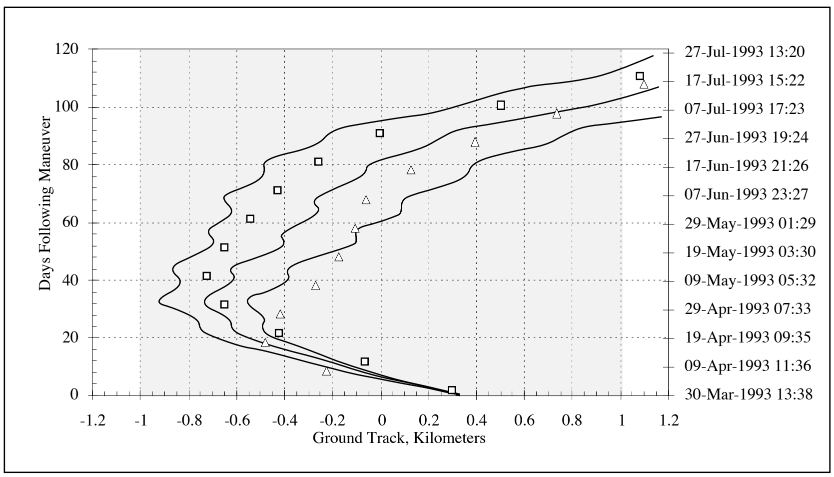
EM 314-563

TOPEX/POSEIDON PROJECT

GTARG - The TOPEX/POSEIDON Ground Track Maintenance Maneuver Targeting Program - User's Reference Manual



Bruce E. Shapiro

June 1, 1993

J

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91109

JPL D-10825

GTARG - The TOPEX/POSEIDON Ground Track Maintenance Maneuver Targeting Program - User's Reference Manual

Abstract

This document provides the basic background, inputs, and procedures necessary to use GTARG, the TOPEX/POSEIDON ground track maintenance maneuver targeting program. Maneuvers ensure that the ground track is kept within a ±1 km. control band of an ≈ 9.9 day exact repeat pattern. GTARG combines orbit prediction and targeting algorithms to design the maneuvers. The analytic mean-element propagation algorithm includes all perturbations that are known to cause significant variations in the satellite ground track. These include earth oblateness, luni-solar gravity, and drag, as well as the thrust due to impulsive maneuvers and unspecified along-track satellite fixed forces. Targeting strategies will either (a) maximize the time between maneuvers (*longitude targeting*) or (b) force control band exit to occur at specified intervals (*time targeting*). A runout mode allows for ground track propagation without targeting. Error models include uncertainties due to orbit determination, maneuver execution, drag unpredictability, and the knowledge of along-track satellite fixed forces.

GTARG - The TOPEX/POSEIDON Ground Track Maintenance Maneuver Targeting Program - User's Reference Manual

Contents

GTARG - The TOPEX/POSEIDON Ground Track Maintenance Maneuver Targeting Program - User's Reference Manual

Figures and Tables

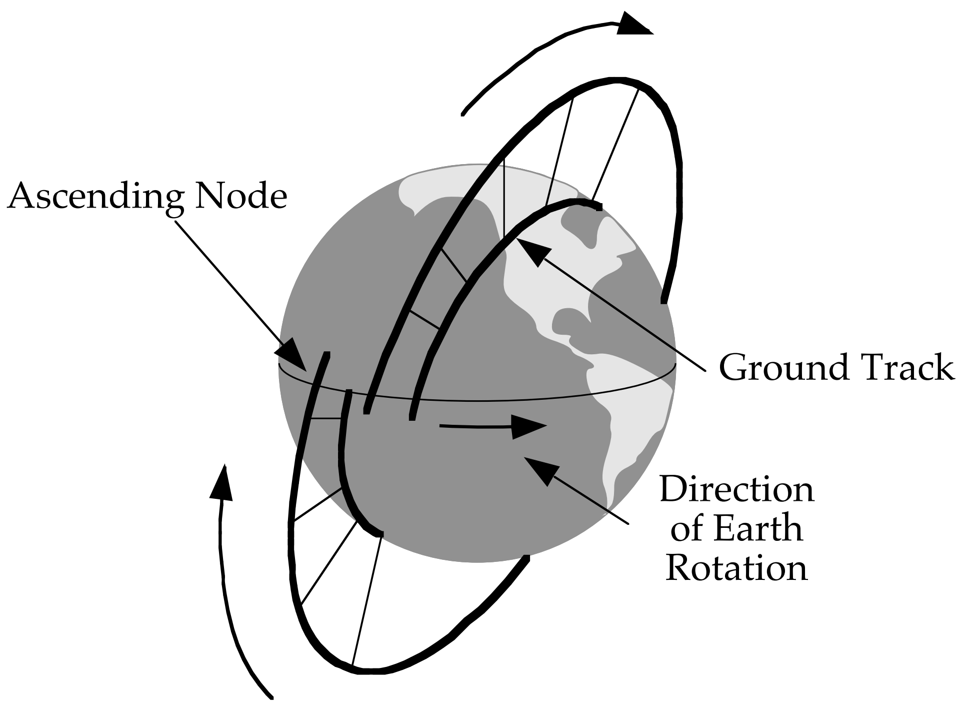
GTARG - The TOPEX/POSEIDON Ground Track Maintenance Maneuver Targeting Program - User's Reference Manual

Introduction

1.1. Background

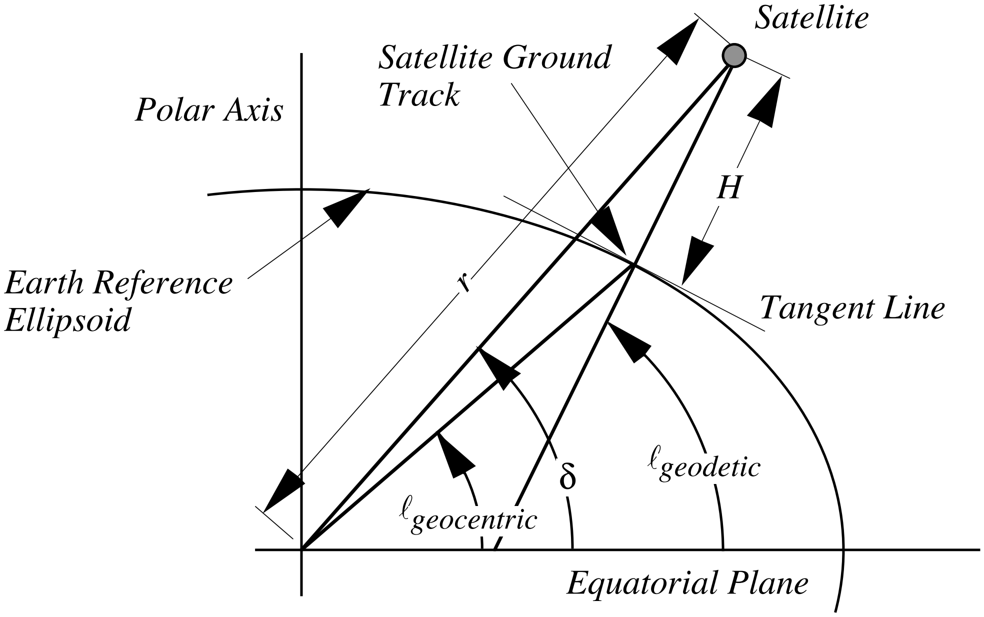
This document describes the inputs and procedures necessary to use GTARG, the TOPEX/POSEIDON ground track maintenance maneuver targeting program. This section summarizes some of the basic concepts needed to understand the use of a ground track targeting program. Section 1.2 provides an overview of the rest of this document. The references provide more details on the concepts of ground track, ground track maintenance, and the implementation of these concepts in GTARG.

Figure 1.1.
Definition of satellite ground track.



The *ground track* of the orbit is defined to be the locus of points traced out on the Earth's surface directly "beneath" the spacecraft orbit, as illustrated in figure 1.1. More precisely, the ground track is found by dropping a perpendicular line from the satellite to the reference ellipsoid representing the surface of the earth. The intersection of this line, which does not pass through the center of the Earth (except at the equator), with the reference ellipsoid, is the ground track (see figure 1.2). Typically, if the ground track repeats itself after some number of orbits, the satellite is said to be in an orbit with an *exact repeat ground track.* The desired ground track is referred to as the *reference ground track*, and the ascending node crossing longitudes of the reference ground track are called the *reference nodes* or *reference grid*.

Figure 1.2.
Definition of satellite ground track.[1] The ground track is found by dropping a perpendicular to the surface (defined by the reference ellipsoid) from the satellite. The geodetic altitude *H*, the declination δ, and the geodetic and geocentric latitudes $\ell_{geodetic}$ and $\ell_{geocentric}$ are shown.

