDIR	INTEGER	180	Number of look directions (12×15)
DALPH	REAL*4(400)		Polar angle for each direction in degrees. Default: $\text{DALPH}(j + 15 * i + 1) = 7.5 + 15i$ where i varies from 0 to 11 and j from 0 to 14.
DBETA	REAL*4(400)		Azimuthal angle for each direction in degrees. Default: $\text{DBETA}(j + 15 * i + 1) = 24j$ where i varies from 0 to 11 and j from 0 to 14.
XOMEGA	REAL*4(400)	0	Solid angle (sr) for each direction. When XO-MEGA is set to zero, XALPH and XBETA are used as polar and azimuthal opening angle to compute the solid angle (sr).
XALPH	REAL*4(400)	15	Polar opening angle (deg)
XBETA	REAL*4(400)	24	Azimuthal opening angle (deg)
IFULL	INTEGER	0	When IFULL> 0, the file PROJECT.SPD is produced.

orbit-averaged trapped proton differential spectra is evaluated as

$$\bar{j}(E, \zeta, \eta) = \frac{\Delta\Omega}{\sum\limits_{\bar{p}} \delta t(\bar{p})} \sum_{\bar{p}} W(E, \bar{p}, \zeta, \eta) \left[-\frac{dJ(E, \bar{p})}{dE} \right] , \qquad (2.12)$$

where the summation is taken over each orbital point, δt is the elapsed time between two successive orbit points, $\Delta\Omega$ the solid opening angle (XOMEGA), $J(E, \bar{p})$ the TREP omni-directional integral spectrum and W the anisotropy correction factor where the look direction (ζ, η) is related to the correct proton velocity direction (α, ϕ). The orbit-averaged integral spectra are computed as

$$\overline{J}(E, \zeta, \eta) = \int_E^\infty \bar{j}(E', \zeta, \eta) dE' . \qquad (2.13)$$

When the namelist parameter IFULL is greater than zero, the file PROJECT.SPD is

Table 2.13. Format of the file PROJECT.TRD. When more than one trajectory is specified, the whole structure is repeated.

Record	Format	Description
		Header Records
1	1X,A80	Project header
2	5H E-W ,A8	Omnidirectional trapped proton model header
3	1X,A32	Header of the internal geomagnetic field model
4	1X,A32	Header of the external geomagnetic field model
5	I3,3X,I3,3X, F8.1,12X,I3	Numbers of internal and external field models, epoch for internal magnetic field model and number of look directions
6	F6.1,2F8.1	Orbit inclination (deg), perigee and apogee height (km)
7	34X,E11.4, 4X,A6	Total orbit time (hrs) and anisotropy model header
		Data Records
8	2F10.5	Polar and azimuthal angle (deg) of the first look direction
9	2F10.5,F20.6	Polar and azimuthal opening angle (deg) and solid opening angle ($100 = 4\pi$ sr)
10	I4,16X,2I4	Number of energies (NENERP) in trapped proton spectra, index of the look direction and number of look directions
11	3E11.4	Proton energy (MeV), integral ($\mathrm{cm}^{-2}\mathrm{s}^{-1}$) and differential ($\mathrm{cm}^{-2}\mathrm{s}^{-1}\mathrm{keV}^{-1}$) proton flux
$\vdots$		$\vdots$
10+NENERP	3E11.4	Proton energy, integral and differential flux
11+NENERP		Blank line
12+NENERP		Blank line
13+NENERP		Blank line
14+NENERP		Same as lines 8–($13 + $ NENERP) for the second look direction
$\vdots$		$\vdots$

Table 2.14. Format of the file `PROJECT.SPD`. When more than one trajectory is specified, the whole structure is repeated.

Record	Format	Description
1	`7X,I12,12X,3I12`	Number of look directions (`NDIR`) and starting date (year, month, day)
2	`7X,<NDIR>F12.2`	Polar angle `DALPH` (deg) for each direction
3	`7X,<NDIR>F12.2`	Azimuthal angle `DBETA` (deg) for each direction
4	`7X,I12,12X,` `3F12.2`	Number of energies (`NENERP`), altitude, latitude, longitude
5	`F7.1,<NDIR>E12.4`	Energy and conversion factor W for each direction
⋮		⋮
4+NENERP	`F7.1,<NDIR>E12.4`	Energy and conversion factor for each direction
5+NENERP		Same as lines 4–(4+NENERP) for the second orbital point
⋮		⋮
		Blank line

produced. This file contains the angular distribution of the trapped proton fluxes along the orbit. The format of this file is described in Table 2.14.

2.8   ANISOPOS

As a secondary product of `ANISO`, a standalone program called `ANISOPOS` is provided. This program evaluates the trapped proton anisotropy at a user-specified location. `ANISOPOS` works interactively with an on-screen menu described in Table 2.15.

The user has to supply a project name, a title, a trapped proton integral flux and a point in space, and to select an anisotropy model as well as an internal geomagnetic field model. The integral flux is defined either as a power law or exponential function defined by the integral flux at two different energies, or by a `PROJECT.TRI` file.

`ANISOPOS` calculates the unidirectional differential and integral trapped proton fluxes and produces the file `PROJECT.TRD` and a new `PROJECT.TRI` file, overwriting an existing one. The file `PROJECT.TRI` contains the omnidirectional fluxes.

Table 2.15. Interactive menu options of `ANISOPOS`

Code		Action
1		Modify the title
2		Change the geomagnetic field model:
	21	− IGRF/DGRF (default)
	22	− Jensen and Cain 1962
	23	− GSFC 12/66
3		Modify the epoch of the geomagnetic field (default: 1995)
4		Change the trapped proton anisotropy model:
	41	− VF1MIN (default)
	42	− VF1MAX
	43	− BK-MIN
	44	− BKMAX
5		Change the omnidirectional integral flux:
	51	− Power law spectrum, defined by two points (default)
	52	− Exponantially decreasing spectrum, defined by two points
	53	− `PROJECT.TRI`
6		Modify the first point of the spectrum (for options 51 and 52)
7		Modify the second point of the spectrum (for options 51 and 52)
8		Specify the geodetic altitude (default: $500\,\mathrm{km}$)
9		Specify the geodetic latitude (default: $-35°$)
10		Specify the geographic longitude (default: $300°$)
11		Evaluate the (B, L) coordinates
0		Run the model
−1		Exit and print the results

2.9 SHIELDOSE

`SHIELDOSE` (Seltzer 1979, 1980) makes use of pre-calculated, mono-energetic depth-dose data for an isotropic fluence of radiation incident on uniform aluminium plane media and integrates these mono-energetic data over the incident flux spectrum. The output of `SHIEL-DOSE` consists of doses in four different detector materials (Al, H_2O, Si, SiO_2) as a function of shield depth for three different shielding geometries: finite slab, semi-infinite, spherical.

Table 2.16. NAMELIST parameters for SHIELDOSE.

Parameter	Data Type	Default	Function
TINTER	REAL*8	365.25	Mission duration (days)
IDET	INTEGER	3	Selects detector material: 1: AL detector 2: H_2O detector 3: SI detector 4: SIO_2 detector
IUNT	INTEGER	3	Selects units for shield depths: 1: Shield depths in mils 2: Shield depths in $\mathrm{g\,cm^{-2}}$ 3: Shield depths in mm
IMAX	INTEGER	25	Number of shield depth values
Z	REAL*4(50)		Shield depth values. Defaults: 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.8, 1.0, 1.5, 2.0, 2.5, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0, 12.0, 14.0, 16.0, 18.0, 20.0, 22.0, 24.0, 26.0, 28.0, 30.0, 32.0, 34.0, 36.0, 38.0, 40.0, 44.0, 48.0, 52.0, 56.0, 60.0, 64.0, 68.0, 72.0, 76.0, 80.0, 84.0, 88.0, 92.0, 96.0, 100.0
ISHLD	INTEGER	4	Selects shield configurations
IPRINT	INTEGER	0	Dose spectra are listed when set to 1
IPLOT	INTEGER	1	0: No graphical output 1: Output file PROJECT.SHG is created

2.9.1 Input files

SHIELDOSE takes as input the NAMELIST SHIELDOSE and the interface file PRO-JECT.TRI created by TREP, TREPPOS, or TREPAVE. The interface file contains the orbit-averaged spectra of trapped protons and electrons, and of solar protons if a solar flare model was specified.

The NAMELIST parameters for SHIELDOSE are listed in Table 2.16 with their data type and default value, and a brief description.

The number of shield depth values is input with the NAMELIST parameter IMAX. The shield depth values themselves are contained in the vector Z. By default, the units of Z are mm. By setting the parameter IUNT to 1 or 2, the values in Z are interpreted as mils or $\mathrm{g\,cm^{-2}}$, respectively.

Table 2.17. Format of the plot files generated by `SHIELDOSE`, `EQFRUX`, and `EQFRUXGA`. When more than one trajectory is in the interface file, the whole structure is repeated. The first set of plots is not present in `PROJECT.SHG`. The format `A*` means that the length of the string is variable.

Record	Format	Description
1	`1X,A80`	Project header
2	`1X,A*`	Mission description
3	`6I3`	Number of plots, number of values of dependent variable, number of independent variables, for full and summary plots (NPF, NDF, NIF, NPS, NDS, NIS)
4	`1X,A*`	General plot title
5	`1X,A*`	Vertical axis label
6	`1X,A*`	Horizontal axis label and NIF legends (separated by commas)
7...(6+NDF)	`(1+NIF)E11.4`	Independent variable and NIF dependent variables
⋮	`(1+NIF)E11.4`	NPF−1 repetitions of records 4...(6+NDF)
4+(3+NDF)*NPF	`1X,A*`	General plot title
5+(3+NDF)*NPF	`1X,A*`	Vertical axis label
6+(3+NDF)*NPF	`1X,A*`	Horizontal axis label and NIS legends (separated by commas)
7+(3+NDF)*NPF... 6+(3+NDF)*NPF+NDS	`(1+NIS)E11.4`	Independent variable and NIS dependent variables
⋮	`(1+NIS)E11.4`	NPS−1 repetitions of records 7+(3+NDF)*NPF to 6+(3+NDF)*NPF+NDS

The shield configuration is selected with the `NAMELIST` parameter `ISHLD`. This parameter can take the values 1, 2, and 4 for finite slab, semi-infinite, and spherical, respectively, and combinations of these three values. For instance, `ISHLD=7` will generate prints and plots for all three shield configurations, `ISHLD=3` selects only the infinite slab and semi-infinite configurations. The default value is 4.

Table 2.18. NAMELIST parameters for EQFRUX

Parameter	Data Type	Default	Function
TINTER	REAL*8	365.25	Mission duration (days)
BSF	REAL*8	1	Backshielding factor
PEDRI	REAL*8	3000	Damage ratio between protons and electrons for ISC
PEDRV	REAL*8	3000	Damage ratio between protons and electrons for VOC
IPRINT	INTEGER	0	0: Summary tables of flux vs. depth only 1: Tables of equivalent fluence vs. energy as well
IPLOT	INTEGER	1	0: No graphical output 1: Only summary plots of flux vs. depth 2: Full difference fluence spectra vs. energy as well

2.9.2 Output files

SHIELDOSE produces the report file PROJECT.SHP, and the plot file PROJECT.SHG when the NAMELIST parameter IPLOT=1. The format of the file PROJECT.SHG is given in Table 2.17. The same format is used for the plot files PROJECT.EQG and PROJECT.EGG created by EQFRUX and EQFRUXGA, respectively. For SHIELDOSE, the first three numbers in the third record always are zero, since there is only one series of output plots.

2.10   EQFRUX

EQFRUX calculates 1 MeV damage equivalent electron fluences for exposure to the fluences predicted by TREP, TREPAVE or TREPPOS, for a specified duration. The conversion to damage equivalent fluences is based on damage ratios published by JPL for Si cells (Tada et al. 1982).

2.10.1   Input files

EQFRUX takes as input the NAMELIST EQFRUX and the interface file PROJECT.TRI created by TREP, TREPPOS, or TREPAVE. This file contains the orbit-averaged fluence spectra of trapped protons and electrons, and of solar protons if a solar flare model was specified. The NAMELIST parameters for EQFRUX are listed in Table 2.18 with their data type and default value, and a brief description.

Table 2.19. `NAMELIST` parameters for `EQFRUXGA`

Parameter	Data Type	Default	Function
TINTER	REAL*8	365.25	Mission duration (days)
BSF	REAL*8	1	Backshielding factor
PEDRI	REAL*8	400	Damage ratio between protons and electrons for ISC
PEDRV	REAL*8	1400	Damage ratio between protons and electrons for VOC
PEDRP	REAL*8	1000	Damage ratio between protons and electrons for P-MAX
IPRINT	INTEGER	0	0: Summary tables of flux vs. depth only 1: Tables of equivalent fluence vs. energy as well
IPLOT	INTEGER	1	0: No graphical output 1: Summary plots of flux vs. depth only 2: Full difference fluence spectra vs. energy as well

The `NAMELIST` parameter `TINTER` specifies the mission duration in days. It is used for the trapped particle fluxes only. For solar flares, the fluence spectrum produced by `TREP` is used and is not dependent on the value of `TINTER`.

2.10.2   Output files

`EQFRUX` produces the report file `PROJECT.EQP`, and the plot file `PROJECT.EQG` when the `NAMELIST` parameter `IPLOT> 0`. The format of the file `PROJECT.EQG` is given in Table 2.17. The same format is used for the plot files `PROJECT.SHG` and `PROJECT.EGG` created by `SHIELDOSE` and `EQFRUXGA`, respectively.

2.11   EQFRUXGA

`EQFRUXGA` calculates 1 MeV damage equivalent electron fluences for exposure to the fluences predicted by `TREP`, `TREPAVE` or `TREPPOS`, for a specified duration. The conversion to damage equivalent fluences is based on damage ratios published by JPL for GaAs cells.

2.11.1   Input files

EQFRUXGA takes as input the NAMELIST EQFRUX and the interface file PROJECT.TRI created by TREP, TREPPOS, or TREPAVE. This file contains the orbit-averaged fluence spectra of trapped protons and electrons, and of solar protons if a solar flare model was specified. The NAMELIST parameters for EQFRUXGA are listed in Table 2.19 with their data type and default value, and a brief description.

The NAMELIST parameter TINTER specifies the mission duration in days. It is used for the trapped particle fluxes only. For solar flares, the fluence spectrum produced by TREP is used and is not dependent on the value of TINTER.

2.11.2   Output files

EQFRUXGA produces the report file PROJECT.EGP, and the plot file PROJECT.EGG when the NAMELIST parameter IPLOT> 0. The format of the file PROJECT.EGG is given in Table 2.17. The same format is used for the plot files PROJECT.SHG and PROJECT.EQG created by SHIELDOSE and EQFRUX, respectively.

Chapter 3

The plotting program UNIRAD.PRO

The IDL[1] routine `UNIRAD.PRO` is a menu driven plotting program that provides a graphical representation of the various output files produced by `UNIRAD`. The plots can be shown on the screen or sent to files in PostScript format.

3.1 Function and use

`UNIRAD.PRO` consists of one file with IDL code. The program is run as follows, after starting an IDL session:

1. Compile the program by typing
   `.RUN UNIRAD.PRO` `Return`
   In order for IDL to locate the source file, the path containing the file should be added to the user directory path in the IDL configuration file. Alternatively, the full directory can be specified in the `.RUN` command.

2. To run the program, type
   `UNIRAD,'PROJECT'` `Return`
   where `PROJECT` is the name of the project of the `UNIRAD` run. `PROJECT` can include a directory path.

The program will check which `UNIRAD` output files are available in the current directory (or the directory specified in `PROJECT`), and present a menu with options. On an X terminal the menu is a graphical one and is activated by clicking an option with the mouse. On other terminals the menu is in text format and expects numerical input from the keyboard. After making a selection from the main menu, other menus will appear depending on the `UNIRAD` output files available in the current directory.

---

[1]`UNIRAD.PRO` will also run in PV-Wave. However, abnormally long tick marks may appear in some plots when using PV-Wave. The length of horizontal tick marks can be changed by adding `',tick=value'` on the command line, where value ranges between 0 and 1 (default is 1)

3.2   Output from the program

When a selection is made from the menu, `UNIRAD.PRO` produces a plot on the screen[2]. The user then has the option to produce a PostScript version of this plot by making the appropriate menu selection. The program creates PostScript or encapsulated PostScript files in the current directory (or the directory specified in `PROJECT`), with extension `PS` or `EPS`, respectively. The first part of the file names is of the form `PROJECT_XXX_L_M_N`, where

- `XXX` identifies the type of output (`INT`, `TRI`, `SHG`, `EQG`, or `EGG`);

- `L` is the trajectory number (this is equal to one when only one trajectory was specified with the `UNIRAD` run);

- `M` is used to distinguish summary plots (`M=2`) and full plots (`M=1`) in the case of `SHIELDOSE`, `EQFRUX`, and `EQFRUXGA`. For the other programs `_M` is not used.

