N92-15894

1991

NASA/ASEE SUMMER FACULTY FELLOWSHIP PROGRAM

MARSHALL SPACE FLIGHT CENTER
THE UNIVERSITY OF ALABAMA IN HUNTSVILLE

# A HIGH-FIDELITY N-BODY EPHEMERIS GENERATOR FOR SATELLITES IN EARTH ORBIT


| | |
|---|---|
| Prepared By: | David R. Simmons, Ph.D. |
| Academic Rank: | Professor |
| Institution: | Louisiana College<br>Department of Mathematics<br>and Computer Science |
| NASA/MSFC:<br>Office: | Systems Analysis and<br>Integration Laboratory |
| Division: | Mission Analysis |
| Branch: | Flight Mechanics |
| MSFC Colleague: | Larry Mullins, Ph.D. |
| Contract No.: | NGT-01-008-021<br>The University Of Alabama<br>In Huntsville |

## Description of the Project

The Flight Mechanics Branch is currently using a program for mission planning called the Analytic Satellite Ephemeris Program (ASEP). This program, written by Jim McCarter, produces projected data for orbits that remain fairly close to the Earth; ASEP does not take into account lunar and solar perturbations. These perturbations are accounted for in another program called GRAVE, written several years ago by Roger Burrows. This project is a revision of GRAVE which incorporates more flexible means of input for initial data, provides additional kinds of output information, and makes use of structured programming techniques to make the program more understandable and reliable.

## Work Done during Summer 1990

David Simmons wrote the FORTRAN program ORBIT during the first summer; the SAIL1 VAX system was used to develop the program. In keeping with structured programming concepts, the program is divided into numerous sub-programs, each with a well-defined task to perform. The text of the source code for one sub-program can usually be printed on a page or less. Most of the variable names are whole words or short phrases which clearly identify the nature of the variable and its role in the program.

ORBIT is divided into three major phases: initialization, integration, and output. During the linking process, the block data subprogram, Load_Common, gives initial values to the key variables in COMMON. Later during the initialization phase, the Get_Parameter subroutine uses tree-structured menus to give users an opportunity to change the starting and ending times, output defaults and state vectors. Get_Parameter can change any of the three forms of state vector ( cartesian, spherical-polar, and osculating orbital elements ) that are used in the program; the other forms are always re-calculated to conform to the new one. Get_Parameter also provides for the selection of the kind of output to be provided.

The integration phase of the program calculates new values for the elapsed time and the cartesian state vector describing the motion of the satellite. This section of the program follows the GRAVE program very closely. The Encke method is used; subroutine COAST calculates a position along the osculating ellipse from the current position; this position is used by the subroutine DEQG for the calculation of both gravitational and atmospheric forces. DEQG is called by RKG, a general routine for solving first-order differential equations; RKG uses Fehlberg's 13-step version of the Runge-Kutta method. RKG is used in ORBIT with no change from its previous form. COAST has a very tangled structure; Simmons found it necessary to split it into subroutines based on its syntax rather than on its meaning. On the other hand, DEQG has been split into sub-programs in a natural and well-structured way.

ORBIT produces a complete set of output data before the beginning of the integration, and after the end. The user of the program can select an option to generate time and state-vector information on each step of the integration. The integration phase of the program uses the Cartesian form of the state vector, which is assumed to be relative to an inertial reference system. A spherical-polar state vector and a set of osculating elements are calculated for the output phase by a set of routines organized through a master routine called UpDate_Common. These subroutines also adjust various lunar, solar, and time-related variables that are maintained in COMMON. A set of routines controlled by a subroutine called Report_State then display those values and calculate other values which are also displayed. Another user option is to have the displayed values stored in a file.

ORBIT makes either direct or indirect use of about a score of special-purpose routines already available in the MSFC computer systems, along with modified versions of the DEQG and COAST routines from the GRAVE program. It would not have been possible to complete this project in ten weeks if all these routines had not been available. The use of these routines should also make it easier for MSFC personnel who are already familiar with them to understand this program.

Work Done during Summer 1991

The CRRES satellite was launched in July 1990; its orbit has an eccentricity of about 0.71 which made it a very suitable choice for testing the ORBIT program against real data. By summer 1991 there was data available from NORAD for 313 days. Some problems became evident when the program output was compared with this data.

On each output cycle a subroutine calculated a value for the radius of the earth in kilometers; this was stored in a variable named EarthRadius. The same variable name was used in COMMON for the radius of the earth in meters; one of the integration routines used this radius to compute the perturbations due to the oblateness of the earth. Since the radius was too small by a factor of a thousand, the oblateness effects were reduced to insignificance.

The subroutine used to calculate atmospheric effects had one formal argument, the vector acceleration to be modified. The routine which called it, however, used two actual arguments, the cartesian state vector and the acceleration. The result was that the cartesian state was very slightly modified and the vector acceleration was not modified at all; the orbit did not decay as it should.

The 1990 version of the program had an option to print (or save in a file) basic state vector information after every integration step. This was usually too often, so it was modified to produce output on the next integration step after a specified length of time had passed. Since the step size is not likely to be a rational fraction of the orbital period, and since the RKG routine changes the time increment as the integration proceeds, the orbital position at which output occurred had a slight and apparently random variation. This effect was more irritating than serious, but it was eliminated by restructuring the main program. The integration process inherited from GRAVE.FOR adjusts the step size at the end to reach the specified final time. This process is now encapsulated in a subroutine which has a goal time as its only argument. This subroutine is called in a loop in which the goal time is successively set to desired output times.