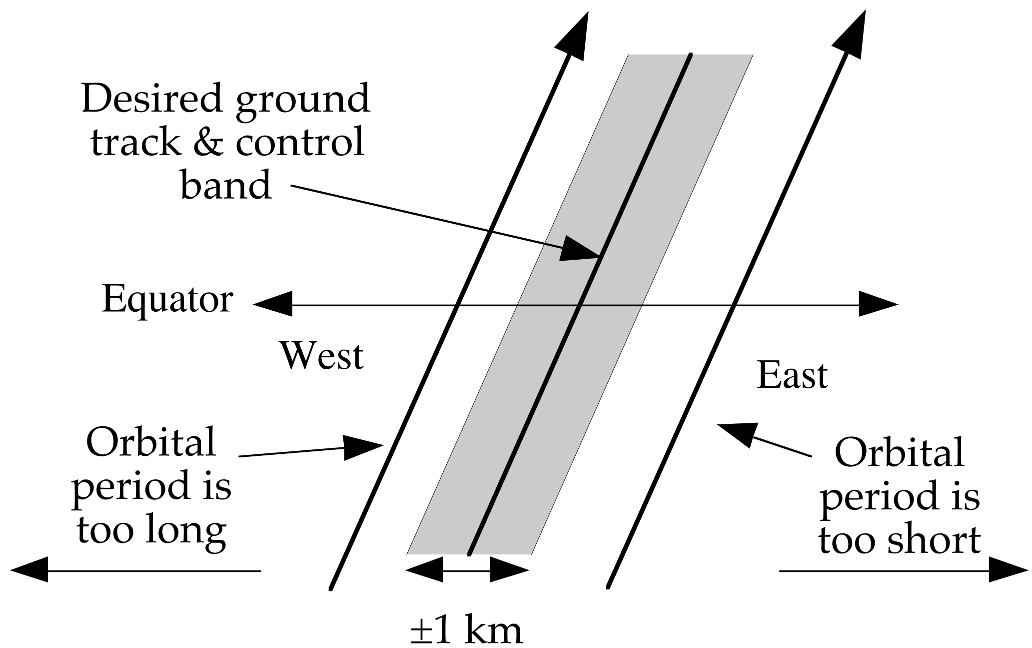


There is a particular combination of the elements a, e, and i which will produce a given exact repeat orbit.[2] Due to the time varying nature of the perturbations on the orbit, however, it is impossible to exactly match the reference ground track for any significant duration of time. Deviations from the reference orbit lead to *ground track drift*. To see this, consider the orbit as it crosses the equator, as shown in figure 1.1. As the Earth rotates from one node crossing to the next the ground track moves westward. If the orbital period is exactly right, successive node crossings match successive reference nodes. If the period is too short, the Earth does not rotate quite far enough, and the true node falls to the east of the reference node. If the period is too long, the earth rotates too far, and the true node falls to the west (figure 1.3). After multiple orbits, the ground track moves further and further to one direction or another, and a ground track drift develops. Ground track maintenance maneuvers must be performed to ensure that the ground track remains within a permissible swath, surrounding the reference orbit. This swath is called the *control band*.

Figure 1.3.
Ground track drift, measured at the equator.

---

[1]Escobal, 1983. The calculation of latitude and longitude from the satellite position vector are given by transformation 3, pp. 398-399.
[2]Cutting, Born, & Frautnick, 1976.



Atmospheric drag is usually the dominant perturbation causing ground track drift. As a result of drag, the semi-major axis decays and hence the period decreases.[1] If the semi-major axis initially exceeds the semi-major axis of the reference orbit, the ground track drifts westward. This drift continues until the orbit matches the reference. As drag continues reducing the semi-major axis, the ground track turns around and begins moving eastward. The result is a somewhat lumpy parabola when plotted as a function of time (see figure 1-4). Various perturbations, such as solar and geomagnetic activity, luni-solar gravity, solar radiation pressure, tides, shadow entrance and exit effects, time-varying drag area, and other satellite fixed and external forces cause deviations from the basic parabola and produce the ripples shown in the curve of figure 1.4.

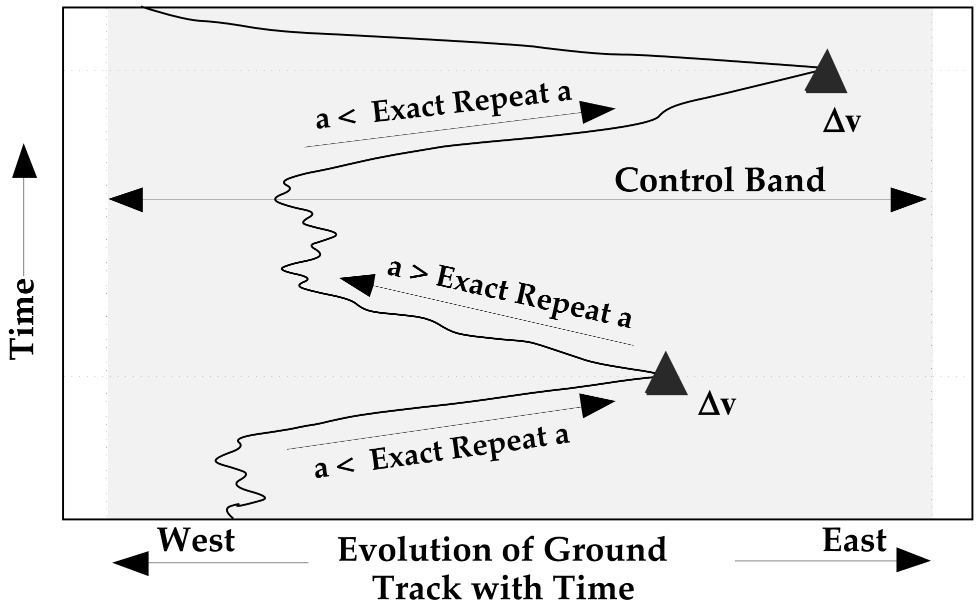
When the ground track approaches edge of the control band, a maneuver is performed to change the direction of the ground track drift. A *ground track maintenance maneuver* will increase (if the ground track is approaching the eastern edge of the control band) or decrease (approaching the western edge) the semi-major axis of the orbit. These maneuvers are along track (parallel or anti-parallel to the direction of motion) and are typically of very small magnitude (Δvs of millimeters per second). The process of determining the magnitude of the maneuver is called *maneuver targeting*. Various strategies can be used for maneuver targeting.[2,3] GTARG combines these targeting strategies with error models to plan maneuvers so that the ground track will remain within the control band for a known period of time with a high degree of confidence. The remaining sections of this document describe how to use GTARG to predict the ground track and perform maintenance maneuver targeting.

Figure 1.4.

---

[1]Bhat, Frauenholz, & Cannell, 1989.
[2]Bhat, Frauenholz, & Cannell, 1989.
[3]Frauenholz & Shapiro, 1991.

Ground track drift as a function of time. The control band is shaded. Ground track maintenance maneuvers are indicated by Δv.



1.2. Scope of the User's Reference Manual

Section 2 describes the various models implemented in GTARG and cross-references the models to the available input parameters. The mathematical and algorithmic details are left to the references.[1] The purpose of section 2 is to provide the user with a basic understanding of the models which are available and the control which he or she has over those models.

GTARG was implemented on the VAX® VMS® operating system. It is assumed that the user is familiar with the use of VMS®. Section 3 describes how to execute GTARG in this environment.

All input to GTARG is provided through the use of standard FORTRAN namelists. Detailed descriptions of the input parameters available are given in section 4. The bulk of the reference material in this document is provided in this section. It has been assumed that the user has access to a standard text editor and is familiar with the editing of VAX® FORTRAN namelists.

Section 5 provides typical output files produced by GTARG.

Section 6 describes how to rebuild the source code on the standard host environment. This material will be useful if the user wishes to supply a user-defined density function, modify the source code, or transport GTARG to another platform.

[1]Shapiro & Bhat, 1993.

Appendix A summarizes the use of the EZPLOT plotting package. EZPLOT is not a part of GTARG. However, GTARG produces output files for plotting which are formatted as EZPLOT input namelists. This appendix provides the information necessary to understand the contents of the EZPLOT namelist files.[1,2]

Subjects left for the references include: the motivation[3] for creating GTARG, the importance and magnitude of the relative effects of the various orbital perturbations on the ground track,[4,5] the mathematical theories underlying the implementation,[6] how to define the reference orbit,[7,8,9] and strategies for performing maneuver targeting[10]. Pertinent references are given throughout the text.

1.3. Acknowledgments

The work described in this document was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration. GTARG was developed under the direction of R. B. Frauenholz in his role as the TOPEX/POSEIDON Navigation Team leader. R. S. Bhat developed the perturbation equations and propagation algorithm into forms which were useful for direct implementation. P. E. Cannell coded the initial versions of GTARG.