- `N` identifies the selection made in the lowest level menu.

The top level menu contains the option "All plots in PostScript format" and "All plots in encapsulated PostScript format". If one of these options is selected, the program will generate automatically all plot files that can be produced with the output files present.

---

[2]The appearance of the plot on the screen depends on the device, e.g. X-window, MS-Windows, and may not always be satisfactory. However, the PostScript output is independent of the platform on which IDL is run.

Chapter 4

A sample run of UNIRAD

In this chapter, we present the output of a sample run of `UNIRAD`. The default values for the `NAMELIST` parameters (see Appendix A) were used, except for the parameters listed below. The project name of the sample run is `TEST`. A `NAMELIST` file `TEST.NML` is provided with the distribution.

```
$SAPRE
 TITLE='Test run'
 IATTI=1
$END
$TREP
 FLAMOD=1
 ISPEC=1
$END
$ANISO
 JANIS=4
$END
$SHIELDOSE
 ISHLD=7
$END
$EQFRUX
 IPLOT=2
$END
$EQFRUXGA
 IPLOT=2
$END
```

The plots in this chapter were produced with `UNIRAD.PRO`.

4.1   Output from SAPRE

Figure 4.1 represents the orbit generated for the sample project.

4.2   Output from BLXTRA

The B and L values calculated for the sample orbit are superposed on Fig. 4.1.

4.3   Output from TREP

The positional proton and electron fluxes calculated for the sample orbit are plotted in Figs. 4.2 and 4.3, respectively. Note that the (B, L) values have been recalculated by TREP with the proper geomagnetic field models.

   The orbit averaged proton and electron fluence spectra are represented in Figs. 4.4 and 4.5, respectively. The solar proton spectrum is shown in Fig. 4.6.

4.4   Output from ANISO

The directional dependence of the integral trapped proton flux > 50 MeV, averaged over the sample orbit, is plotted in Fig. 4.7.

4.5   Output from ANISOPOS

Figure 4.8 shows the directional dependence of the integral trapped proton flux at the location given by: altitude 500 km, latitude $-35°$, longitude $300°$, for a power law spectrum.

4.6   Output from SHIELDOSE

The dose curves calculated for the sample orbit are plotted in Figs. 4.9–4.11 for the three shield configurations.

Test run

Apogee: 36000.0 km   Magnetic models (internal, external):
Perigee: 300.0 km   —— GSFC 12/66 120 Term @1970.0,
Inclination: 0.0°   None
                     ···· Jensen & Cain 1960 @1960.0,
                          None
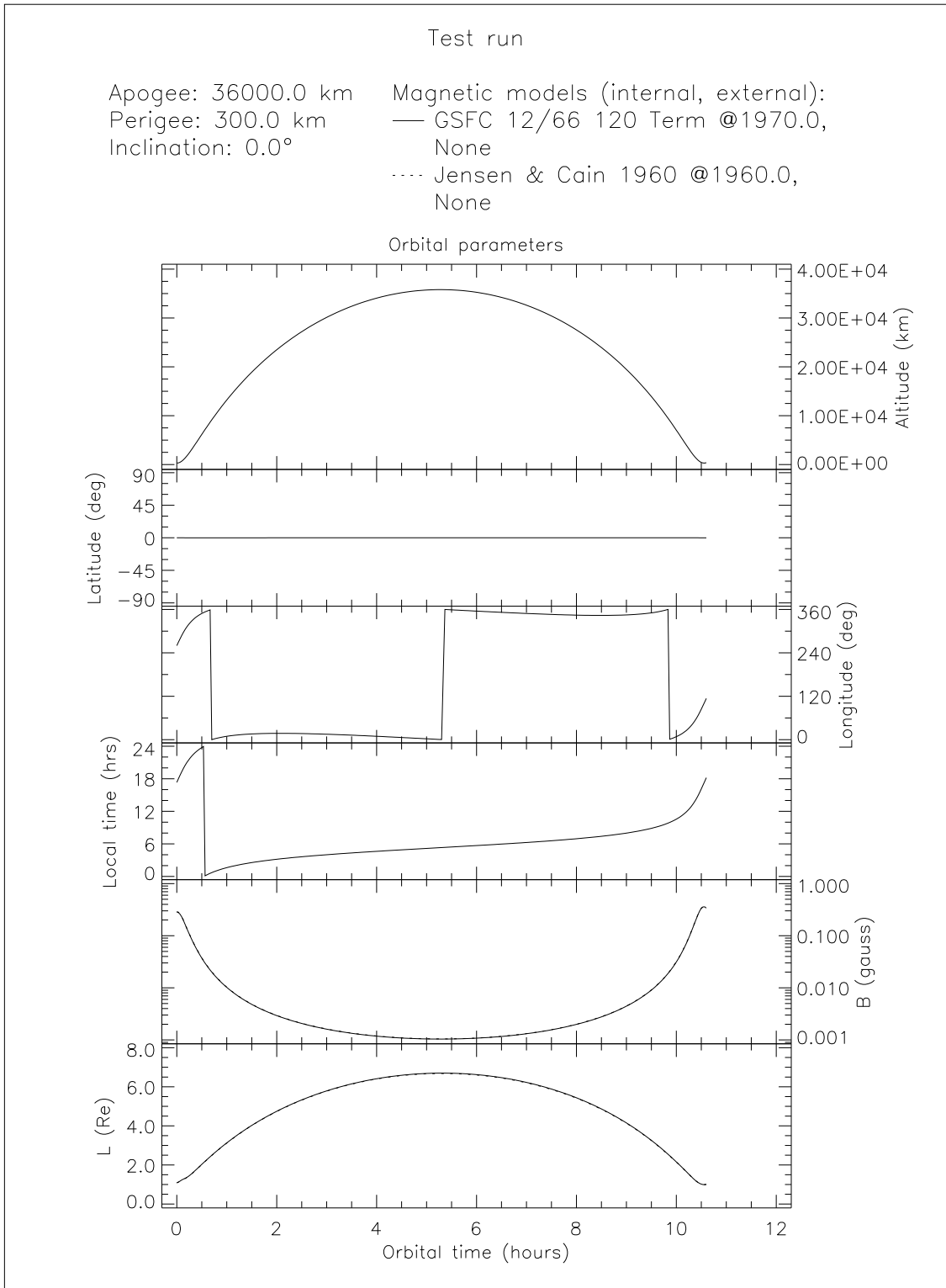
Figure 4.1. Geographic and magnetic coordinates for the sample orbit.
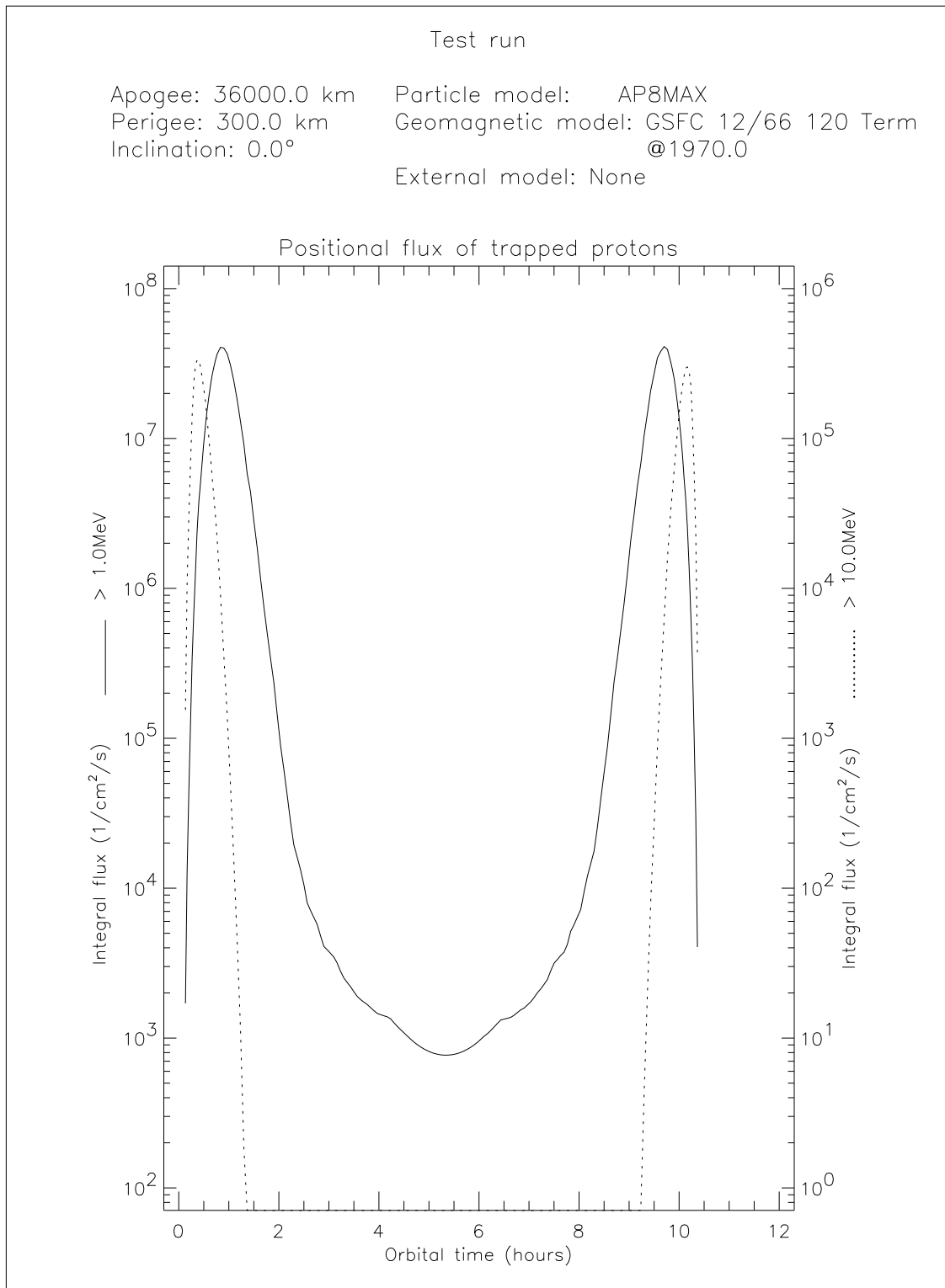
Figure 4.2. Integral trapped proton fluxes for the sample orbit.
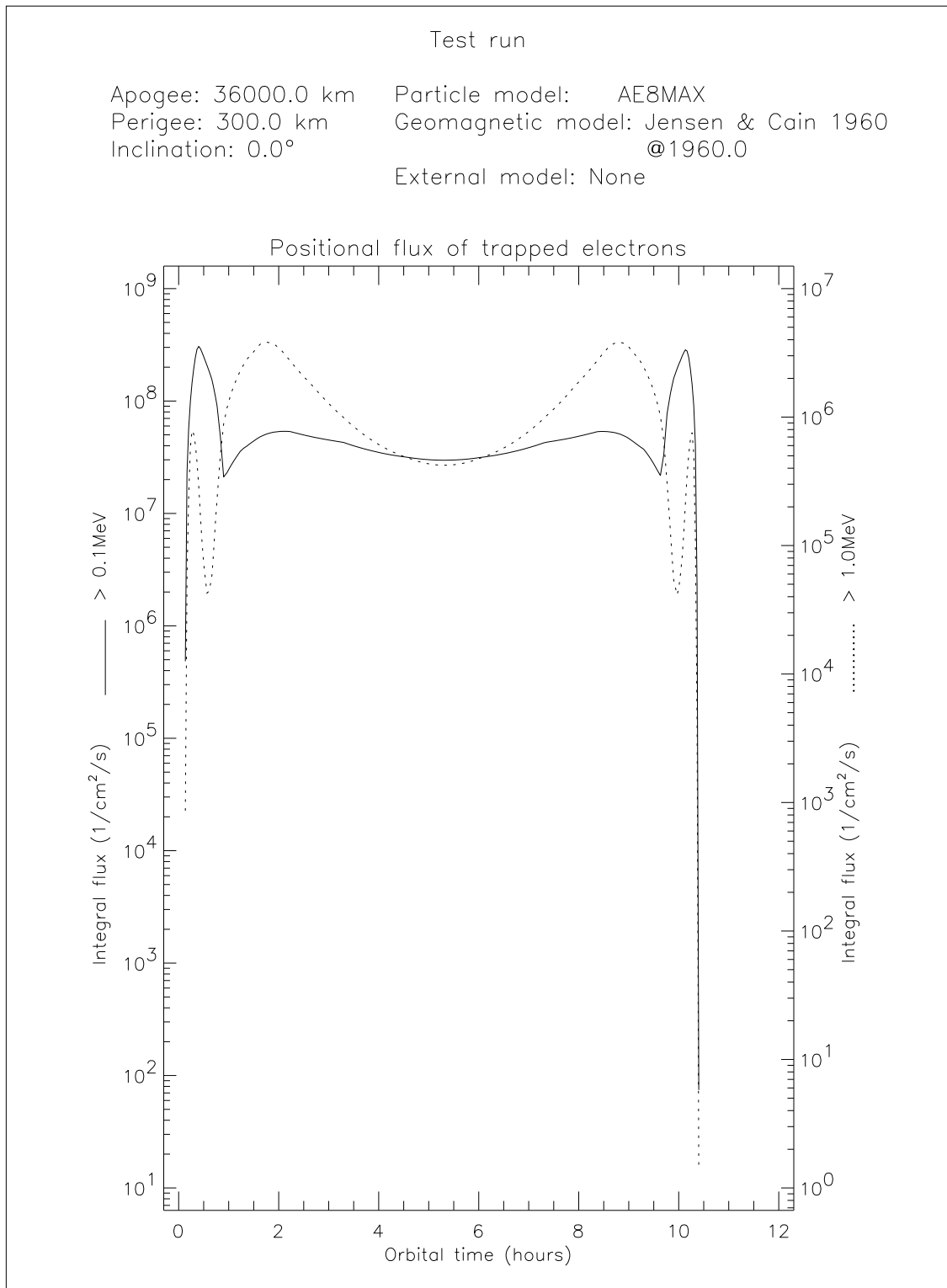
Figure 4.3. Integral trapped electron fluxes for the sample orbit.
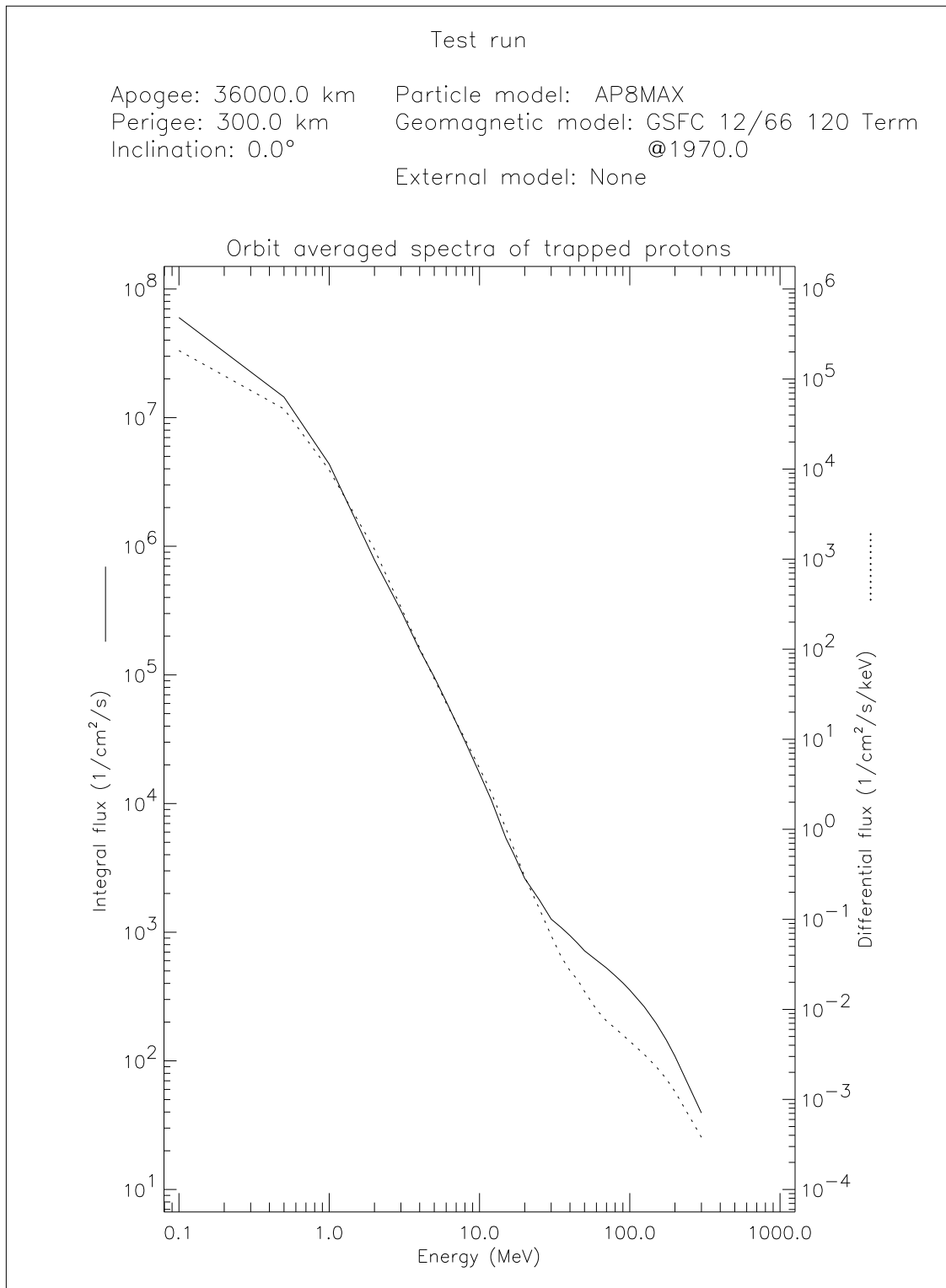
Figure 4.4. Integral and differential trapped proton spectrum for the sample orbit.