Since the orbit is not a simple two-body orbit, the values of the osculating elements depend slightly on the position in the orbit at which they are calculated. In particular, there is a relatively sharp increase in the semi-major axis at perigee. If the data are reported at equal time intervals, the mean anomaly drifts around the orbit. Each time the mean anomaly passes through perigee, the output values of the semi-major axis rise and then fall. This effect also appears in the NORAD data, since they are calculated at the ascending node, and the perigee moves slowly around the orbit. Two options to report results at specified times were added to the program in reaction to these problems: at the ascending node, and at a specified value of the mean anomaly.

The program now employs an exponential atmospheric model, which gives good results when compared with the tracking data. The source program has now been incorporated into three files: ORBIT.FOR, ORBIT.INCL, and a file for the block data subprogram, which will vary from mission to mission.

## Conclusion

ORBIT has now been tested against tracking data for the first 313 days of operation of the CRRES satellite. A sample graph is given comparing the semi-major axis calculated by the program with the values supplied by NORAD. When calculated for points at which CRRES passes through the ascending node, the argument of perigee, the right ascension of the ascending node, and the mean anomaly all stay within about a degree of the corresponding values from NORAD; the inclination of the orbital plane is much closer. The program value of the eccentricity is in error by no more than 0.0002.

It is characteristic of computer programmers never to be completely satisfied with their productions; some improvements are possible. However, both Mullins and Simmons are convinced that ORBIT is accurate and that it is ready for operational use.

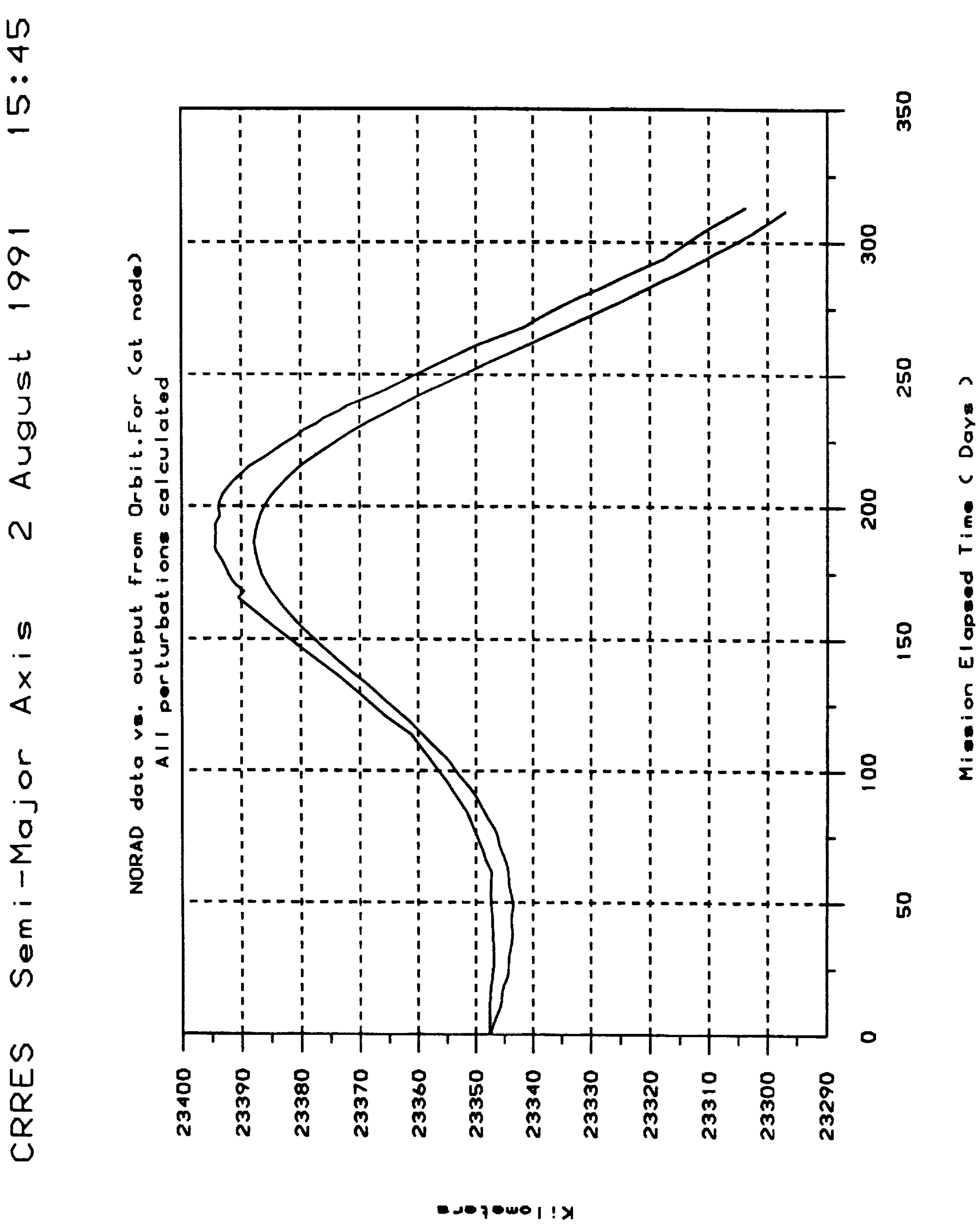
References

Roger Burrows, MSFC: GRAVE.FOR (1985-1988)

J. M. A. Danby: Fundamentals of Celestial Mechanics, 2nd ed.

Larry Mullins, MSFC Course 4181: General Description of Orbits

Accompanying Figure: CRESS Semi-Major Axis


CRRES Semi-Major Axis 2 August 1991 15:45
NORAD data vs. output from Orbit.For (at node)
All perturbations calculated
Kilometers
23400
23390
23380
23370
23360
23350
23340
23330
23320
23310
23300
23290
0
50
100
150
200
250
300
350
Mission Elapsed Time ( Days )