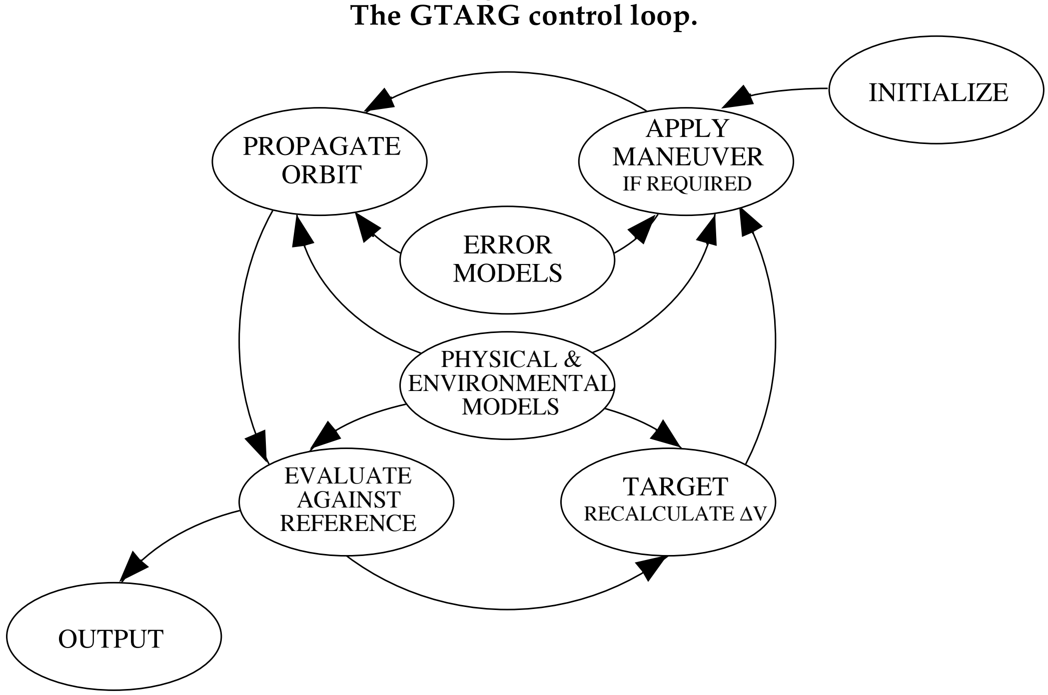
[1]Cannell, 1990.
[2]Pearson, 1989.
[3]Bhat, 1991.
[4]Bhat, Frauenholz, & Cannell, 1989.
[5]Frauenholz & Shapiro, 1991.
[6]Shapiro & Bhat, 1993.
[7]Shapiro, 1992.
[8]Vincent, 1990.
[9]Carlisle, DiCicco, Harris, Salama, & Vincent, 1991.
[10]Bhat, Frauenholz, & Cannell, 1989.

2. Description of GTARG

2.1. The GTARG Control Loop

GTARG combines orbit prediction and targeting algorithms to design ground track maintenance maneuvers. The basic control loop is shown in figure 2.1. The initial state vector is input as a set of Keplerian mean elements. In *runout* mode the ground track is propagated for a desired period of time, incorporating the effect of a pre-specified maneuver on the initial orbit. The orbit propagation model, described in section 2.2, includes all effects which cause significant perturbations on the ground track. Maneuvers are modeled impulsively, as described in section 2.3. Optional error models include the effects of uncertainties due to Δv execution error, drag unpredictability, orbit determination accuracy, and unspecified along-track satellite fixed forces. The error models, as summarized in section 2.4, can be used to predict an envelope of uncertainty about the predicted ground track with a desired confidence level, typically 95%. The orbit propagation can be combined with a targeting strategy to determine the required maneuver Δv magnitude. GTARG provides two targeting strategies, as summarized in section 2.5. These are *longitude targeting*, which maximizes the time between maneuvers, and *time targeting*, in which maneuvers are targeted to occur at specific intervals. Output is provide in the form of an executive report summary and a plot file. The executive report summary includes tables of the mean elements and ground track as a function of time which may be imported into spread sheets. The plot file is in the form of a FORTRAN namelist file. Control of the contents of the output files is described in section 3.

Figure 2.1.
The GTARG control loop.



2.2. Orbit Propagation Model

The initial state is input as a set of mean Keplerian elements (a, e, i, Ω, ω, M). Mean elements are derived using the procedure described by Guinn.[1] Internally, non-singular mean elements (near $e \approx 0$) are propagated with dynamic models that include a high-order Earth gravity field, atmospheric drag, and luni-solar gravity. The propagation algorithm includes all perturbations that cause significant variations in the satellite ground track. Recurrence formulae are used for the geopotential and luni-solar gravitational perturbation. Since the propagation step size is a integral multiple of the satellite's period, a polynomial fit[2] to the mean orbital Jacchia-Roberts density[3] at the TOPEX/POSEIDON altitude is used to predict drag. A constant density atmosphere is also available. User-defined density functions can be incorporated by re-linking the executable. (Instructions for this are given in section 6.) A variable mean area (VMA) model is used to account for drag area variation due to the nearly continuous yaw-steering of the satellite. A constant area may also be utilized. The ground track is computed once per orbit, as the longitudinal difference at the ascending node between the actual and reference nodal longitude. The orbit is propagated either for a user specified time interval (*runout* mode) or until the ground track crosses a specified edge of the control band (*targeting* mode).

The recurrence formulae enable the use of zonal harmonics to any order. GTARG was implemented to include terms from J_2 through J_{29}. Due to the form of the equations, computational speed is no longer dominated by lengthy field evaluations, and hence is relatively independent of field size. Merson's extension[4] of Grove's geopotential[5] provided the required recurrence relations for the Geopotential Perturbations in terms of internal non-singular forms of the mean elements.[6] The secular effect of $J_2{}^2$ uses the explicit expressions given by Merson. The method is based upon the theory of Kozai.[7]

Kaula's disturbing function[8] was used to develop expressions for the change in orbital parameters due to luni-solar gravity. Escobal's analytic form for the planetary ephemeris in ecliptic mean elements is used to predict the positions of the sun and the moon.[9]

Along-track satellite-fixed forces (also called "Boost" forces) which are not modeled by any of the other models in GTARG are described by table look-up of the change of semi-major axis induced by the force. The table contains a list of daily or time-tagged values of *da/dt*.

Relevant parameters are summarized in table 2.1.

---

[1]Guinn, 1991.
[2]Frauenholz & Shapiro, 1991.
[3]Roberts, 1971.
[4]Merson, 1966.
[5]Groves, 1960.
[6]Cook, 1965.
[7]Kozai, 1959.
[8]Kaula, 1962.
[9]Escobal, 1983.

Table 2.1.
Orbit Propagation Parameters.

Parameter	Description
ATDEN	Atmospheric density (for constant model).
ATDEN_ANN	Coefficients of annual correction to density (for polynomial model).
ATDEN_POLY	Polynomial coefficients for atmospheric density.
ATDEN_SEMI	Coefficients of semiannual correction to density (for polynomial model).
ATMOS	Atmospheric model selected.
ATARGONLY	Stop targeting after first guess.
BREAKPTS	TOPEX VMA model breakpoints.
CAREAS	TOPEX VMA model constant areas corresponding to breakpoints.
CD	Drag coefficient.
DAYS	Length of propagation for runout.
DATE	Epoch of ORBIT.
DRAG	Select drag perturbation.
DRAGAREA	Drag area (for constant area model).
DRAGMODEL	Drag model selected.
DSMADT_DATA	Table of *da/dt* values for boost force.
DSMADT_DATES	Table of dates for DSMADT_DATA.
JEARTH	Earth zonal coefficients.
LSFLAG	Select luni-solar gravitational perturbation.
LTOP	Number of earth zonals to use.
M	Propagation step size in orbits.
MASS	Satellite mass.
NDSMADT_DATA	Number of data points in DSMADT_DATA.
ORBIT	Initial Keplerian state vector.
REV	Revolution number at DATE.
VMATAB	TOPEX VMA table.

2.3. Maneuver Modeling

GTARG predicts the ground track evolution following an impulsive maneuver, measured with respect to the reference track. The maneuver is defined in terms of magnitude (ΔV) and direction, represented by yaw and pitch angles. The yaw and pitch angles give the direction of the ΔV vector. Relevant parameters are summarized in table 2..2.