Test run

Apogee: 36000.0 km   Particle model: AE8MAX
Perigee: 300.0 km   Geomagnetic model: Jensen & Cain 1960
Inclination: 0.0°   @1960.0
                        External model: None

Orbit averaged spectra of trapped electrons
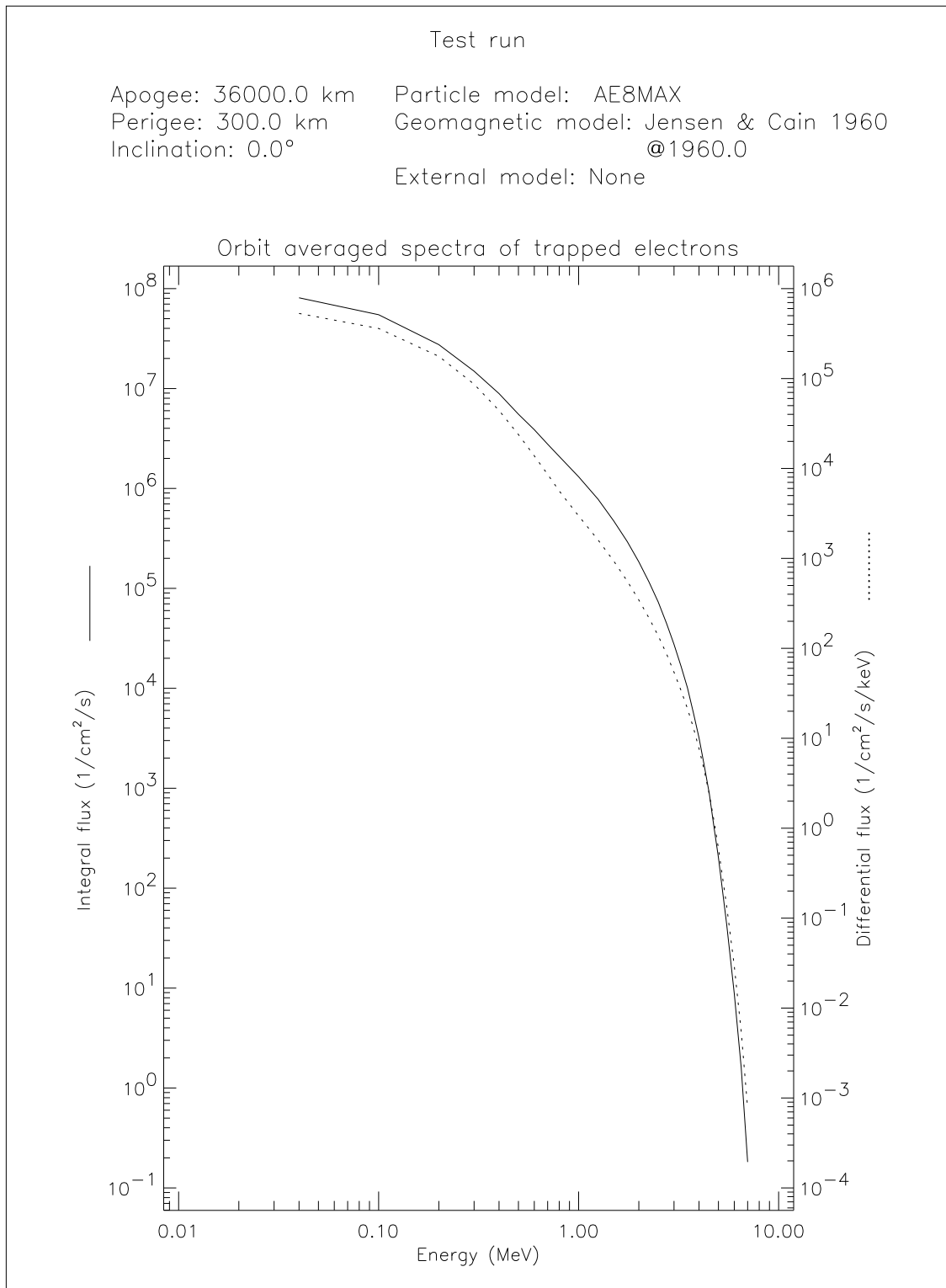
Figure 4.5. Integral and differential trapped electron spectrum for the sample orbit.
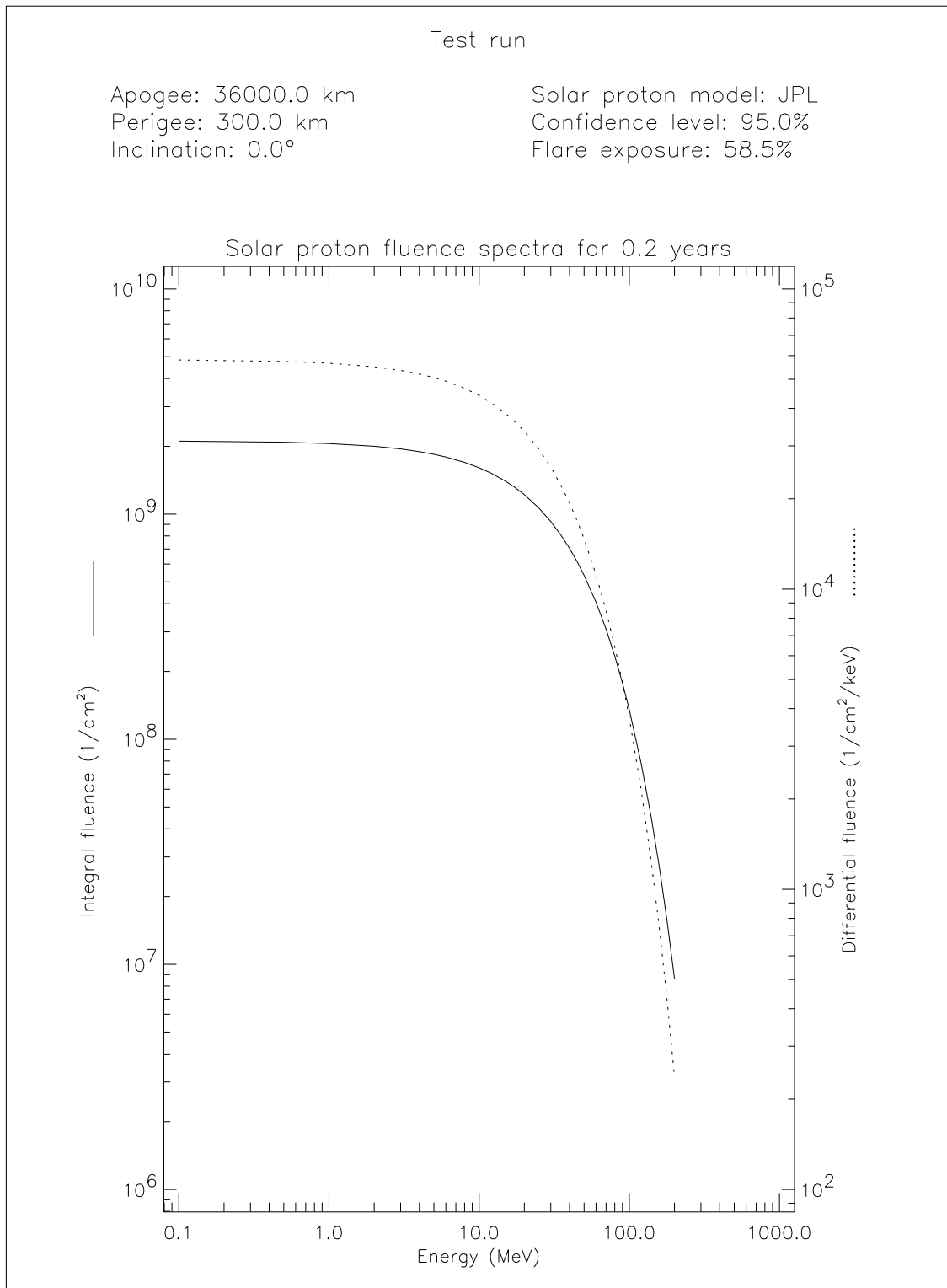
Figure 4.6. Integral and differential solar proton spectrum for the sample orbit.

Trapped proton unidirectional integral flux E > 50.0MeV

incl.=   0.0°, per.=   300.0 km, apo.= 36000.0 km
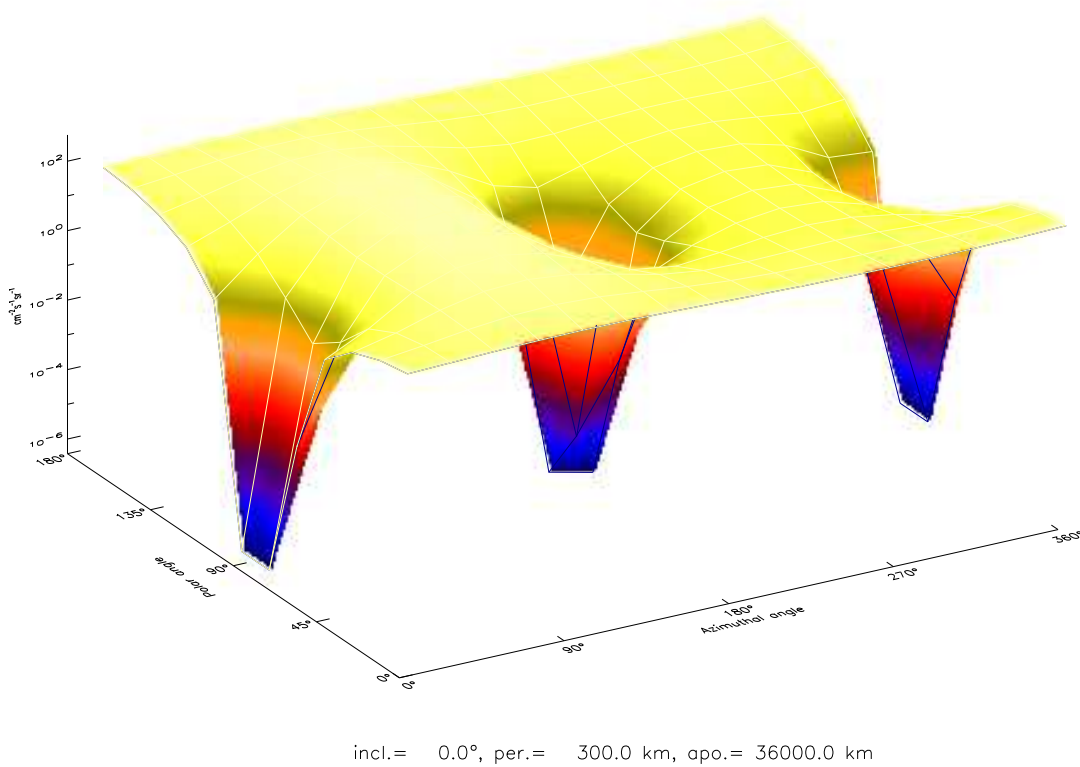
Figure 4.7. Directional dependence of the integral trapped proton flux $> 50\,\text{MeV}$ averaged over the sample orbit.
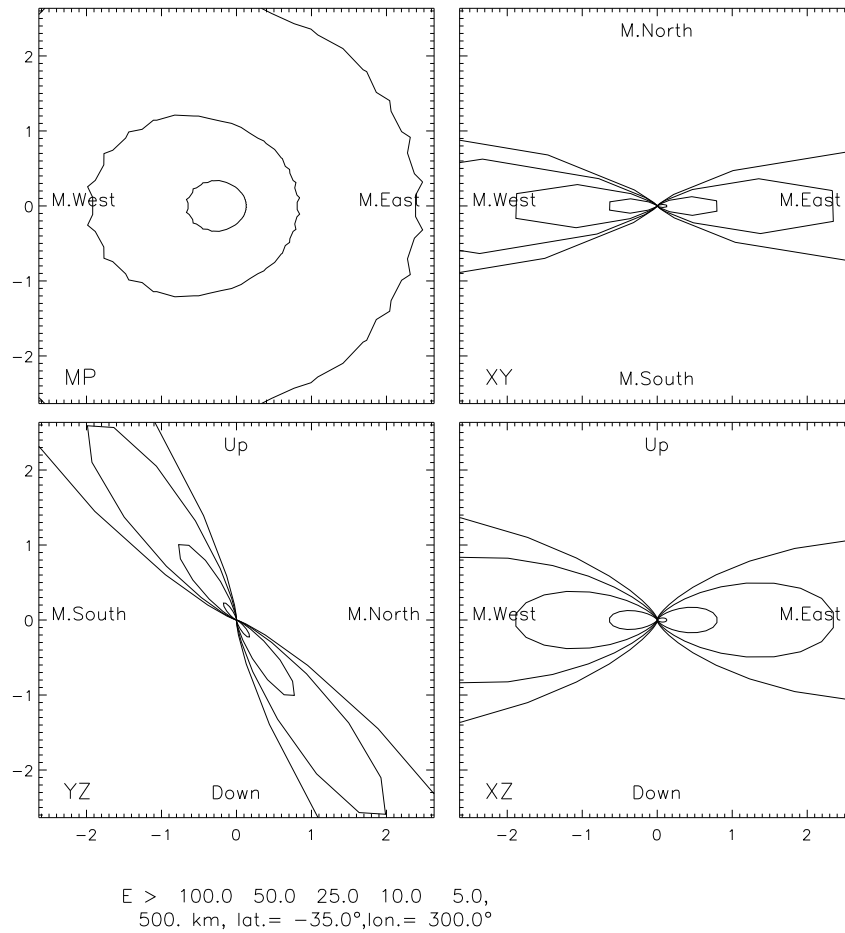
Figure 4.8. Directional dependence of the integral trapped proton flux at the location indicated in the plot for a power law spectrum.