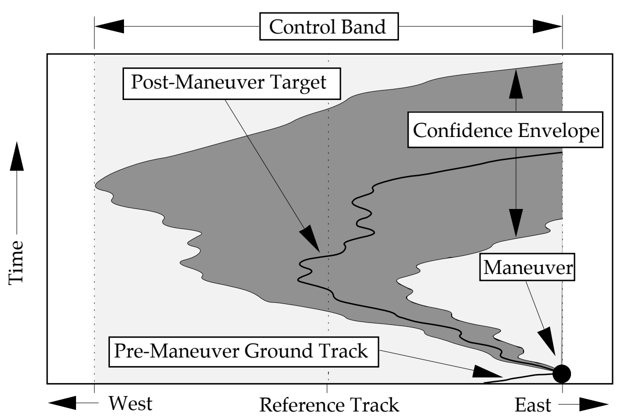
Table 2.2.
Maneuver Modeling Parameters.

Parameter	Description
DV	Maneuver ΔV magnitude.
DVQUANTA	Maneuver ΔV quantization.
PITCH	Satellite attitude pitch angle.
YAW	Satellite attitude yaw angle.

2.4. Error Models

Eastern and western error envelopes on the ground track are calculated along with the unbiased ground track as illustrated in figure 2.2. The error envelope defines the most eastward and most westward ground tracks which can reasonably be expected with a specified degree of confidence.

Figure 2.2.
Ground track confidence envelope.



The longitudinal width of the envelope is derived from anticipated maneuver execution, orbit determination (OD), drag prediction, and unmodeled along-track force ("boost/decay") errors. The drag modeling error is dominated by uncertainties in solar activity prediction. Maneuver execution errors are categorized into fixed, proportional, and pointing errors. The orbit determination error is reflected primarily as an error in the semi-major axis. Anticipated error σ's in the solar and geomagnetic indices $F_{10.7}$, $\overline{F_{10.7}}$, and K_P are used to generate high-density and low-density trajectories; the resulting differences in the ground track with the error-free trajectory are used to calculate the drag error. The three types of errors are propagated, converted into ground track units, and then added in quadrature with weight factors to determine the total error envelope.[1] Relevant parameters are summarized in table 2.3.

Table 2.3.
Error Model Parameters.

Parameter	Description
BOOST_ERROR_MODEL	Select optimistic or pessimistic "boost" error model.
DELTA_A_OD	Initial Δa error due to orbit determination.
DRAG_ERROR_MODEL	Select optimistic, pessimistic, or table look-up error model for drag.
DSMADT_DATA_SIGMA	1σ error for "boost" model (constant).
DSMADT_DATES	Dates for data in DSMADT_SIGMAS (array of time-tagged values).
DSMADT_EPOCH	Epoch of first data point in DSMADT_SIGMAS (array of daily values).
DSMADT_SIGMAS	1σ error for "boost" model (array of daily or time-tagged values, corresponding to data points in DSMADT_DATA).
GTBIASFILE	Array of daily 1σ ground track errors for drag prediction (table look-up drag error model).
NDSMADT_DATA	Number of data points in DSMADT_SIGMAS and DSMADT_DATES.
SIGMA_DV_FIXED	1σ fixed ΔV execution error (mm/sec).
SIGMA_DV_PROP	1σ proportional ΔV execution error (ratio).
SIGMA_SF_DRAG	Size of drag error envelope in standard deviations.
SIGMA_SF_BOOST	Size of "boost" error envelope in standard deviations.
SIGMA_SF_DVOD	Size of ΔV and OD error envelope in standard deviations.

[1]Shapiro, 1993A.

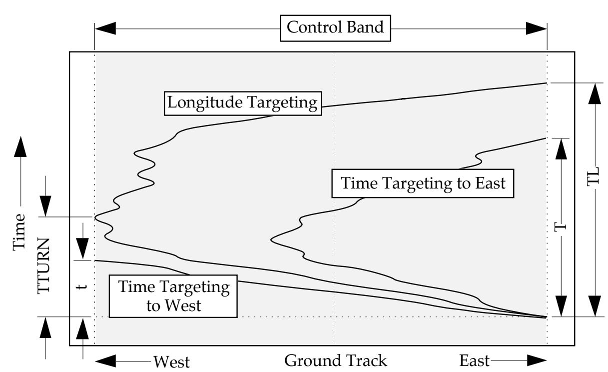
2.5 Targeting

Two classes of targeting strategies are implemented in GTARG: *longitude targeting* and *time targeting*. These are illustrated in figure 2.3.

Longitude targeting utilizes the full control band to maximize the time between maneuvers. Applying a ΔV at the eastern edge of the band, the semi-major axis is increased. The resulting higher nodal period causes the ground track to drift westward. Drag continuously reduces the nodal period until the ground track becomes just tangent to the western boundary; the ground track then reverses eastward as the period continues to decrease. Eventually the ground track returns to the eastern boundary after a time (*TL* in figure 2.3).

Alternatively, in *time targeting*, the time between maneuvers is selected first. A smaller ΔV, which will allow the ground track to return to the eastern boundary sooner than *TL* (*T* in fig. 2), is utilized for *time targeting to the eastern boundary*. Similarly, in *time targeting to the western boundary* , a larger ΔV is used, causing the ground track to cross the western boundary some time *t* prior to the longitude targeting turn-around time *TTURN*. GTARG implements these three targeting schemes along with a simple *runout* mode in which the ground track profile is predicted but no maneuver targeting is performed.

Figure 2.3.
Longitude and time targeting strategies. The control band is shaded.



Targeting involves determination of the correct ΔV magnitude for the selected targeting mode. GTARG makes as its first guess a constant-drag approximation, ignoring the earth oblateness and lunar and solar gravity perturbations. The trajectory is then successively propagated and the ΔV is modified, until a satisfactory value of ΔV, which produces the desired ground track evolution, is found. For *longitude targeting,* successful targeting means that the westernmost ground track of the western confidence envelope just reaches (within a user-supplied tolerance) but does not cross the western edge of the control band. For *time targeting*, the desired confidence envelope will reach the desired edge of the control within a user-specified tolerance of the targeted time. The second iteration will either increase or decrease ΔV, depending upon whether the targeted ground track is overshot or undershot. Subsequent iterations for ΔV are found by linear interpolation on the maximum westward ground track (for *longitude targeting)* or time of leaving the control band (for *time targeting*). The algorithm may terminate earlier if successive ΔV guesses are smaller than the allowed command quantization level.

Relevant parameters are summarized in table 2.4.

Table 2.4.
Maneuver Targeting Input Parameters.

Parameter	Description
BNDFUZ	Allowed targeting error for longitude targeting.
BOUNDS	Size of control band.
DV	First guess ΔV magnitude.
DVBRACKET	Post-targeting runouts selected.
DVQUANTA	ΔV quantization.
STRAT	Targeting strategy selected: runout, time targeting to the east or west, longitude targeting.
TARGET_STRAT	Ground track to target on: eastern error envelope, western error envelope, unbiased ground track.
TIMFUZ	Allowed targeting error for time targeting.
TIMTGT	Target time for time targeting.
WATCHINT	Saves result of intermediate iterations.

2.6. Reference Grid

Ground track offsets are measured at the ascending nodes, as longitudinal distances along the equator from the nearest reference node. Input parameters are summarized in table 2.5.

Table 2.5.
Reference Nodes.

Parameter	Description
ORBITS	Number of reference nodes.
XINGS	Node crossing longitudes of reference nodes.

Since GTARG uses the mean elements and a zonal earth oblateness perturbation, mean reference nodes, rather than osculating reference nodes, must be specified in XINGS. The mean reference nodes are equally spaced about the equator and are assumed to be ordered chronologically in the input array.

2.7. Satellite Environment and Physical Parameters

Various environmental parameters may be controlled by user input. These are summarized in table 2.6.

Table 2.6.
Satellite Environmental and Physical Parameters.

Parameter	Description
ATDEN	Atmospheric density (constant model).
ATDEN_ANN	Coefficients of annual correction to density (for polynomial model).
ATDEN_POLY	Polynomial coefficients for atmospheric density.
ATDEN_SEMI	Coefficients of semiannual correction to density (for polynomial model).
CD	Satellite drag coefficient.
EARTH_RAD	Earth equatorial radius.
FBAR_BIAS	Bias of data in FLXBAR.
FBAR_SLOPE	Slope of data in FLXBAR.
FLAT	Earth flattening f.
FLUX_BIAS	Bias of data in FLX.
FLUX_SLOPE	Slope of data in FLX.
FLX	$F_{10.7}$ solar Flux.
FLXBAR	81-day average of $F_{10.7}$ Solar Flux.
JEARTH	Earth zonal coefficients.
KP	Geomagnetic indices.
MASS	Satellite mass.
MU_EARTH	Earth GM.
MU_MOON	Lunar GM.
MU_SUN	Solar GM.
SID_DAY	Length of sidereal day.