Test run

Apogee: 36000.0 km   Trapped models: AP8MAX   AE8MAX
Perigee: 300.0 km   Solar proton model: 95 % JPL
Inclination: 0.0°
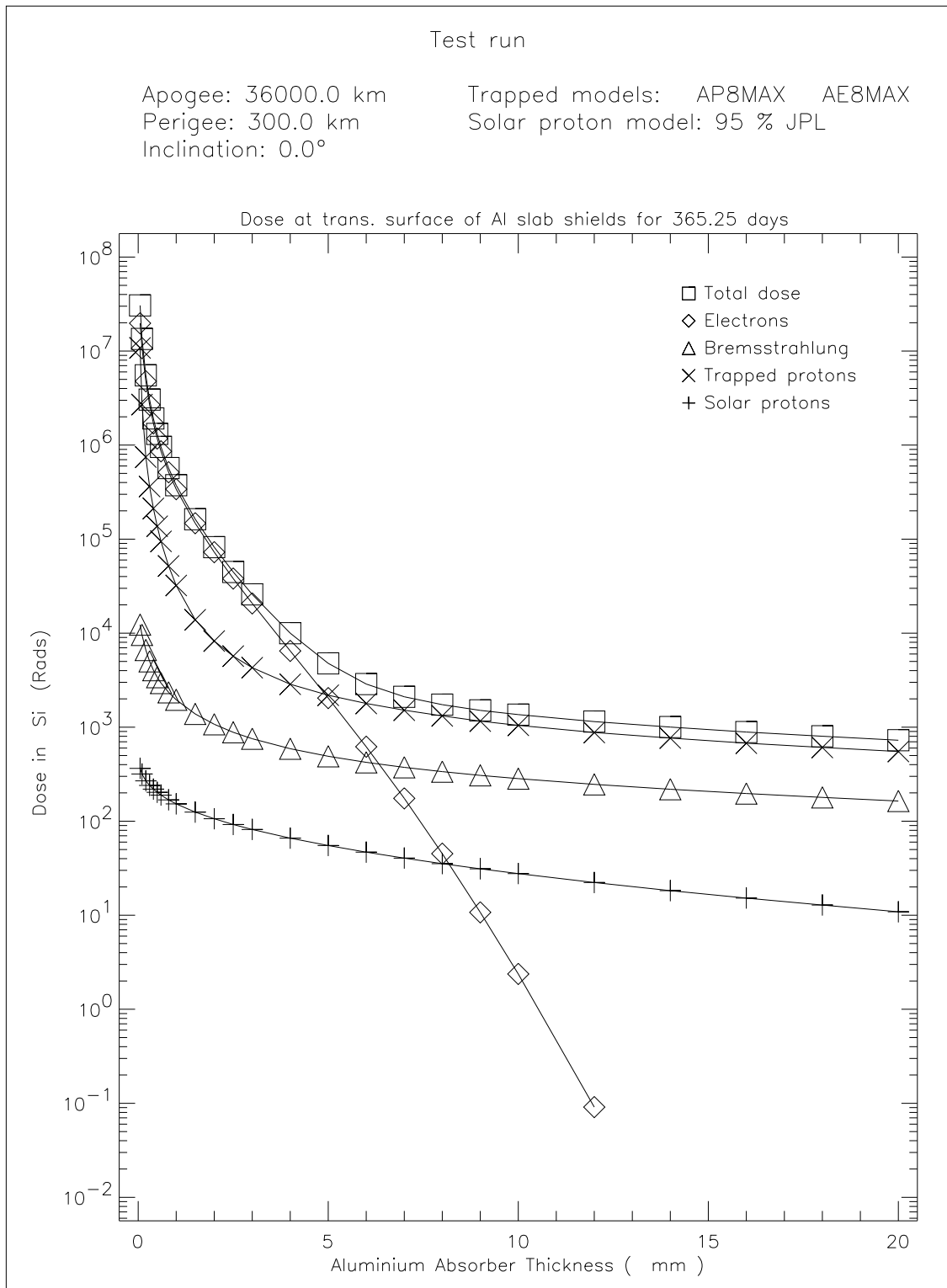
Figure 4.9. Dose in Si behind an Al slab shield for the sample orbit.

Test run

Apogee: 36000.0 km   Trapped models: AP8MAX   AE8MAX
Perigee: 300.0 km   Solar proton model: 95 % JPL
Inclination: 0.0°

Dose in semi−infinite Al medium for 365.25 days
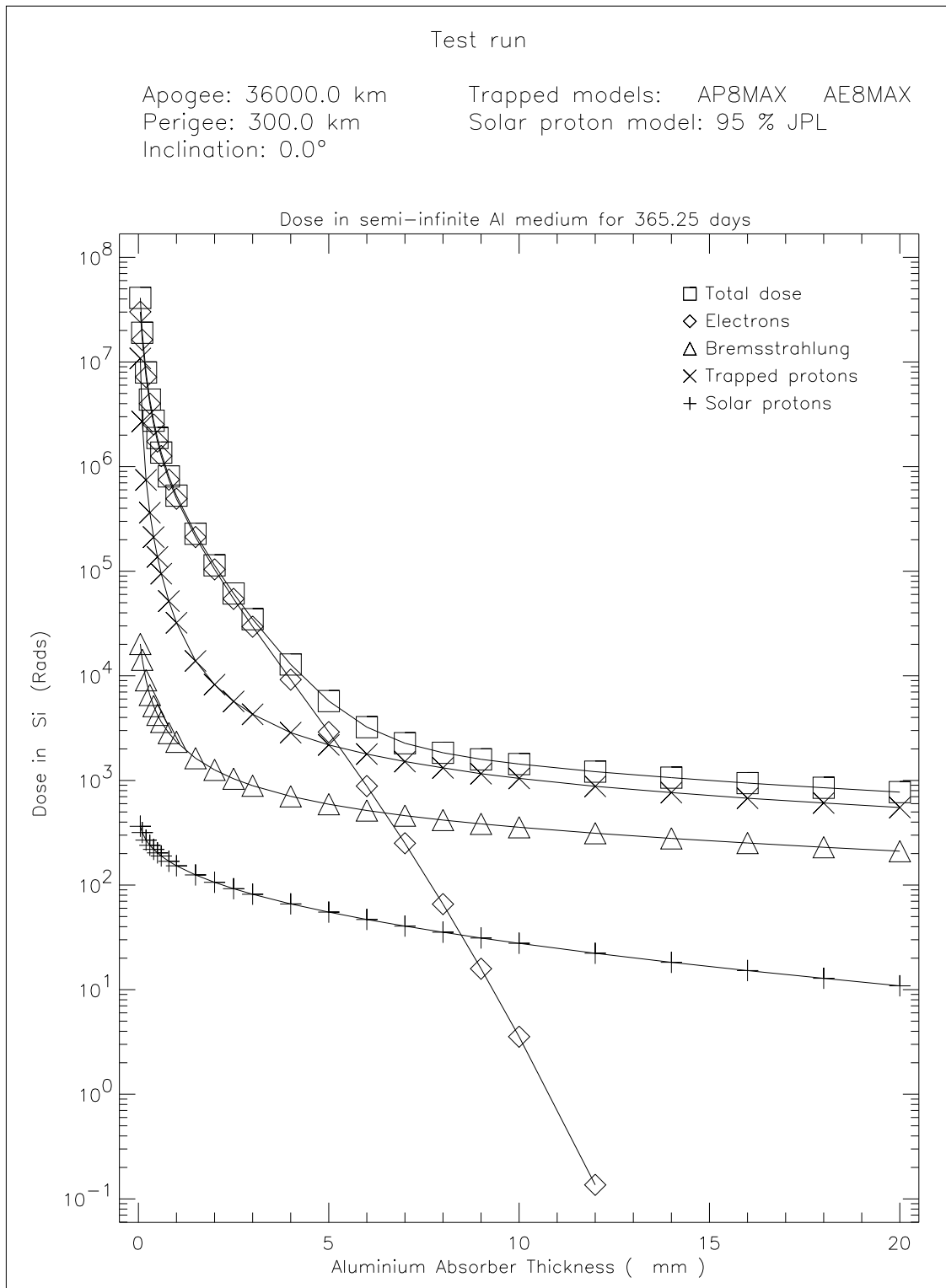
Figure 4.10. Dose in Si behind a semi-infinite Al medium for the sample orbit.

Test run

Apogee: 36000.0 km   Trapped models: AP8MAX   AE8MAX
Perigee: 300.0 km   Solar proton model: 95 % JPL
Inclination: 0.0°

4*PI Dose at centre of Al spheres for 365.25 days
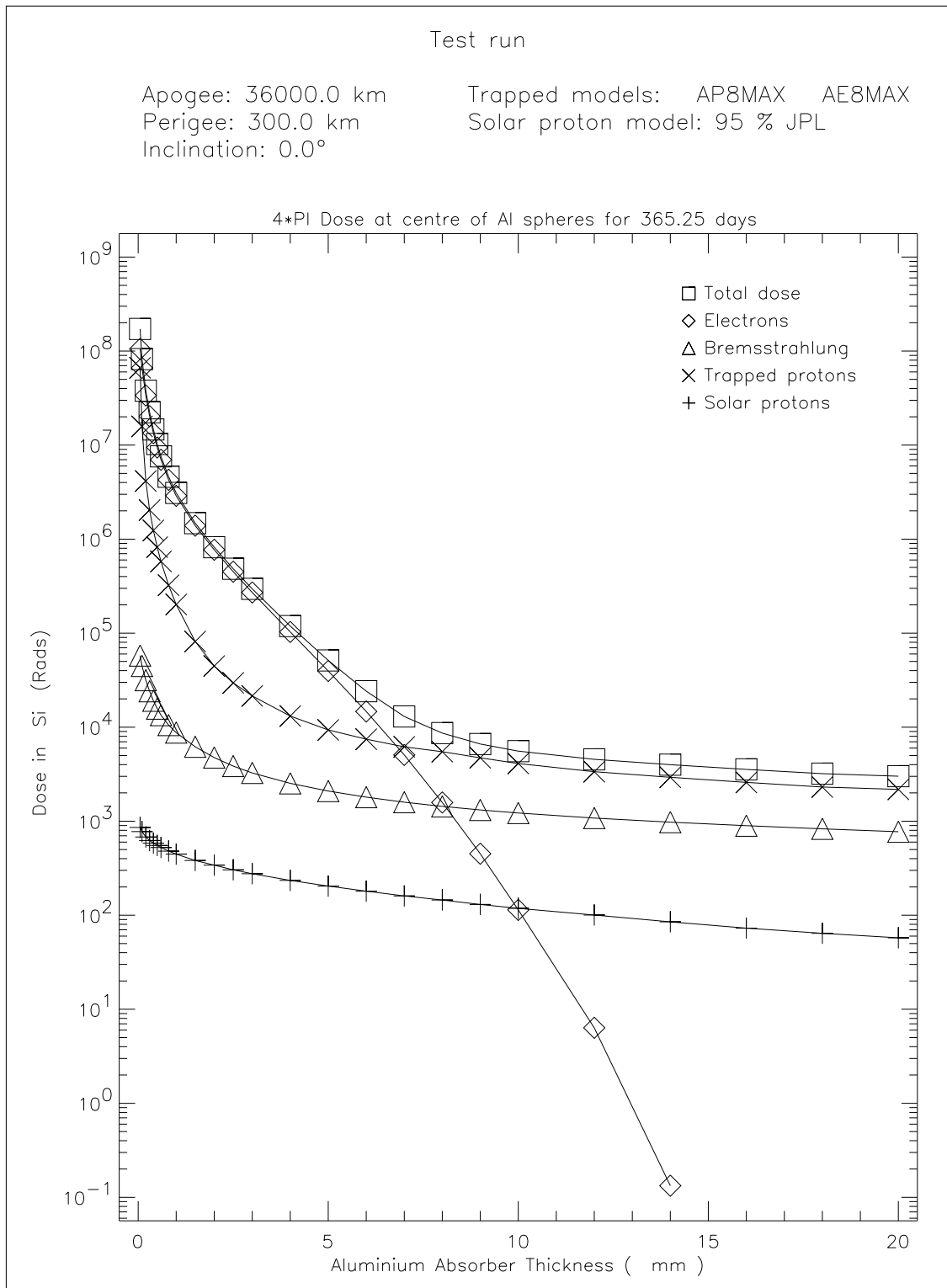
Figure 4.11. Dose in Si at the centre of Al spheres for the sample orbit.

4.7   Output from EQFRUX

The damage equivalent fluxes in Si, calculated for the sample orbit are plotted in Figs. 4.12 and 4.13 as a function of glass cover depth. Figures 4.14–4.18 show the dependence of the damage equivalent fluxes on energy.

4.8   Output from EQFRUXGA

The damage equivalent fluxes in GaAs, calculated for the sample orbit are plotted in Figs. 4.19–4.21 as a function of glass cover depth. Figures 4.22–4.26 show the dependence of the damage equivalent fluxes on energy.
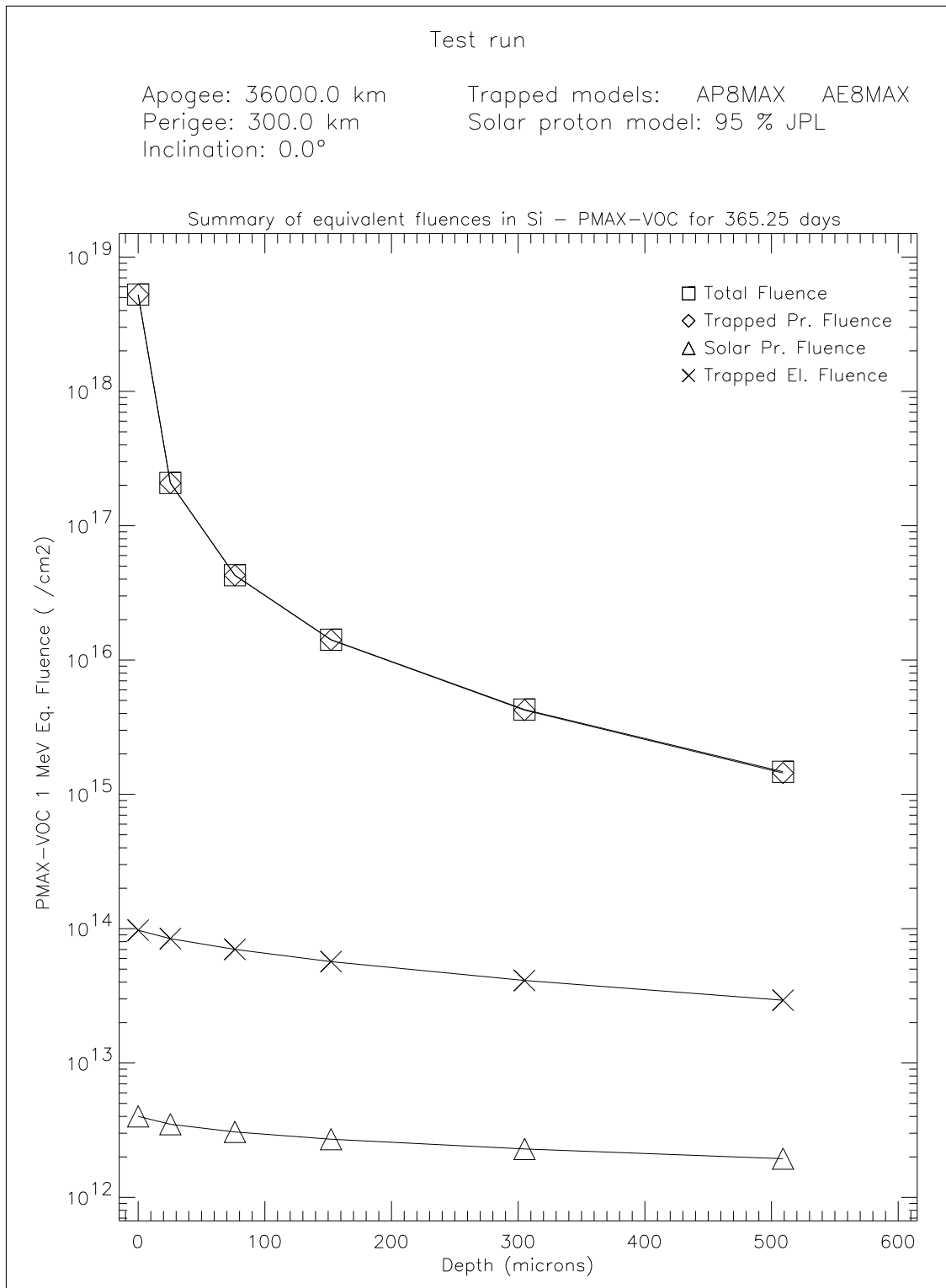
Figure 4.12. Summary of PMAX-VOC equivalent fluences for the sample orbit.