2.8. Site Over Flights

Although the information is not directly used in the targeting process, GTARG has the capability of predicting the longitudinal overflight distances at a set of verification sites.[1] A table of offsets will be printed and, if requested, the offsets will be written to the EZPLOT output file. Key parameters are summarized in Table 2.7.

Table 2.7.
Verification Site Input Parameters.

Parameter	Description
`NSITES`	Number of sites.
`PLOTSITE`	Selects Plotting of site offsets.
`SITE_LOC`	Latitude and calibrated longitudes of sites.
`SITE_LOC_TYPE`	Indicates whether geodetic or geocentric data are input.
`SITE_NAME`	Names of the sites.
`SITE_NODE`	Number of reference node corresponding to the orbit on which the overflight occurs.
`SITE_STRAT`	Selects calculation method - Keplerian approximation or propagated solution.
`SITE_SYM`	Plot symbol for EZPLOT output.
`SITES`	Enable calculation of site offsets.

[1]Shapiro, 1993B.

2.9. Calibration of Input Parameters

GTARG propagates mean elements using only zonal terms for the earth oblateness perturbation. To account for modeling biases, the mean elements and reference nodes, must be properly calibrated with respect to the true mean elements and reference site locations. The calibration procedure is outlined in this section.

Once the osculating reference nodes are defined, the mean reference nodes are determined by selecting one of the osculating nodes and defining the remaining ones to be equally spaced around the equator. These reference nodes are all shifted by a fixed constant to null the mean offset at the overflight sights, if any. This introduces a bias in the nodes at the equator which is later corrected by shifting the right ascension of the ascending nodes of the mean elements.

The mean element calibration procedure[1] is as follows. A reference orbit is defined by turning off all perturbations except for Earth gravity and tweaking the elements until the ground track repeats after 127 revolutions.[2] This procedure is performed with both a precision numerical integrator, such as DPTRAJ[3,4] or GTDS,[5,6] and GTARG. Let $\hat{M}_i^{INTEG}$ be the reference mean elements derived from the integrator, and $\hat{M}_i^{GTARG}$ be a similar set of reference elements derived with GTARG. In general, $\hat{M}_i^{GTARG} \neq \hat{M}_i^{INTEG}$ because of propagation and other modeling differences. The mean elements M_i^{GTARG} used for propagation by GTARG are then

$$M_i^{GTARG} = M_i^{INTEG} + \hat{M}_i^{GTARG} - \hat{M}_i^{INTEG}$$

In particular, GTARG does not model tesseral field harmonics, time transformations (e.g., UT1-UTC), solid earth tides, precession, nutation and polar motion. This calibration procedure must be repeated whenever new data (e.g. timing or polar motion parameters) is added to the precision system. Adjustments will be required in the semi-major axis a, the right ascension of ascending nodes Ω, and the inclination i. First, Ω is adjusted to null the initial ground track offset introduced by the earlier shifting of the reference nodes and an additional minute change to account for variations in timing and polar motion. Next, a is adjusted so that the ground track does not precisely repeats without drift over a full cycle. Finally, the inclination is adjusted to minimize the offsets at the sites. Some iteration in the values of a and i will be required until the correct point is found. It may become desirable to slightly modify the reference grid and then repeat the entire process, until satisfactory convergence is obtained.

---

[1]Shapiro & Bhat, 1993.
[2]The actual number of revolutions is a function of the characteristics of the particular exact repeat orbit. For the TOPEX/POSEIDON mission, this number is 127.
[3]Spier, 1971.
[4]Ekelund, Sunseri, & Collier, 1991.
[5]Long, Cappellari, Velez, & Fuchs, 1989.
[6]Squier & Byers, 1987.

3. Program Execution

3.1. Sample Run Stream

GTARG is executed via the command file `GTARG.COM`. The command format is

@`GTARG` *input-file flux-file*

where

input-file - contains the input namelist parameters in namelist $`INPUT`.

flux-file - optional flux data file contains the input namelist parameters in namelist $`FLXKP`. This file is only required if parameter `DRAG = .TRUE.` and `ATMOS = 'TOPEXJR'` in namelist $`INPUT`.

The minimum contents of the run stream `GTARG.COM` are shown in figure 3.1.

Figure 3.1.
Sample run stream.

```
$DEFINE/USER_MODE   "IN_GTARG"    'P1'
$DEFINE/USER_MODE   "FLUX_DATA"   'P2'
$RUN GTARG.EXE
```

3.2. Controlling Program Output

GTARG will always produce an output summary file (parameter `OFILE`). Other optional output files are summarized in table 3.1. Output file content control and selection parameters are summarized in table 3.2.

GTARG also produces plots which are formatted as EZPLOT input namelists. These plots contain the ground track as a function of time and the components of the error envelope due to the various error models selected. The content of the EZPLOT file may be controlled by the parameters of table 3.3. EZPLOT is a PGPLOT[1] compatible utility. EZPLOT input parameters are summarized in the Appendix A.

[1]Pearson, 1989.

Table 3.1.
Output Files.

Parameter Giving File Name	Parameter Enabling File Creation	Description
BFILE	BOOT	Boot file.
NEWBIASFILE	MAKE_BIAS_FILE	New ground track biases.
OFILE		Standard output file.
ZFILE	PLOT	Plot file.
ZFILE2	PLOT_COMPONENTS	Plot of error components

Table 3.2.
Output Control Parameters

Parameter	Description
ECHO_BIAS	Print contents of ground track bias file.
ECHO_BOOST	Print formatted listing of "boost" model data.
ECHO_FLUX	Print formatted listing of solar and geomagnetic data parameters.
ECHO_GRAV	Print formatted listing of physical constants.
ECHO_INPT	Dump contents of namelist $INPUT as is.
ECHO_VMATAB	Print formatted listing of TOPEX VMA table.
ECHO_XING	Print formatted listing of reference grid.
TERMINAL	Video terminal used.
WATCH	Watch calculations interactively.
WATCHINT	Save results of intermediate targeting iterations.

Table 3.3.
EZPLOT Parameters Which May be Controlled via GTARG Input

Parameter	Description
EZHEAD	Print namelist $ZFRAME to plot file.
ODAYS	Duration of time scale on plot.
ODEVICE	PGPLOT output device.
PLOTBOOST	Plot the "boost" force as well as ground track.
PLOTCYCLE	Display and annotate the cycle boundaries.
PLOTDATE	Origin of time scale.
PLOTSITE	Display the ground track at the sites.
PLOTTITLE	Title of plot.

4. GTARG Input

All input parameters are in namelist `$INPUT` in the input control file unless otherwise specified.

ATARGONLY

Dimensions: 1
Type: Logical
Units: n/a
Default: .FALSE.

If `ATARGONLY = .TRUE.` only the first guess ΔV is calculated, and no further targeting is performed.

ATDEN

Dimensions: 1
Type: DP
Units: kg/km^3
Default: 10^{-6}

The constant atmospheric density. Used only when `DRAG = .TRUE.` and `ATMOS = 'CONST'`.

ATDEN_ANN

Dimensions: 2
Type: DP
Units: kg/km^3, radians
Default: 0.0454889572120465d+00,
6.1054952840340235d+00

Parameters of the annual correction term to the log of the density (see description under ATDEN). The form of the correction is

$$(\Delta \log \rho)_{Annual} = ATDEN\_ANN(1) \times \cos[2\pi t + ATDEN\_ANN(2)]$$

where t is the time in years into the current year, i.e., Jan. 1 at 00:00 GMT is t=0 and Dec. 31 at 24:00 GMT is t=1.0. Used only when `DRAG = .TRUE.` and `ATMOS = 'POLYNOMIAL`.

ATDEN_POLY

Dimensions: 6
Type: DP
Units: $\log_{10}$(°K), $\log_{10}$(kg/km^3)
Default: 0.102444406127929688D+04
0.414349639892578125D+03
-.581061572488852007D+01
0.720650161644122988D+00
0.842769258090778408D-04

The Chebyshev polynomial coefficients to the log of the density. The form of the polynomial is

$$
\begin{aligned}
u &= \frac{T_\infty - ATDEN\_POLY(1)}{ATDEN\_POLY(2)} \\
z_N &= ATDEN\_POLY(N+3) \quad for \quad N = 2 \\
z_{N-1} &= 2uz_N + ATDEN\_POLY(N+2) \\
z_i &= 2uz_{i+1} - z_{i+2} + ATDEN\_POLY(i+3) \\
& \quad for \quad i = N-2, \ldots, 1 \\
P(T_\infty) &= uz_1 - z_2 + ATDEN\_POLY(3)
\end{aligned}
$$

where N=2 is the order of the Chebyshev basis (see above under `ATMOS`). Used only when `DRAG = .TRUE.` and `ATMOS = 'POLYNOMIAL`.