Test run

Apogee: 36000.0 km   Trapped models: AP8MAX   AE8MAX
Perigee: 300.0 km   Solar proton model: 95 % JPL
Inclination: 0.0°

Summary of equivalent fluences in Si − ISC for 365.25 days
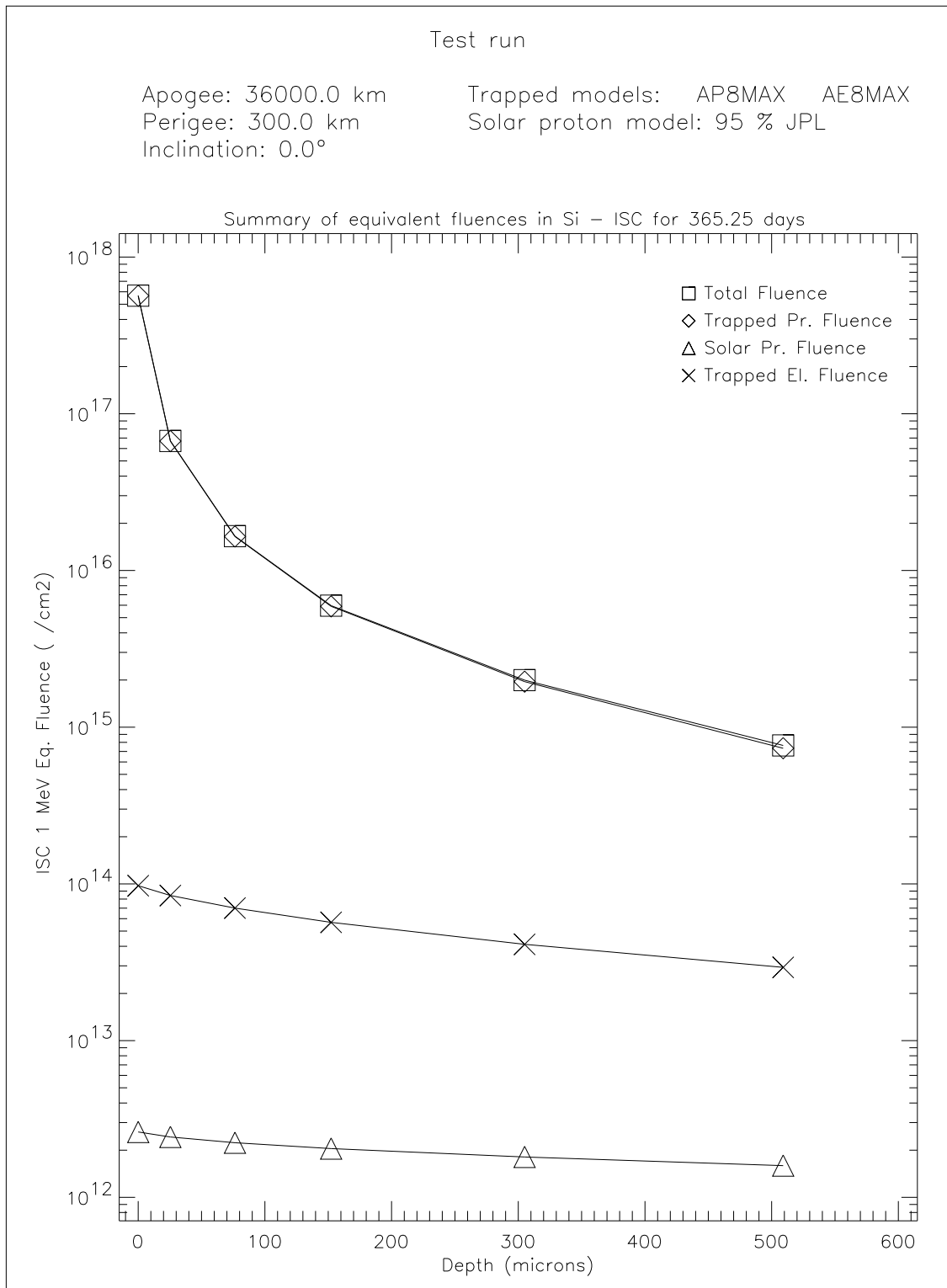
Figure 4.13. Summary of ISC equivalent fluences for the sample orbit.
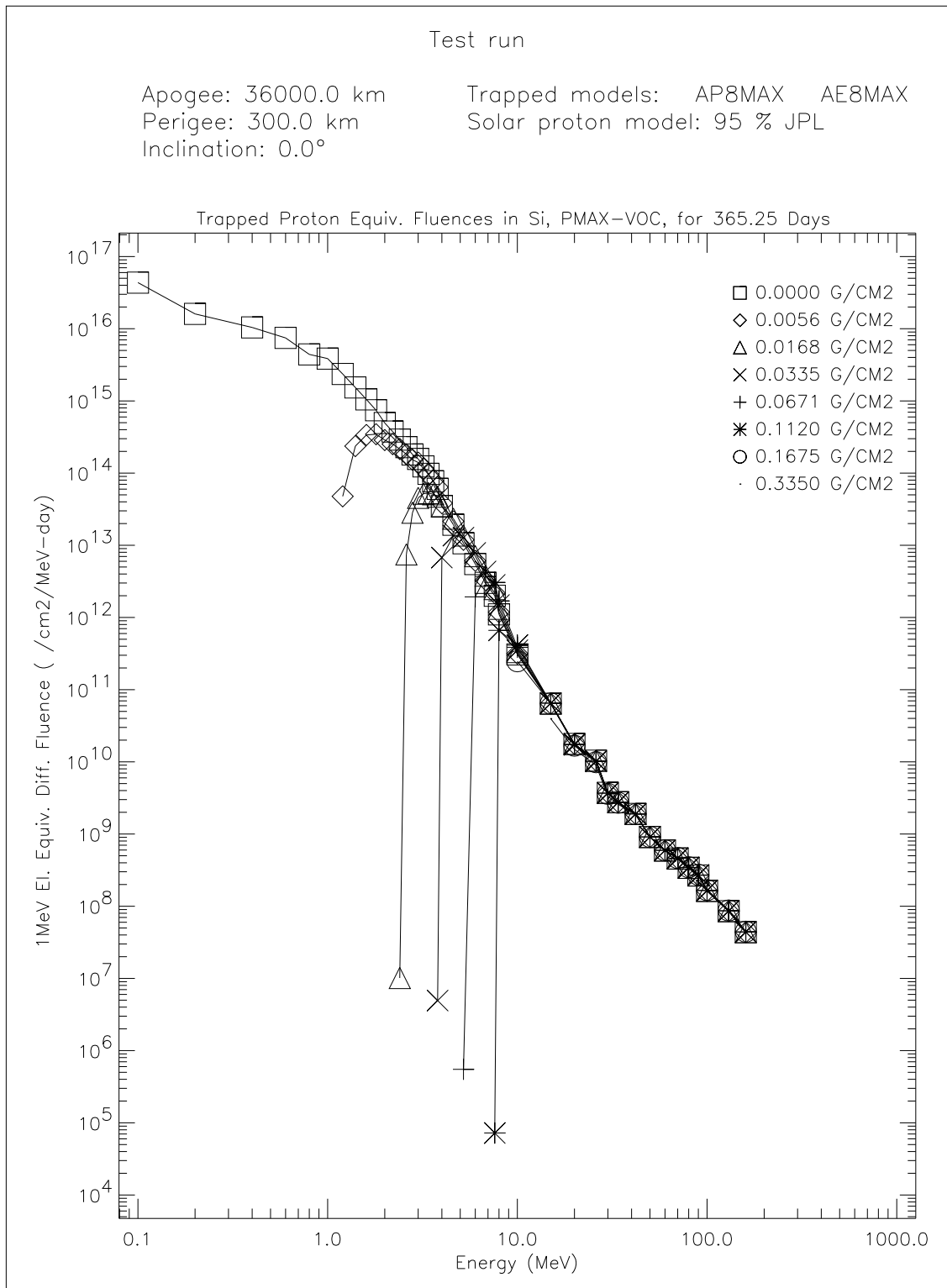
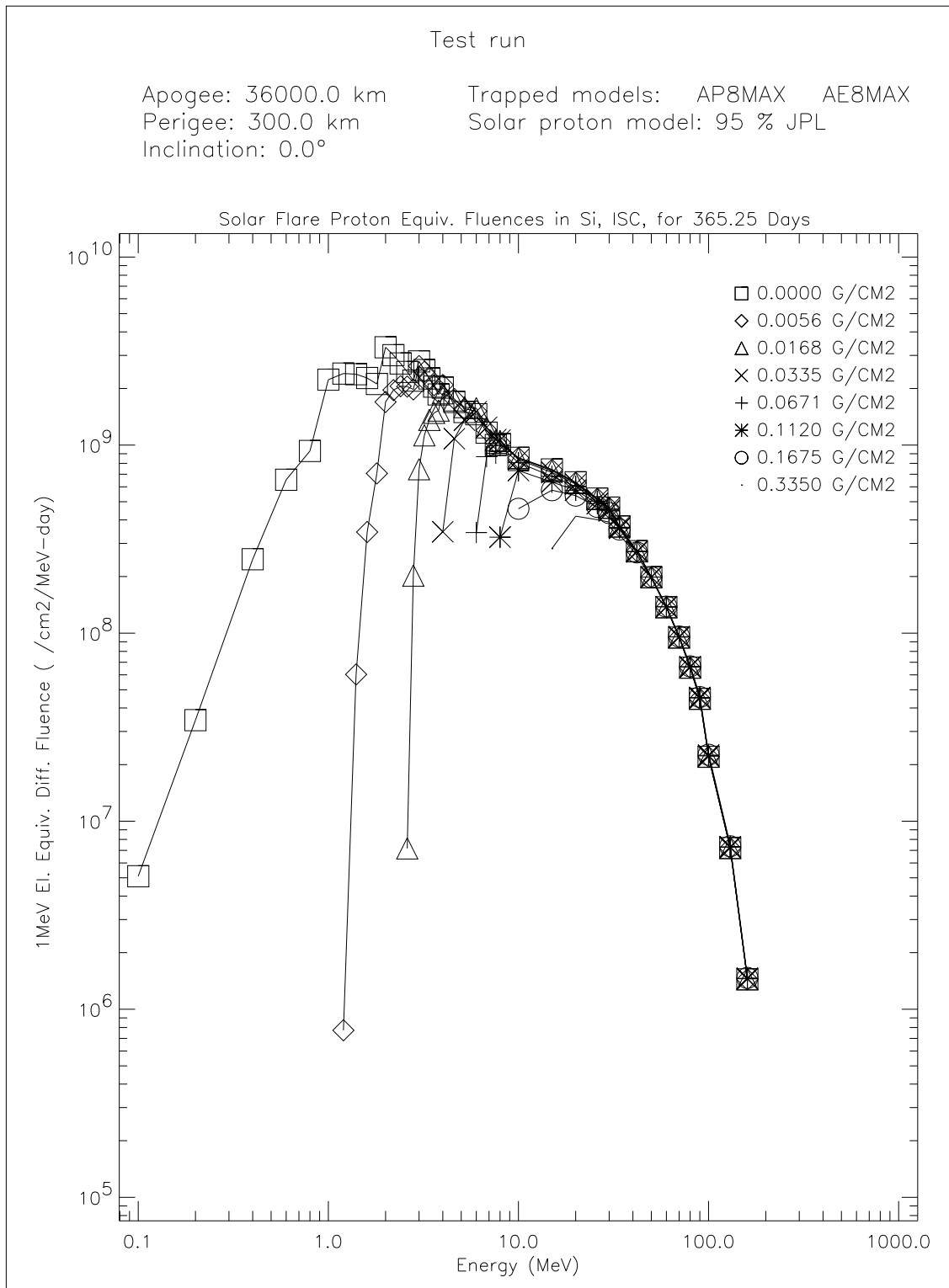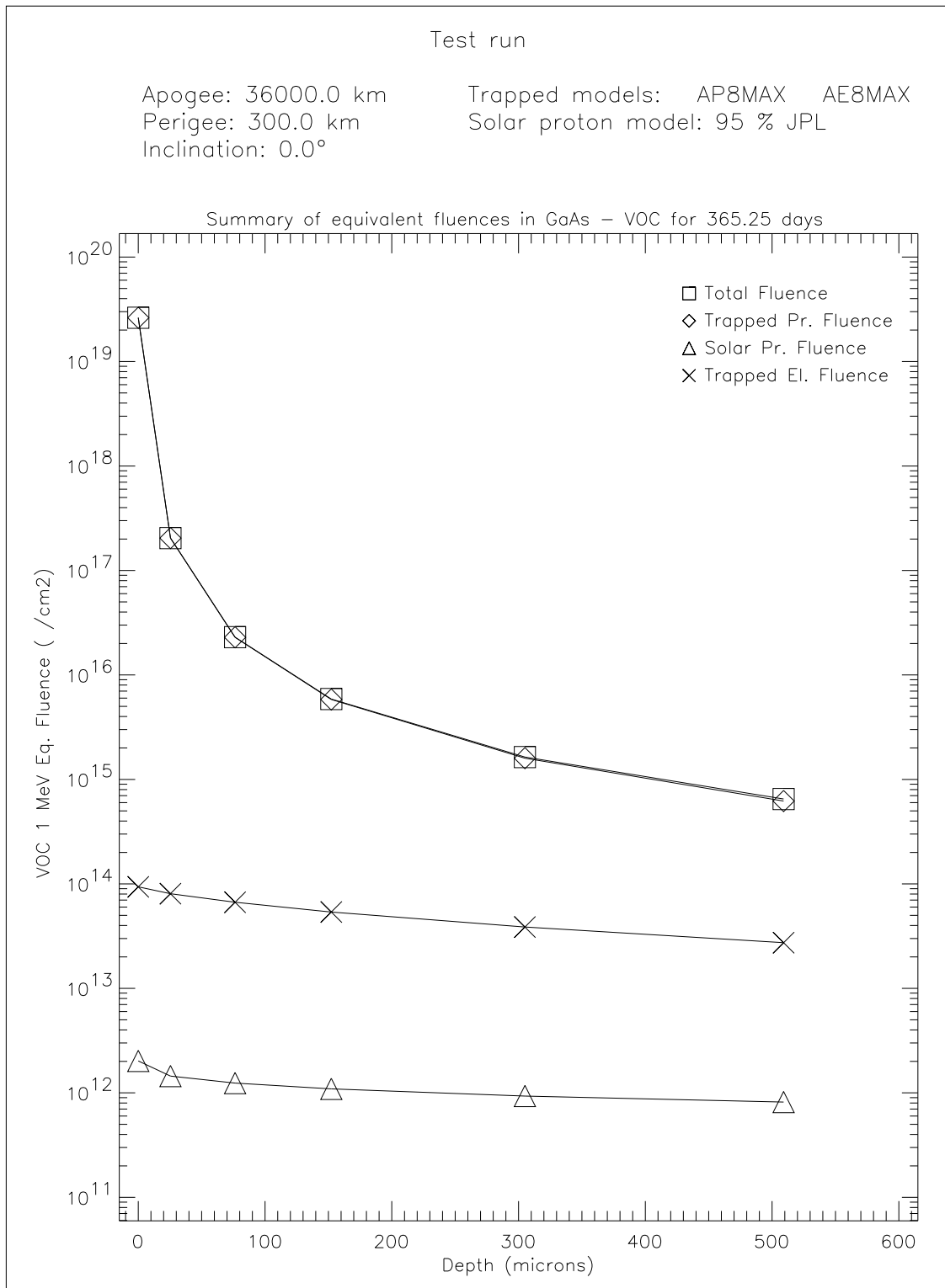**Figure 4.14.** Trapped proton equivalent fluences (PMAX-VOC) for the sample orbit.
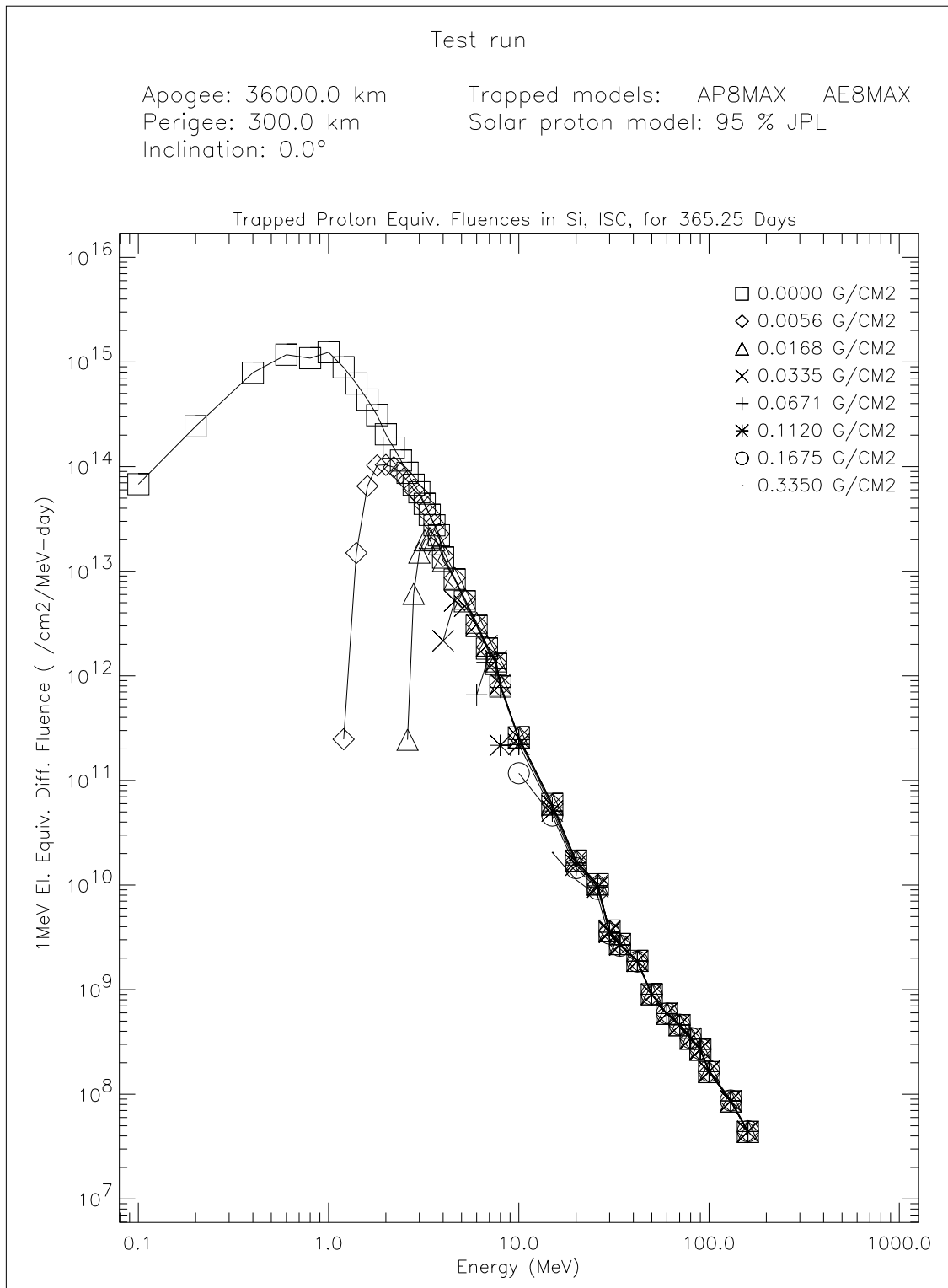
Figure 4.15. Trapped proton equivalent fluences (ISC) for the sample orbit.
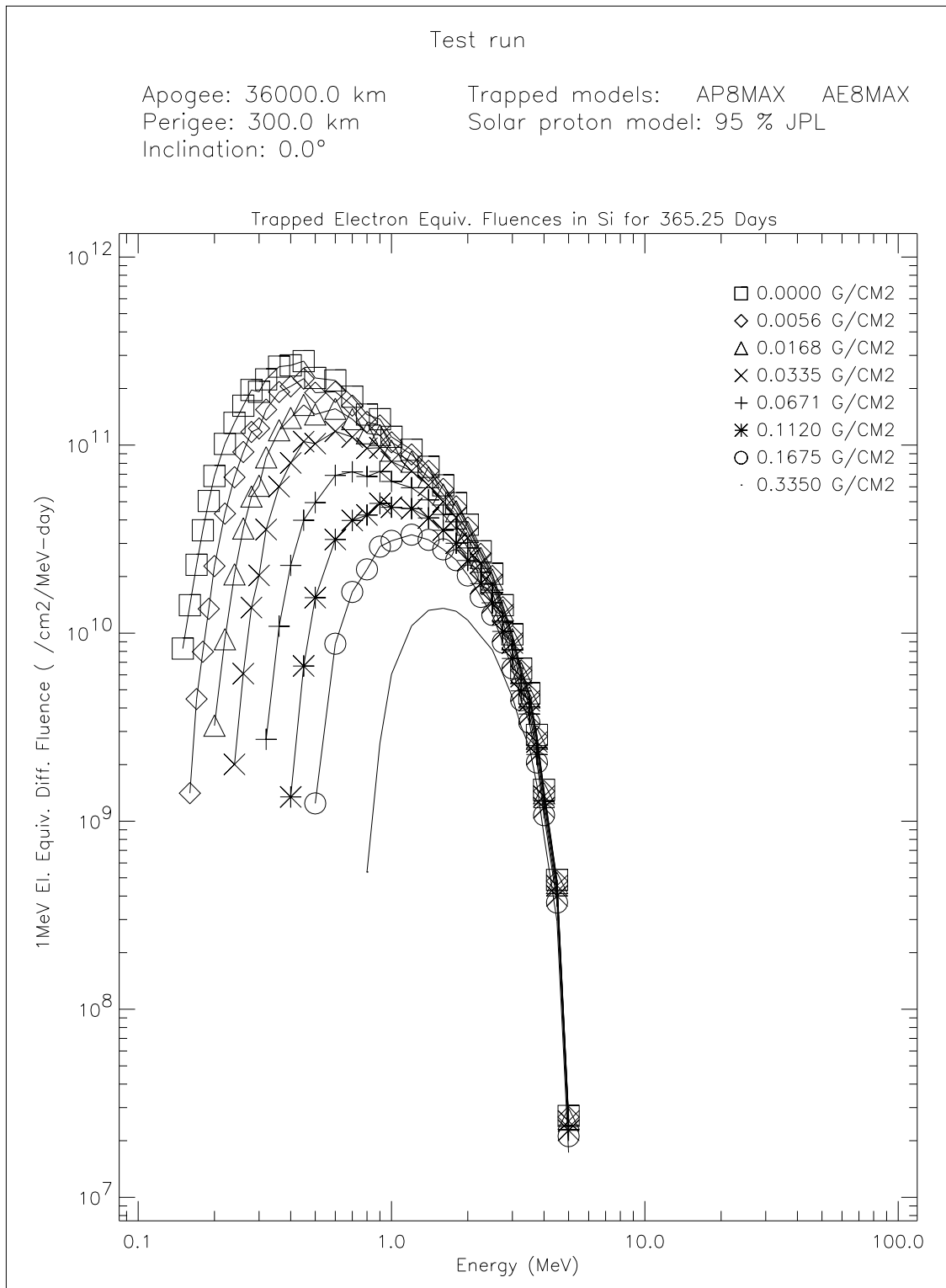
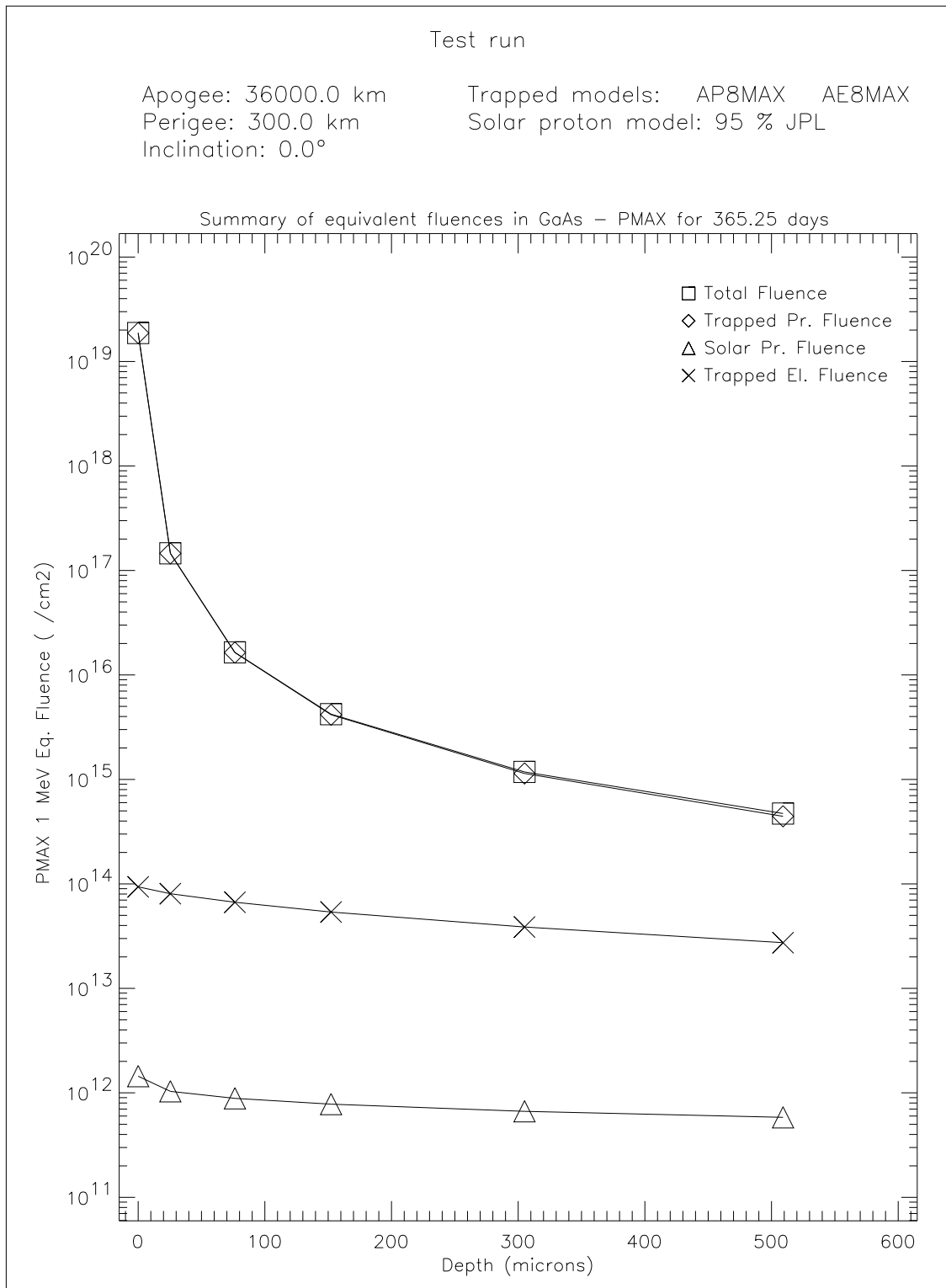**Figure 4.16.** Trapped electron equivalent fluences for the sample orbit.
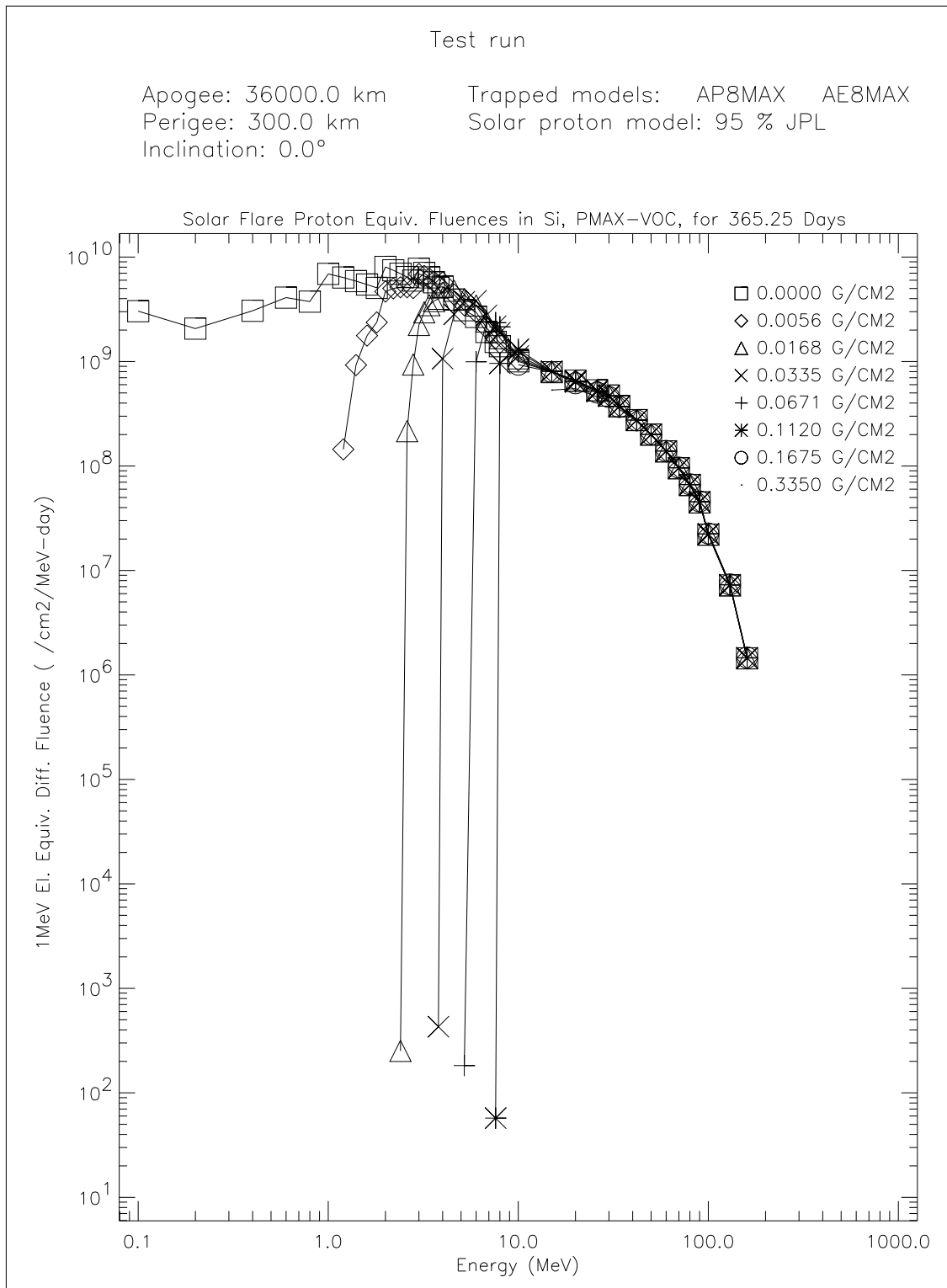
Figure 4.17. Solar flare proton equivalent fluences (PMAX-VOC) for the sample orbit.

Figure 4.18. Solar flare proton equivalent fluences (ISC) for the sample orbit.
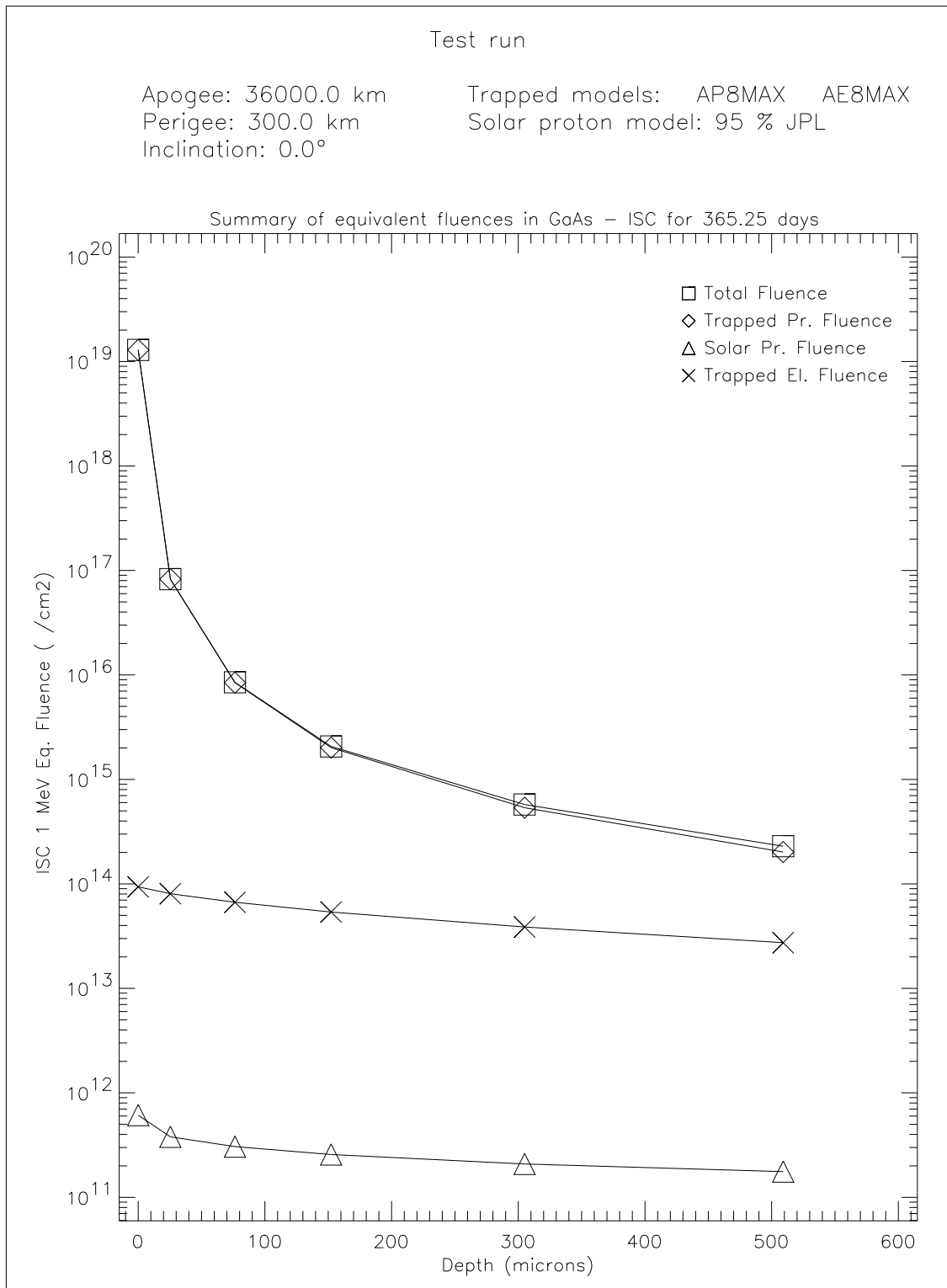
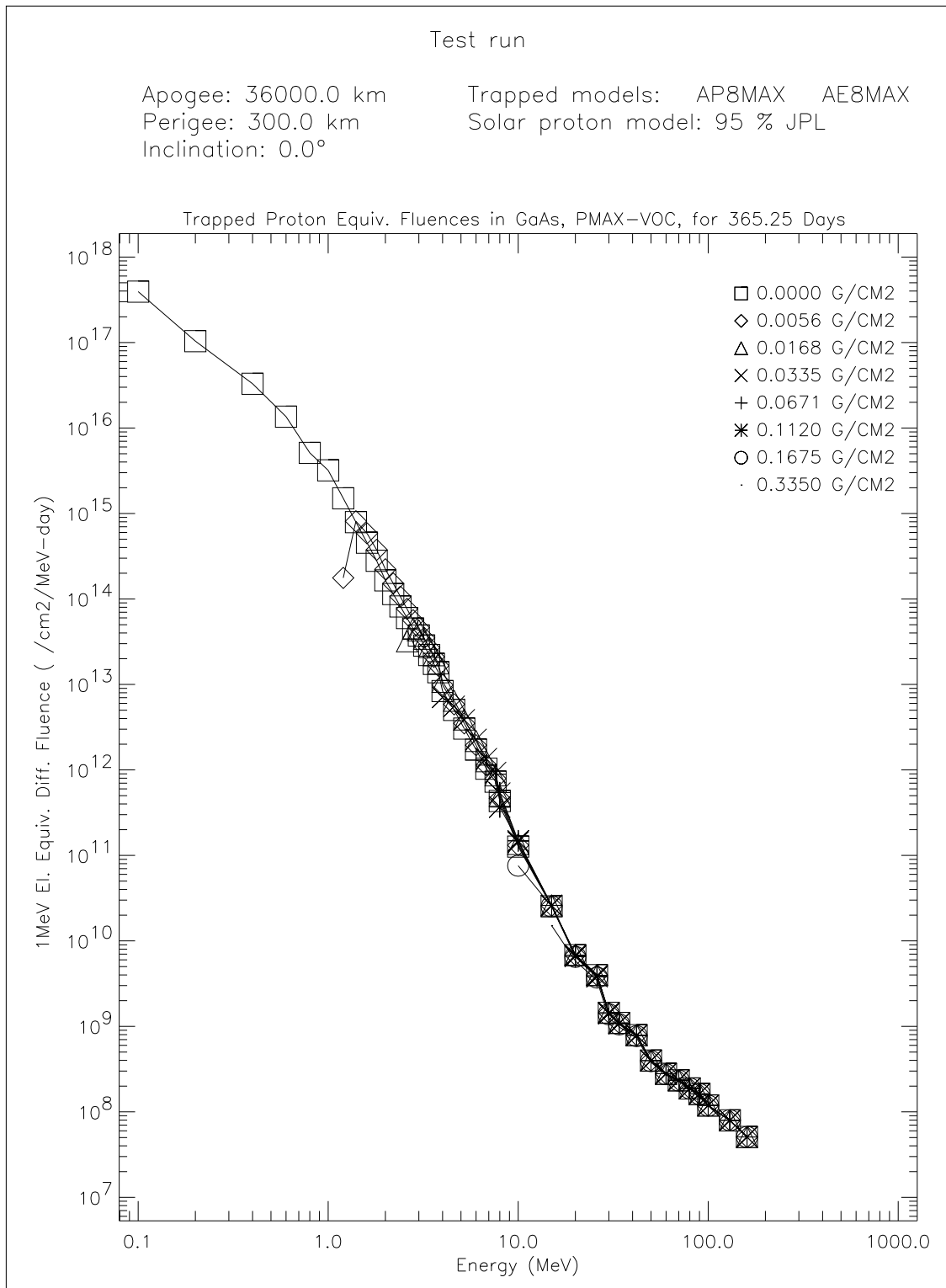**Figure 4.19.** Summary of VOC equivalent fluences for the sample orbit.