ATDEN_SEMI

Dimensions: 2
Type: DP
Units: kg/km^3, radians
Default: -0.0791350355550989d+00
5.4144904588160714d+00

Parameters of the semiannual correction term to the log of the density (see description under ATDEN). The form of the semiannual correction is

$$
\begin{aligned}
(\Delta \log \rho)_{Semiannual} &= ATDEN\_SEMI(1) \\
& \times \cos\left[4\pi t + ATDEN\_SEMI(2)\right]
\end{aligned}
$$

where t is the time in years into the current year, i.e., Jan. 1 at 00:00 GMT is t=0 and Dec. 31 at 24:00 GMT is t=1.0. Used only when `DRAG = .TRUE.` and `ATMOS = 'POLYNOMIAL`.

ATMOS

Dimensions: 1
Type: C*10
Units: n/a
Default: `'TOPEXJR'`

Selects the atmospheric model. Not applicable unless `DRAG = .TRUE.`

`ATMOS = 'CONST'` means use a constant, fixed atmospheric density. The density is given by `ATDEN`

`ATMOS = 'TOPEXJR'` means use the TOPEX/POSEIDON mean orbital Jacchia-Roberts density calculated by `JRSMPL2`.

`ATMOS = 'POLYNOMIAL'` is the same model as `TOPEXJR` but uses the coefficients in `ATDEN_POLY`, `ATDEN_ANN`, and `ATDEN_SEMI`. The atmospheric density in kg/km^3 is modeled using the function

$$\log_{10}\rho = P(T_\infty) + (\Delta\log\rho)_{Semiannual} + (\Delta\log\rho)_{Annual}$$

where the exospheric temperature is calculated as

$$T_\infty = 379^\circ + 3.24^\circ \overline{F_{10.7}} + 1.3^\circ\left[F_{10.7} - \overline{F_{10.7}}\right] + 28^\circ Kp + 0.03^\circ e^{Kp}$$

The polynomial coefficients in P are given by `ATDEN_POLY`; the parameters of the Semiannual correction are given by `ATDEN_SEMI`; and the parameters of the annual correction are given by `ATDEN_ANN`. The solar and geomagnetic parameters are read from arrays `FLX`, `FLXBAR`, and `KP`.

`ATMOS = 'USER' means` use a user-supplied density function calculated by `USER_DENSITY`. To create a user-supplied density function, the template in `USER_DENSITY.FOR` must be completed with appropriate code to calculate the density with the user supplied model, compiled, and the executable re-linked with the new density function, prior to execution. See the section of this document on Compiling and Linking for more information.

BFILE

Dimensions: 1
Type: C*80
Units: n/a
Default: `'GTARG.BOOT'`

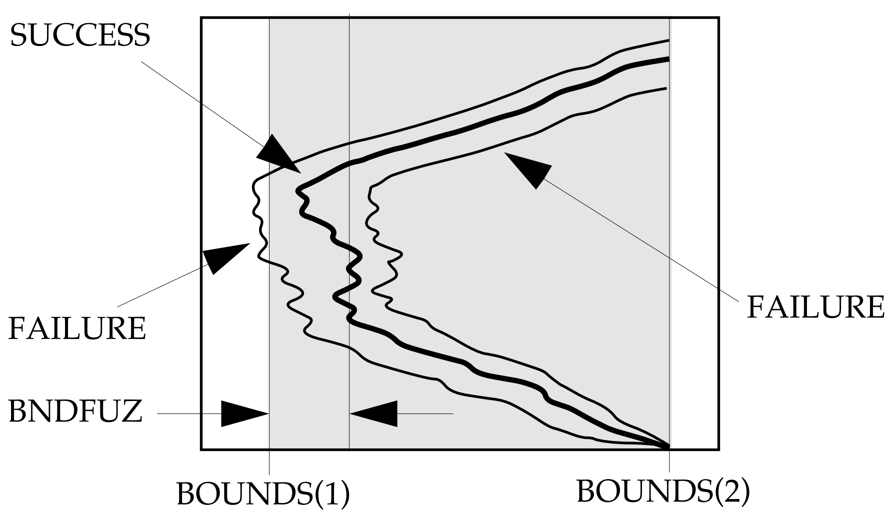
The name of the boot file. The boot file contains the namelist elements `ORBIT` and `DATE` at each propagation step, so that subsequent GTARG runs can be started ("booted") at any point. The boot file will not be created unless `BOOT = .TRUE.`

BNDFUZ

Dimensions: 1
Type: DP
Units: kilometers
Default: 0.01

Defines the convergence criterion for longitude targeting iteration (figure 4.1). The targeting iteration will continue until either (a) `BOUNDS(1)` < maximum westward ground track < `BOUNDS(1) - BNDFUZ`, or (b) the difference between successive iterations on `DV` is < `DVQUANTA`.

Figure 4.1.
Definition of BOUNDS and BNDFUZ, showing the targeting success criterion. The control band is shaded.



BOOST_ERROR_MODEL

Dimensions: 1
Type: C*12
Units: none
Default: 'OPTIMISTIC'

Selects the model used to determine the contribution to the error envelope due to boost modeling errors.[1]

`BOOST_ERROR_MODEL = 'OPTIMISTIC'` will assume that the errors for each propagation step are completely independent random variables. The total accrued error will be determined by adding the boost errors in quadrature.

`BOOST_ERROR_MODEL = 'PESSIMISTIC'` will assume that the errors for each propagation step are completely dependent random variables. The total accrued error will be determined by adding the boost errors linearly.

The boost errors will be determined from `DSMADT_DATA_SIGMA` (a constant error, the same each propagation step) or `DSMADT_SIGMAS` (a time-tagged array of errors).

BOOT

Dimensions: 1
Type: Logical
Units: n/a
Default: .FALSE.

If `BOOT = .TRUE.`, a boot file will be generated. See `BFILE`.

BOUNDS

Dimensions: 2
Type: DP
Units: kilometers
Default: -1.0, 1.0

The bounds of the control band in kilometers (see figure 4.1). Used for targeting and for labeling the EZPLOT output file.

`BOUNDS(1)` is the western boundary of the control band.

`BOUNDS(2)` is the eastern boundary of the control band.

BREAKPTS

Dimensions: 2
Type: DP

[1]Shapiro, 1993A.

Units: deg.
Default: 15˚, 75˚

Breakpoints for yaw steering logic. Used only when `DRAGMODEL = 'VMA'`.

CAREAS

Dimensions: 2
Type: DP
Units: m^3
Default: 0.0

Constant drag areas for VMA model. Used only when `DRAG = .TRUE.` and `DRAGMODEL = 'VMA'`.

`CAREAS(1)` is the area at $\beta' = 0°$.
`CAREAS(2)` is the area at $\beta' = 90°$.

CD

Dimensions: 1
Type: DP
Units: none
Default: 0.0

Drag coefficient C_D. Used only when `DRAG = .TRUE.`

DATE

Dimensions: 1
Type: C*25
Units: n/a
Default: ' '

Epoch of the initial state vector in `ORBIT`. The format of `DATE` is as in `'23-MAR-1992 17:27:54.0000'`

DAYONE

Dimensions: none
Type: C*25
Units: n/a
Default: ' '
Namelist: `$FLXKP`

Data epoch of first point in solar and geomagnetic data arrays `FLX, FLXBAR, KP, DFLX, DFLXBAR, DKP.` The remaining data in the arrays is assumed to be spaced at 1 day intervals up to `NUMDAY.` Used only when `DRAG = .TRUE.`

DAYS

Dimensions: 1
Type: Integer
Units: Days
Default: 0

If `STRAT = 'RUNOUT'`, `DAYS` gives the duration of the propagation, starting from `DATE`, in days.

If `ODAYS` is not specified, `DAYS` also gives the length of the y-axis for the `EZPLOT` output file, for any value of `STRAT`.

DELTA_A_OD

Dimensions: 1
Type: DP
Units: meters
Default: 0.0

The 1σ uncertainty in the semi-major axis of the initial elements, due to orbit determination errors alone.

DFLX

Dimensions: 1:1000, selected by `NUMDAY`.
Type: DP
Units: Solar Flux ($F_{10.7}$) Units
Default: 0.0
Namelist: `$FLXKP`

The 1σ uncertainty in the values of $F_{10.7}$ given in array `FLX`. `DFLX(I)` is the 1σ uncertainty in the value of `FLX(I)`. The data in this array will be ignored unless `DRAG = .TRUE.` and `DRAGBIASMODE = 'FLUX'`.
Namelist: `$FLXKP`

DFLXBAR

Dimensions: 1:1000, given by `NUMDAY`.
Type: DP
Units: Solar Flux ($F_{10.7}$) Units
Default: all 0.0
Namelist: `$FLXKP`

The 1σ uncertainty in the values of the 81-day centered means of $F_{10.7}$ given in `FLXBAR`. `DFLXBAR(I)` is the 1σ uncertainty in the value of `FLXBAR(I)`. The data in this array will be ignored unless `DRAG = .TRUE.` and `DRAGBIASMODE = 'FLUX'`.