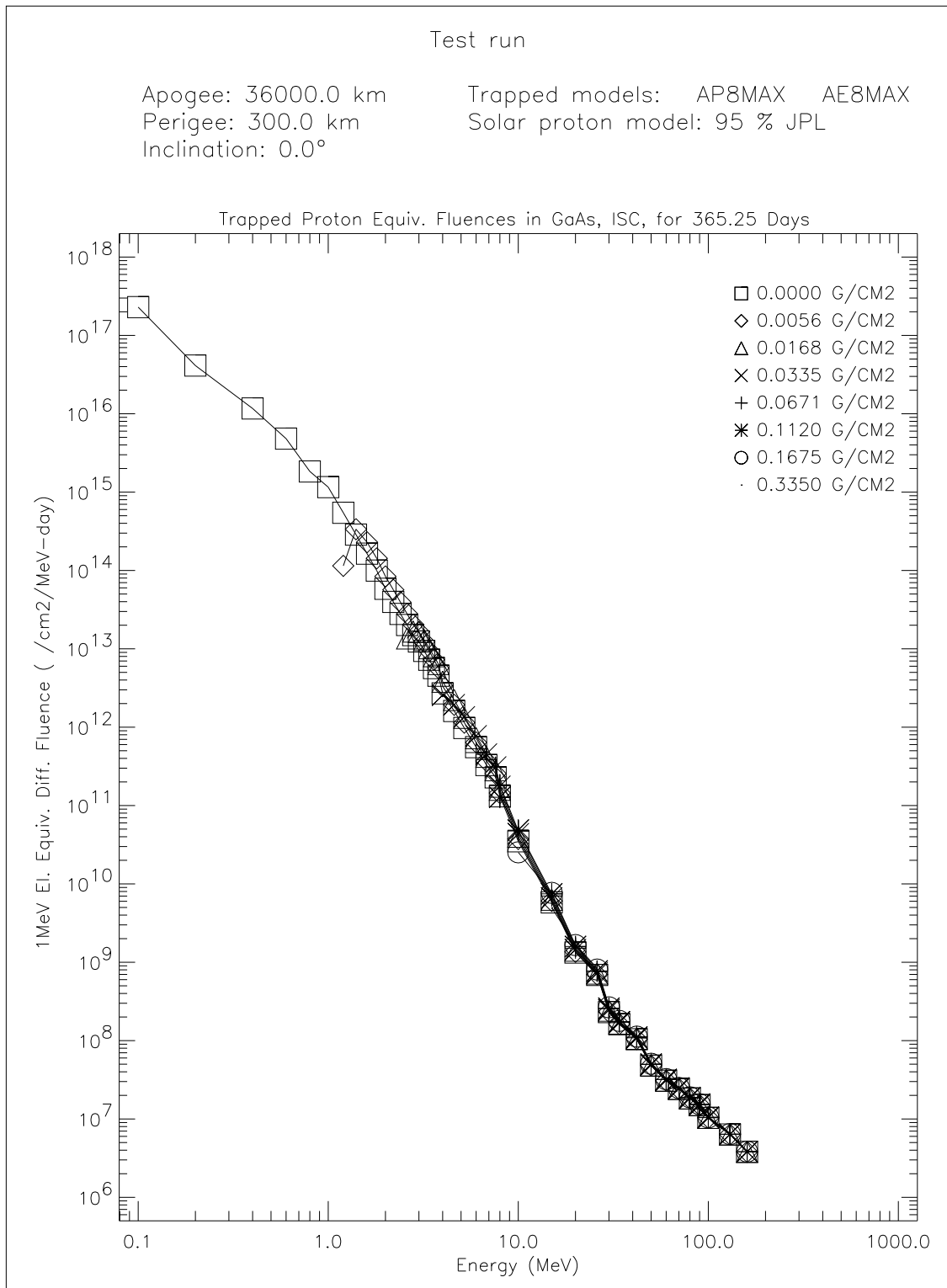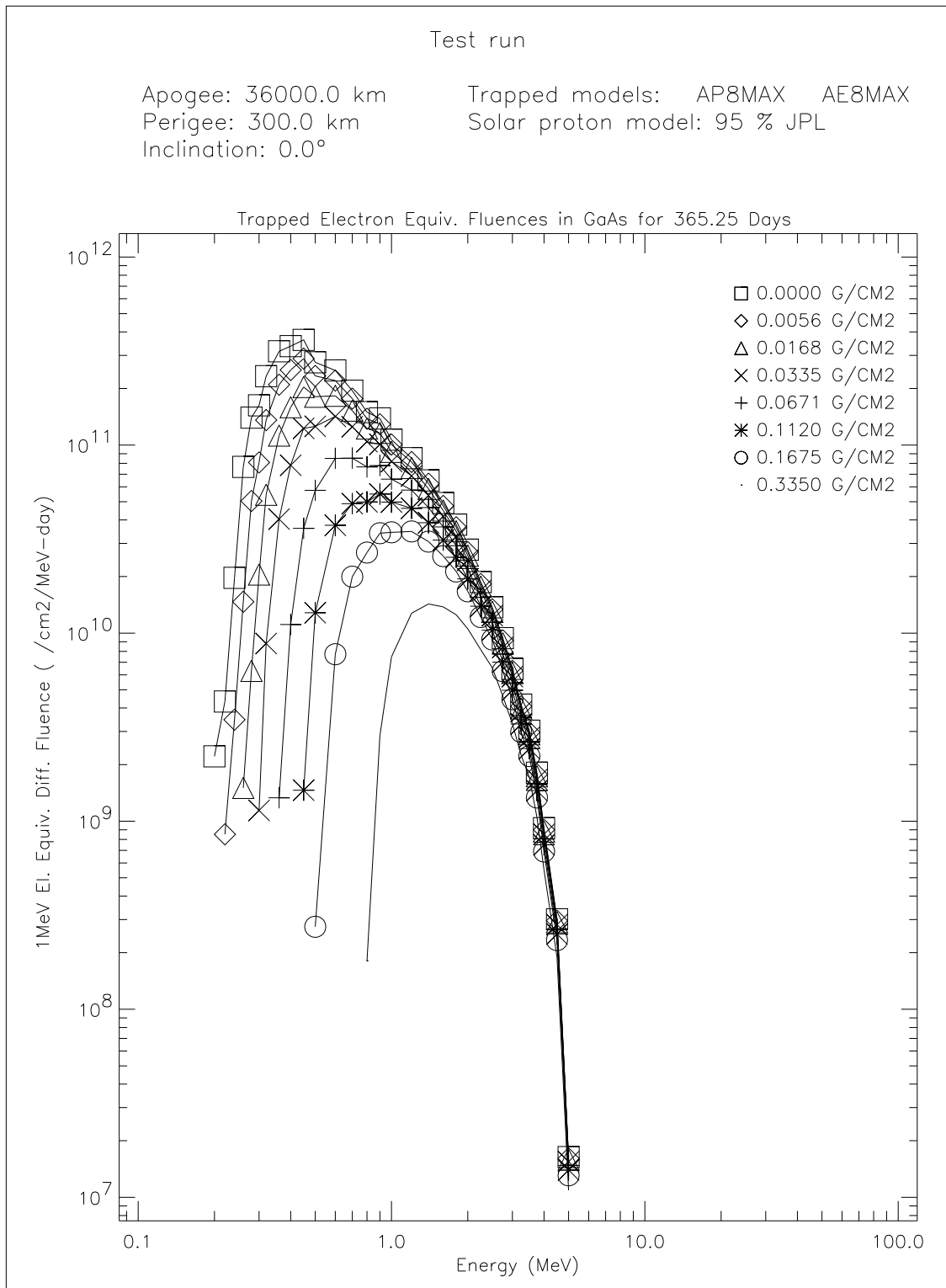**Figure 4.20.** Summary of PMAX equivalent fluences for the sample orbit.

Figure 4.21. Summary of ISC equivalent fluences for the sample orbit.
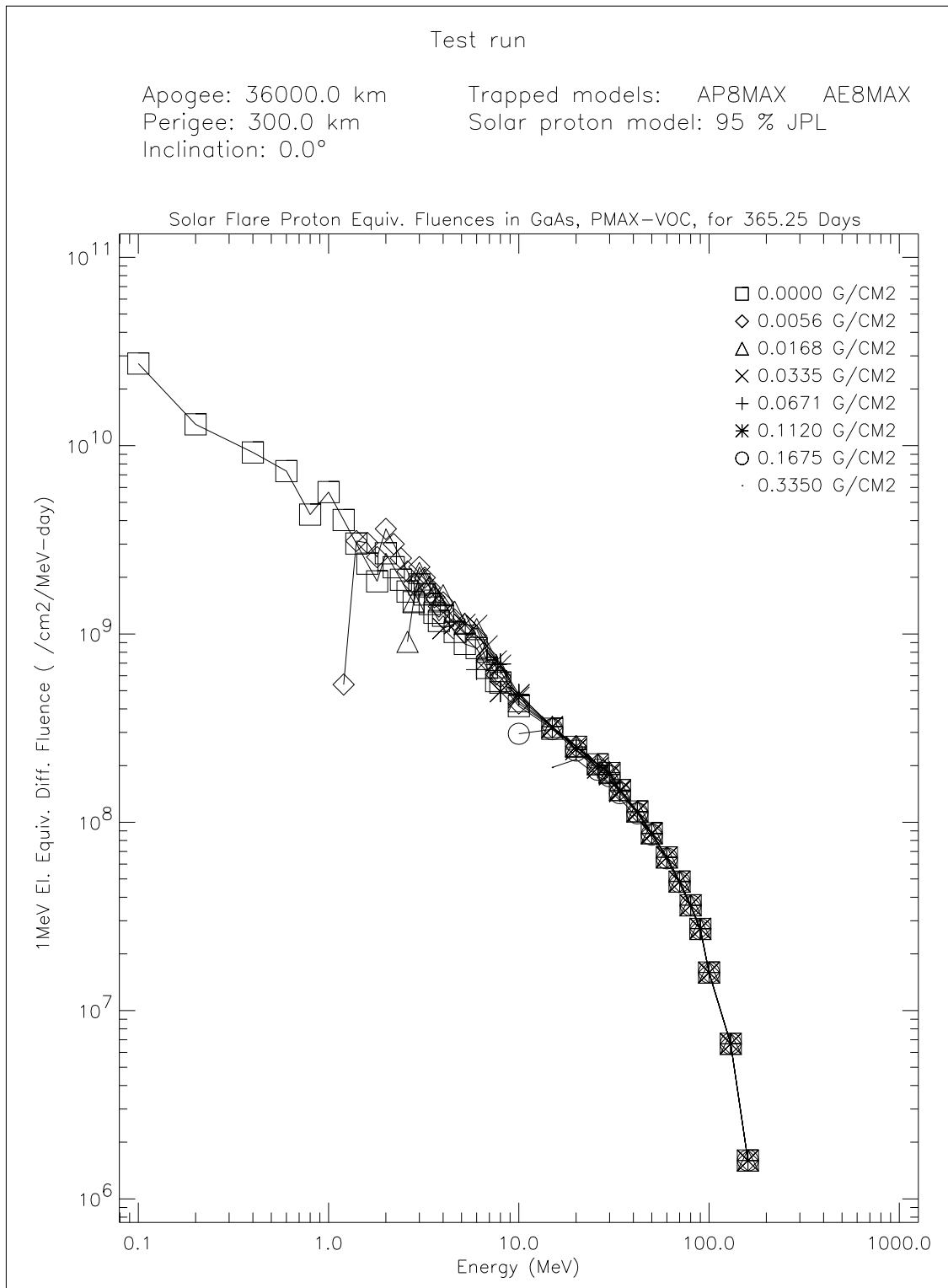
Figure 4.22. Trapped proton equivalent fluences (PMAX-VOC) for the sample orbit.
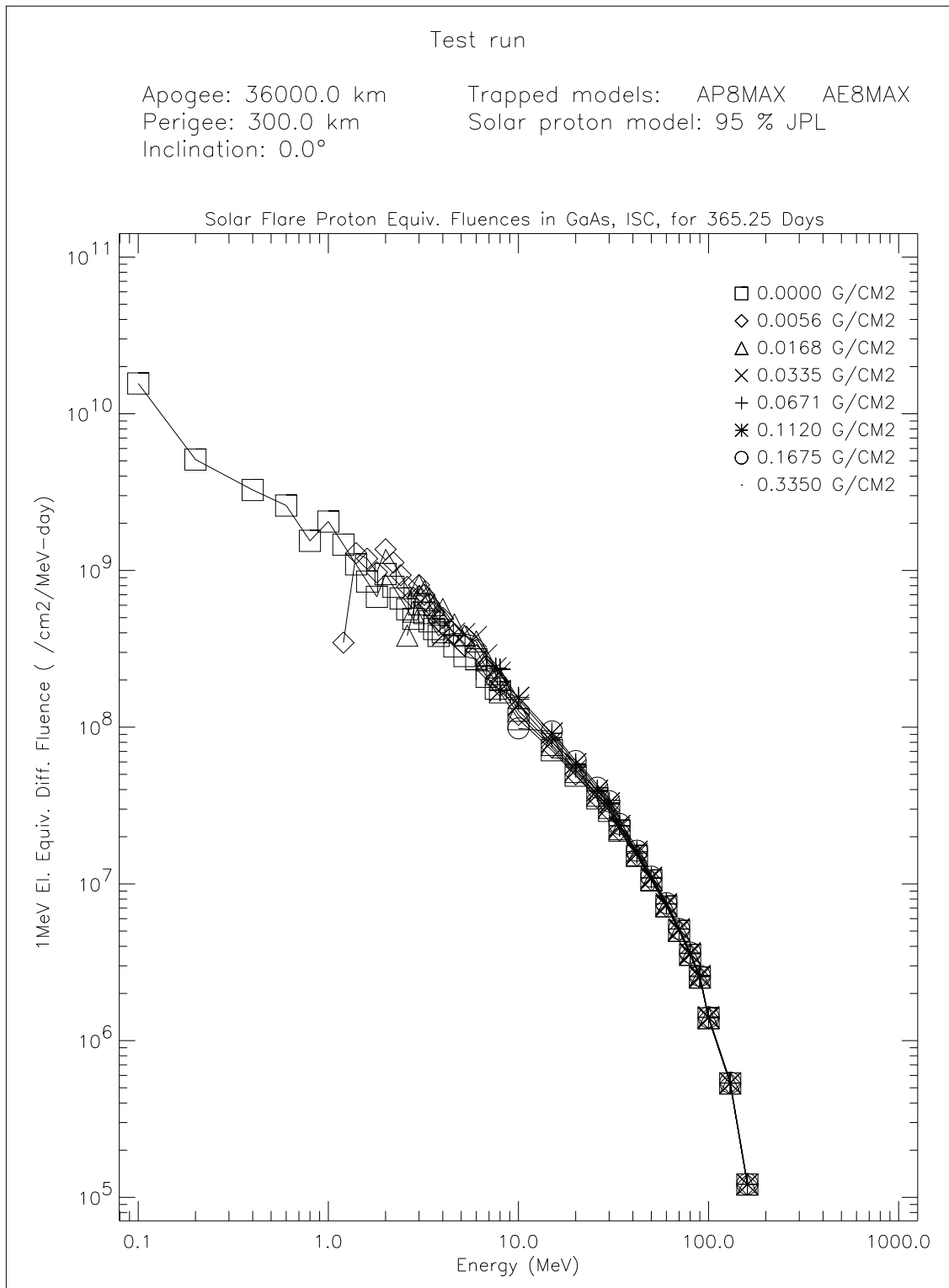
Figure 4.23. Trapped proton equivalent fluences (ISC) for the sample orbit.

Figure 4.24. Trapped electron equivalent fluences for the sample orbit.

Figure 4.25. Solar flare proton equivalent fluences (PMAX-VOC) for the sample orbit.

Figure 4.26. Solar flare proton equivalent fluences (ISC) for the sample orbit.

Appendix A

Listing of namelist file DEFAULT.NML

This appendix contains the listing of the default `NAMELIST` file `DEFAULT.NML` provided in the distribution. PC users should note that a `NAMELIST` has to end with / instead of `&END`.

```
$SAPRE
 TITLE = 'Sample run with default parameter values'
 TYPE = 'GEN'
 OEYEAR = 1995
 OEMON = 1
 OEDAY = 1
 OEHRS = 0
 OEMIN = 0
 OESEC = 0.0D0
 AMJD = -1.0D0
 SEYEAR = -1
 SEMON = -1
 SEDAY = -1
 SEHRS = -1
 SEMIN = -1
 SESEC = -1.0D0
 IAE = 0
 HAPO = 36000.0D0
 HPER = 300.0D0
 RINCL = 0.0001D0
 ALT = 300.0D0
 A = 6671.0D0
 E = 0.0D0
 IG50 = 1
 RAAN = 0.0D0
 ARGPER = 0.0D0
 TRANO = 0.0D0
 LONGAPO = 999.9D0
 LONGPER = 999.9D0
```

```
 OELONG = -9999.999D0
 SOLTIM = -9999.999D0
 ORBITS = 1.0D0
 STEP = 2.0D0
 EPDUR = 0.0D0
 DT1 = 120.0D0
 DT2 = 240.0D0
 DT3 = 1200.0D0
 DH2 = 1.0D4
 DH3 = 1.0D6
 WIBAIR = 0.0D0
 WIBSPR = 0.0D0
 NCIRA = 5
 SUN = 1
 MOON = 1
 KZONAL = 6
 KTESS = 3
 IPRINT = 1
 IORBEL = 0
 ORBNUM = 1.0D0
 RUN = 1
 IBUG = 0
 PB = 22.164D0
 PM = 15.0D0
 XB = 15.25D0
 XM = 10.0D0
 IATTI = 0
 PA = 90.0D0
$END
$BLXTRA
 BLTIME = 1995.0D0
 MODEL = 0
 OUTER = 0
 VALUE_KP = 0.0D0
 DEN = 25.0D0
 VEL = 300.0D0
 DST = -30.0D0
 IPRINT = 60
 IASCII = 0
 CUTOFF = 1.0D6
$END
$TREP
 SOLACT = 'MAX'
 TREMOD = 1
 IPRINT = 60
 INTERF = 1
```

```
 ISPEC = 0
 ISPSHT = 0
 PERIOD = 0.0
 ISIG = 0
 ILTV = 0
 FLUXTH = 1.0, 1.0
 ETHP = 1.0, 10.0
 ETHE = 0.1, 1.0
 SAAROT = 1
 OMNI = 1
 NENERP = 30
 NENERE = 30
 NENERS = 30
 PROEN =    0.1,    0.5,    1.,   2.,   3.,   4.,    5.,    6.,    8.,
           10.0,   12.0,   15.,  17.,  20.,  25.,   30.,   35.,   40.,
           45.0,   50.0,   60.,  70.,  80.,  90.,  100.,  125.,  150.,
          175.0,  200.0,  300.
 ELEEN = 0.04, 0.1, 0.20, 0.3, 0.40, 0.5, 0.60, 0.7, 0.80, 1.0,
          1.25, 1.5, 1.75, 2.0, 2.25, 2.5, 2.75, 3.0, 3.25, 3.5,
          3.75, 4.0, 4.25, 4.5, 4.75, 5.0, 5.50, 6.0, 6.50, 7.0
 ENERFL =    0.1,    0.5,    1.,   2.,   3.,   4.,    5.,    6.,    8.,
            10.0,   12.0,   15.,  17.,  20.,  25.,   30.,   35.,   40.,
            45.0,   50.0,   60.,  70.,  80.,  90.,  100.,  120.,  140.,
           160.0,  180.0,  200.
 NAL = 0
 NOR = 0
 IFORM = 0
 RAU = 1.0
 TFLARE = 1.0
 FLSTAR = -1.0
 FLPROB = 95.0
 FLAMOD = 0
 IFLATT = 0
$END
$TREPAVE
 WEIGHTS = 200 * 1.0
$END
$ANISO
 JANIS = 1
 NDIR = 180
 XOMEGA = 400 * 0.0D0
 XALPH = 400 * 15.0D0
 XBETA = 400 * 24.0D0
 IFULL = 0
$SHIELDOSE
 TINTER = 365.25
```

```
 IDET = 3
 IMAX = 25
 IUNT = 3
 ISHLD = 4
 Z =  0.05,  0.1,  0.2,  0.3,  0.4,  0.5,  0.6,  0.8,  1.0,
       1.5,  2.0,  2.5,  3.0,  4.0,  5.0,  6.0,  7.0,  8.0,  9.0,
      10.0, 12.0, 14.0, 16.0, 18.0, 20.0, 22.0, 24.0, 26.0, 28.0,
      30.0, 32.0, 34.0, 36.0, 38.0, 40.0, 44.0, 48.0, 52.0, 56.0,
      60.0, 64.0, 68.0, 72.0, 76.0, 80.0, 84.0, 88.0, 92.0, 96.0,
     100.0
 IPRINT = 0
 IPLOT = 1
$END
$EQFRUX
 TINTER = 365.25D0
 BSF = 1.0D0
 PEDRI = 3000.0D0
 PEDRV = 3000.0D0
 IPRINT = 0
 IPLOT  = 1
$END
$EQFRUXGA
 TINTER = 365.25D0
 BSF = 1.0D0
 PEDRI = 400.0D0
 PEDRV = 1400.0D0
 PEDRP = 1000.0D0
 IPRINT = 0
 IPLOT  = 1
$END
```

Appendix B

Format of the common interface file

The program flow in `UNIRAD` was modified so that `SAPRE`, `BLXTRA` and `TREP` make use of one common data file for input and output. We have further modified the structure of this file in order to incorporate the use of external magnetic field models and the implementation of pitch angle dependence.

The new structure of the interface file is given in Table B.1. The file is in `ASCII` format, with fixed record length of 178 bytes, in direct access. The first seven records contain header information on orbit parameters, magnetic field models, and energy thresholds for flux calculations. The next records are data records, one record per orbital point. These records contain the modified Julian day from 1950.0, position vector, α, B at the orbital point, L_{m}, [the (B, L_{m}) pair is repeated so that the two field models used by `TREP` can be accomodated] and the integral proton and electron fluxes for the two threshold energies specified in the header. The last record contains negative values and signifies the end of the orbit. This whole structure may be repeated for another orbit.

Table B.1. Record structure of the common interface file. N indicates the number of points on the orbit. The whole structure is repeated for additional orbits.

Record	Format	Description
Header Records		
1	1X,A80	Project descriptor
2	1X,5I6,F8.4, 2E14.6,F10.5	Year, month, day, hour, minute, second of trajectory epoch, height (km) of apogee and perigee, orbit inclination (deg)
3	2(1X,A32),2I6	Description labels and model numbers for two internal magnetic field models
4	2(1X,A32),2I6	Description labels and model numbers for two external magnetic field models
5	1X,2F10.4,F9.6, 4E14.6,2(2X,A8), 2I3	Epochs for two internal magnetic field models, geomagnetic moment, four energy thresholds (two each for protons and electrons), trapped particle model headers, solar flare model number, attitude model number (IATTI)
6	1X,A80	Format string for data records
7	A177	Table headers
Data Records		
1–N	given in header record 6	Modified Julian day (days) from 1950.0, eastern longitude (deg), latitude (deg), height above the Earth's surface (km), pitch angle (deg), magnetic field intensity at the orbital point (Gauss) and $L_{\mathrm{m}}(R_{\mathrm{E}})$ for both magnetic field models, integral proton fluxes above two energy thresholds, integral electron fluxes above two energy thresholds
$N+1$	given in header record 6	13 * -1.0

References

Armstrong, T.W., Colborn, B.L., Watts, J.W.: 1990, *Characteristics of Trapped Proton Anisotropy at Space Station Freedom Altitudes*, Science Applications International Corporation Report SAIC-90/1474

Badhwar, G.D., Konradi, A.: 1990, *Conversion of Omnidirectional Proton Fluxes Into a Pitch Angle Distribution*, J. Spacecraft Rockets **27**, 350

Cain, J.C., Hendricks, S.J., Langel, R.A.: 1967, *A Proposed Model for the International Geomagnetic Reference Field-1965*, J. Geomag. Geoelectr. **19**, 335–355

Colborn, B.L., Watts, J.W., Armstrong, T.W.: 1990, *Data Base Description and Retrieval Program for the Trapped Proton Vector Flux Data Bases VF1MAX and VF1MIN*, Science Applications International Corporation Report SAIC-90/1475

Daly, E.J., Evans, H.D.R.: 1993, *Problems in Radiation Environment Models at Low Altitudes*, Memorandum ESA/ESTEC/WMA/93-067/ED

Feynman, J., Gabriel, S.: 1990, *A new model for calculation and prediction of solar proton fluences*, AIAA 90-0292, JPL, Pasadena

Fraser-Smith, A.C.: 1987, *Centered and Eccentric Geomagnetic Dipoles and Their Poles, 1600–1985*, Rev. Geophys. **25**, 1–16

Heckman, H.H., Nakano, G.H.: 1969, *Low-Altitude Trapped Protons during Solar Minimum Period*, J. Geophys. Res. **74**, 3575–3590

Heynderickx, D., Lemaire, J.: 1993, *Improvements to Trapped Radiation Software*, Technical Note 1 for ESA/ESTEC/WMA TRP Contract No. 9828/92/NL/FM

Heynderickx, D., Lemaire, J., Daly, E.J.: 1996, *Historical Review of the Different Procedures Used to Compute the L-Parameter*, Nucl. Tracks Radiat. Meas. **26**, 325–331

Jensen, D.C., Cain, J.C.: 1962, *An Interim Geomagnetic Field*, J. Geophys. Res. **67**, 3568

Johnson, D.L., Smith, R.E.: 1985, *The MSFC/J70 Orbital Atmosphere Model and the Data Bases for the MSFC Solar Activity Prediction Technique*, NASA TM-86522

King, J.H.: 1974, *Solar Proton Fluences for 1977–1983 Space Missions*, J. Spacecraft and Rockets **11**, 401

Kruglanski, M., Lemaire, J.: 1996, *Trapped Proton Anisotropy at Low Altitudes*, Technical Note 6 for ESA/ESTEC/WMA TRP Contract No. 10725/94/JG(SC)

Lenchek, A.M., Singer, S.F.: 1962, *Effects of Finite Gyroradii of Geomagnetically Trapped Protons*, J. Geophys. Res. **67**, 4073–4075

McIlwain, C.E.: 1961, *Coordinates for Mapping the Distribution of Magnetically Trapped Particles*, J. Geophys. Res. **66**, 3681–3691

Mead, G.D., Fairfield, D.H.: 1975, *A Quantitative Magnetospheric Model Derived From Spacecraft Magnetometer Data*, J. Geophys. Res. **80**, 523–534

Olson, W.P., Pfitzer, K.A.: 1977, *Magnetospheric Magnetic Field Modeling*, Annual Scientific Report, AFOSR Contract No. F44620-75-C-0033, McDonnell Douglas Astronautics Company, Huntington Beach, CA

Pfitzer, K.A.: 1991, *Improved Models of the Inner and Outer Radiation Belts*, Phillips Laboratory Scientific Report PL-TR-91-2187

Pfitzer, K.A., Olson, W.P., Mogstad, T.: 1988, *A Time Dependent, Source Driven Magnetospheric Magnetic Field Model*, EOS **69**, 426

Sawyer, D.M., Vette, J.I.: 1976, *AP-8 Trapped Proton Environment for Solar Maximum and Solar Minimum*, NSSDC/WDC-A-R&S 76-06

Seltzer, S.M.: 1979, *Electron, Electron-Bremsstrahlung and Proton Depth-Dose Data for Space Shielding Applications*, IEEE Trans. Nucl. Sci. NS-26, 4896

Seltzer, S.M.: 1980, *A Computer Code for Space Radiation Shielding Methods*, NBS Technical Note 116

Singley, G.W., Vette, J.I.: 1972a, *The AE-4 Model of the Outer Radiation Zone Electron Environment*, NSSDC 72-06

Singley, G.W., Vette, J.I.: 1972b, *A Model Environment for Outer Zone Electrons*, NSSDC 72-13

Stassinopoulos, E.G., King, J.H.: 1973, *Empirical Solar Proton Model for Orbiting Spacecraft Applications*, IEEE Trans. on Aerosp. and Elect. Systems AES-10, 442

Stern, D.P., Tsyganenko, N.A.: 1992, *Uses and Limitations of the Tsyganenko Magnetic Field Models*, EOS **73**, 489

Tada, H.Y., Carter, J.R., Anspaugh, B.E., Downing, R.G.: 1982, *Solar Cell Radiation Handbook*, 3rd edition, NASA JPL 82-69

Tsyganenko, N.A.: 1987, *Global Quantitative Models of the Geomagnetic Field in the Cislunar Magnetosphere for Different Disturbance Levels*, Planet. Space Sci. **35**, 1347–1358

Tsyganenko, N.A.: 1989, *A Magnetospheric Magnetic Field Model with a Warped Tail Current Sheet*, Planet. Space Sci. **37**, 5–20

Vette, J.I.: 1991a, *The AE-8 Trapped Electron Environment*, NSSDC/WDC-A-R&S 91-24

Vette, J.I.: 1991b, *The NASA/National Space Science Data Center Trapped Radiation Environment Model Program*, NSSDC/WDC-A-R&S 91-29

Watts J.W., Parnell, T.A., Heckman, H.H.: 1989, *Approximate Angular Distribution and Spectra for Geomagnetically Trapped Protons in Low-Earth Orbit*, Conf. on High-Energy Radiation in Background Space, A.C. Rester Jr. and J.I. Trombka (Eds.), Santibel Island, FL, 1987, Am. Inst. Phys. Conf. Proc., New York, 75–85