DKP

Dimensions: 1:1000, given by `NUMDAY`.
Type: DP
Units: Geomagnetic index (Kp) units.
Default: all 0.0.
Namelist: `$FLXKP`

The 1σ uncertainty in the values of the geomagnetic index given in array `KP`. `DKP(I)` is the 1σ uncertainty in the value of `KP(I)`. The data in this array will be ignored unless `DRAG = .TRUE.` and `DRAGBIASMODE = 'FLUX'`.

DRAG

Dimensions: 1
Type: Logical
Units: n/a
Default: .TRUE.

The perturbation due to atmospheric drag is calculated when `DRAG = .TRUE.` The drag model used is selected by `DRAGMODEL`. Other related parameters are `ATMOS`, `ATDEN`, `CD`, `DRAGAREA`, and `VMATAB`.

DRAGAREA

Dimensions: 1
Type: DP
Units: m^2
Default: 0.0

The satellite drag area. Used only when `DRAGMODEL = 'CONSTANT'` and `DRAG = .TRUE.`

DRAG_ERROR_MODEL

Dimensions: 1
Type: C*12
Units: none
Default: `'PESSIMISTIC'`

Selects the model used to calculate the contribution to the error envelope due to drag.[1]

`DRAG_ERROR_MODEL = 'OPTIMISTIC'` - use the 1σ values of Kp, $F_{10.7}$, and Fbar in arrays `DFLX`, `DFLXBAR`, and `DKP`. To calculate the error envelope, a high-drag and low-drag trajectory will be propagated along with the nominal-drag trajectory. The contribution to the envelope will be determined from the differences between the ground track of the nominal trajectory and the ground track of the off-nominal trajectory. The

[1] Shapiro, 1993A.

daily errors are assumed to be completely independent random variables at each propagation step and the ground track error due to drag is accrued in quadrature at each propagation step. Since three trajectories are being propagated, this mode consumes approximately three times as much CPU time as DRAG_ERROR_MODEL = 'FILE'.

DRAG_ERROR_MODEL = 'PESSIMISTIC' - use the 1σ values of Kp, $F_{10.7}$, and Fbar in arrays DFLX, DFLXBAR, and DKP. To calculate the error envelope, a high-drag and low-drag trajectory will be propagated along with the nominal-drag trajectory. The contribution to the envelope will be determined from the differences between the ground track of the nominal trajectory and the ground track of the off-nominal trajectory. The daily errors are assumed to be dependent random variables at each propagation step and the ground track error due to drag is accrued linearly with time. Since three trajectories are being propagated, this mode consumes approximately three times as much CPU time as DRAG_ERROR_MODEL = 'FILE'.

DRAG_ERROR_MODEL = 'FILE' The contributions to the error envelope are input as an array of ground track errors, one point per day, in file GTBIASFILE, array GTBIAS_DRAG. The information in arrays DFLX, DFLXBAR, and DKP is ignored. Using DRAG_ERROR_MODEL = 'FILE' is approximately three times faster than using DRAGBIASMODE = 'OPTIMISTIC' or 'PESSIMISTIC'.

Hint to speed execution: if the error envelope information due to drag is not required, use DRAG_ERROR_MODEL = 'FILE', and set SIGMA_SF_DRAG= 0 to null the envelope calculation for drag. This will triple the execution speed. If the error information is needed, and multiple runs are to be performed, set DRAG_ERROR_MODEL = 'OPTIMISTIC' or 'PESSIMISTIC' MAKE_BIAS_FILE = .TRUE. on the first run. An array GTBIAS_DRAG will be created with the ground track biases in this first run. This array can be used as input on subsequent runs with DRAG_ERROR_MODEL = 'FILE' While this is not as accurate, the improved throughput may outweigh the benefit gained by higher accuracy.

DRAGMODEL

Dimensions:	1
Type:	C*10
Units:	n/a

Default: 'VMA'

Chooses the model used to compute drag. Only used when DRAG = .TRUE. The atmospheric density model must be selected via input parameter ATMOS.

DRAGMODEL = 'VMA' means use the TOPEX VMA model for area as specified in parameters VMATAB, BREAKPTS, and CAREAS.

DRAGMODEL = 'CONSTANT' means use the constant satellite area as specified in parameter DRAGAREA.

DSMADT_DATA

Dimensions: 1:1000, selected by NDSMADT_DATA
Type: DP
Units: meters/day
Default: all 0.0

DSMADT_DATA is used to describe additional forces which are not modeled by any of the physical models in GTARG. It contains a table of daily da/dt values to describe the additional "boost" force.

If DSMADT_EPOCH = ' ', then DSMADT_DATA(I) gives da/dt for the date in DSMADT_DATES(I). GTARG will interpolate between the values.

If DSMADT_EPOCH contains a date in standard format (DD-MMM-YYYY hh:mm:ss.ffff) then DSMADT_DATA contains an array of daily points. DSMADT_DATA(I) gives da/dt I-1 days after DSMADT_EPOCH. The values are applied in a step function.

The 1-σ error for DSMADT_DATA(I) is DSMADT_DATA_SIGMA or DSMADT_SIGMAS(I).

DSMADT_DATA will only be printed if ECHO_BOOST = .TRUE.

DSMADT_DATA_SIGMA

Dimensions: 1
Type: DP
Units: meters
Default: 0.0

The 1-σ uncertainty in the data values in array DSMADT_DATA. This error is used to determine the contribution of unmodeled forces to the error envelope. Each element DSMADT_DATA(I) is assumed to have a 1σ uncertainty of DSMADT_DATA_SIGMA. If DSMADT_DATA_SIGMA < 0 then the data in DSMADT_SIGMAS will be used.

DSMADT_DATES

Dimensions: 1:1000, selected by NDSMADT_DATA
Type: C*25
Units: n/a
Default: all ' '

DSMADT_DATES(I) is the date in standard format (DD-MMM-YYYY hh:mm:ss.ffff) for which the data DSMADT_DATA(I) is valid. DSMADT_DATES will be used only when DSMADT_EPOCH=' '.

DSMADT_EPOCH

Dimensions: 1
Type: C*25
Units: n/a
Default: none

DSMADT_EPOCH is the date in standard format (DD-MMM-YYYY hh:mm:ss.ffff) for which the data DSMADT_DATA(1) is valid. If DSMADT_EPOCH = ' ' then the array of dates in DSMADT_DATES will be used instead.

DSMADT_SIGMAS

Dimensions: 1:1000, given by NGTBIAS_DRAG.
Type: DP
Units: meters
Default: 0.0

The 1σ uncertainty in the data values in array DSMADT_DATA. This error is used to determine the contribution of unmodeled forces to the error envelope. DSMADT_DATA(I) is assumed to have a 1-σ uncertainty of DSMADT_SIGMAS(I). DSMADT_SIGMAS will not be used unless DSMADT_DATA_SIGMA < 0.

DV

Dimensions: 1
Type: DP

Units: mm/sec.
Default: 0.0

If STRAT = 'RUNOUT', DV gives the magnitude of maneuver to be implemented prior to propagating the ground track.

IF STRAT = 'LONG', 'EAST', or 'WEST', DV gives the first guess for targeting. If DV = 0.0, the first guess will be calculated automatically by GTARG.

DVBRACKET

Dimensions: 3
Type: I
Units: none
Default: 0, 0, 1

Only used when DVQUANTA > 0.

Following a targeting run, a number of different maneuver magnitudes will be run out in the following do loop

```
DO I = N - DVQUANT(1), N + DVQUANTA(2), DVQUANTA(3)
```

where N is the number of DVQUANTA corresponding to the targeted ΔV.

DVQUANTA

Dimensions: 1
Type: DP
Units: mm/sec
Default: 0.0

Maneuver quantization level in millimeters/second. The maneuvers will be truncated to an integer number of DVQUANTA.

DVQUANTA = 0 indicates that the maneuvers are not quantized.

EARTH_RAD

Dimensions: 1
Type: DP
Units: km
Default: 6378.140 km

The radius of the earth in kilometers.

ECHO_BIAS

Dimensions: 1
Type: Logical
Units: n/a
Default: .TRUE.

When ECHO_BIAS = .TRUE. and DRAG_ERROR_MODEL = 'FILE' the array of daily biases used will be written to the standard output file.

ECHO_BOOST

Dimensions:	1
Type:	Logical
Units:	n/a
Default:	.TRUE.

When ECHO_BOOST = .TRUE. the data in DSMADT_DATA, DSMADT_DATA_SIGMA or DSMADT_SIGMAS, and DSMADT_EPOCH or DSMADT_DATES, will be written to the standard output file.

ECHO_FLUX

Dimensions:	1
Type:	Logical
Units:	n/a
Default:	.TRUE.

When ECHO_FLUX = .TRUE. the data in FLX, KP, FLXBAR, DFLX, DKP and DFLX will be written to the standard output file.

ECHO_INPT

Dimensions:	1
Type:	Logical
Units:	n/a
Default:	.FALSE.

When .TRUE., namelist $INPUT will be written as is to the standard output file.

ECHO_GRAV

Dimensions:	1
Type:	Logical
Units:	n/a
Default:	.TRUE.

When .TRUE., the geopotential and other physical constants will be printed to the standard output file.

ECHO_XING

Dimensions: 1
Type: Logical
Units: n/a
Default: .TRUE.

When `.TRUE.`, the reference equator crossing grid will be written to the standard output file.

ECHO_VMATAB

Dimensions: 1
Type: Logical
Units: n/a
Default: .TRUE.

When .`TRUE`., if `DRAGMODEL = 'TOPEXVMA'` then the VMA table will written to the standard output file.

EZHEAD

Dimensions: 1
Type: Logical
Units: n/a
Default: .TRUE.

When .`TRUE`., the $`ZFRAME` namelist will be written to the EZPLOT output file `ZFILE`. This parameter is only applicable when `PLOT = .TRUE.`

FBAR_BIAS

Dimensions: 6
Type: DP
Units: SFU, SFU/day, ..., SFU/day**5
Default: all 0.0

`FBAR_BIAS` and `FBAR_SLOPE` allow two polynomials to be added to the mean solar flux in `FLXBAR`,

$$\overline{F_{10.7}(t)} = \overline{F_{10.7}(t)}_{input\ in\ FLXBAR} + \overline{F_{10.7}(t)}_{bias} + \overline{F_{10.7}(t)}_{slope}$$

where t is measured in days from `DAYONE` and

$$\overline{F_{10.7}(t)}_{bias} = \sum_{k=1}^{6} FBAR\_\ BIAS(I)t^{k-1}$$

FBAR_SLOPE

Dimensions: 6
Type: DP
Units: SFU, SFU/day, ..., SFU/day**5
Default: all 0.0

`FBAR_BIAS` and `FBAR_SLOPE` allow two polynomials to be added to the mean solar flux in `FLXBAR`,

$$\overline{F_{10.7}(t)} = \overline{F_{10.7}(t)}_{input\ in\ FLXBAR} + \overline{F_{10.7}(t)}_{bias} + \overline{F_{10.7}(t)}_{slope}$$

where t is measured in days from DAYONE and

$$\overline{F_{10.7}(t)}_{slope} = \sum_{k=1}^{6} FBAR\_SLOPE(I)t^{k-1}$$

FLAT

Dimensions: 1
Type: DP
Units: n/a
Default: 1/298.25

Gives the Earth flattening.

FLUX_BIAS

Dimensions: 6
Type: DP
Units: SFU, SFU/day, ..., SFU/day**5
Default: all 0.0

FLUX_BIAS and FLUX_SLOPE allow two polynomials to be added to the mean solar flux in FLX,

$$F_{10.7}(t) = F_{10.7}(t)_{input\ in\ FLX} + F_{10.7}(t)_{bias} + F_{10.7}(t)_{slope}$$

where t is measured in days from DAYONE and

$$F_{10.7}(t)_{bias} = \sum_{k=1}^{6} FLUX\_BIAS(I)t^{k-1}$$

FLUX_SLOPE

Dimensions: 6
Type: DP
Units: SFU, SFU/day, ..., SFU/day**5
Default: all 0.0

FLUX_BIAS and FLUX_SLOPE allow two polynomials to be added to the mean solar flux in FLX,

$$F_{10.7}(t) = F_{10.7}(t)_{input\ in\ FLX} + F_{10.7}(t)_{bias} + F_{10.7}(t)_{slope}$$

where t is measured in days from DAYONE and

$$F_{10.7}(t)_{slope} = \sum_{k=1}^{6} FLUX\_SLOPE(I)t^{k-1}$$

FLX

Dimensions: 1:1000, given by NUMDAY.
Type: DP
Units: Solar Flux ($F_{10.7}$) Units
Default: 0.0
Namelist: $FLXKP

$F_{10.7}$ solar flux. FLX(1) is the $F_{10.7}$ on DAYONE, and FLX(I) is $F_{10.7}$ I-1 days later.

FLXBAR

Dimensions: 1:1000, given by NUMDAY.
Type: DP
Units: Solar Flux ($F_{10.7}$) Units
Default: 0.0
Namelist: $FLXKP

The 81-day centered average $\overline{F_{10.7}}$ of the $F_{10.7}$ solar flux. FLXBAR(1) is $\overline{F_{10.7}}$ on DAYONE, and FLXBAR(I) is $\overline{F_{10.7}}$ I-1 days later.

GTBIAS_DRAG

Dimensions: 1:1000, given by NGTBIAS_DRAG.
Type: DP
Units: kilometers
Default: all 0.0
Namelist: $GTBIAS, in file whose name is specified in GTBIASFILE.

Gives the contribution of drag prediction uncertainty to the ground track error envelope. Will not be used unless DRAG_ERROR_MODEL = 'FILE'. This array contains the 1σ uncertainty in the ground track due to drag prediction uncertainty. GTBIAS_DRAG(I) contains the uncertainty on the i'th day after DATE.

JEARTH

Dimensions: 2:29, selected by LTOP.
Type: DP
Units: none
Default: all 0.0

The zonal coefficients of the Earth's geopotential field.

KP

Dimensions: 1:1000, given by NUMDAY.
Type: DP
Units: Geomagnetic Index (Kp) Units
Default: 0.0
Namelist: $FLXKP

Geomagnetic index K_p KP(1) is K_p on DAYONE and KP(I) is K_p I-1 days later.

LSFLAG

Dimensions: 1
Type: Logical
Units: n/a
Default: .TRUE.

Selects the luni-solar gravitational perturbation.

LTOP

Dimensions: 1
Type: I
Units: n/a
Default: 0

Selects the size of geopotential model to be used. The requested zonal coefficients are input via JEARTH.

M

Dimensions: 1
Type: Integer
Units: n/a
Default: 4

The propagation step size as an integral number of orbits. Values are allowed only ranging from 1 to 10.

MAKE_BIAS_FILE

Dimensions: 1
Type: Logical
Units: n/a
Default: .TRUE.

If MAKE_BIAS_FILE = .TRUE. and DRAG_ERROR_MODEL is different from 'FILE', the drag biases will be written to NEWBIASFILE.

MASS

Dimensions: 1
Type: DP
Units: kg
Default: 0.0

The mass of the satellite in kilograms.

MU_EARTH

Dimensions: 1
Type: DP

Units: km^3/sec^2
Default: 398600.44807345 km^3/sec^2

The earth's gravitational GM.

MU_MOON

Dimensions: 1
Type: DP
Units: km^3/sec^2
Default: 4902.7927809104 km^3/sec^2

The lunar gravitational GM. Luni-solar gravity is only used when `LSFLAG = .TRUE.`

MU_SUN

Dimensions: 1
Type: DP
Units: km^3/sec^2
Default: 132712441933.00783456 km^3/sec^2

The solar gravitational GM. Luni-solar gravity is only used when `LSFLAG = .TRUE.`

NDSMADT_DATA

Dimensions: 1
Type: Integer
Units: n/a
Default: 0

The number of data points (up to 1000) in `DSMADT_DATA`, `DSMADT_DATES`, and `DSMADT_SIGMAS`.

NEWBIASFILE

Dimensions: 1
Type: C*80
Units: n/a
Default: `'NEWBIASES.OUT'`

If `MAKE_BIAS_FILE = .TRUE.` and `DRAG_ERROR_MODEL` is different from `'FILE'`, the drag biases will be written to `NEWBIASFILE`.

NGTBIAS_DRAG

Dimensions: 1
Type: I
Units: none
Default: 0
Namelist: $GTBIAS, in file whose name is specified in GTBIASFILE.

Gives the number o f elements in GTBIAS_DRAG.

NSITES

Dimensions: 1
Type: I
Units: n/a
Default: 0

Gives the number of sites specified in SITE_NAME and SITE_LOC.

NUMDAY

Dimensions: 1
Type: I
Units: n/a
Default: 0

Size of solar and geomagnetic data and error model arrays FLX, DFLX, FLXBAR, DFLXBAR, KP, and DKP.

ODAYS

Dimensions: 1
Type: I
Units: days
Default: DAYS

Length of the y-axis on the EZPLOT output file. Only used when PLOT = .TRUE.

ODEVICE

Dimensions: 1
Type: C*12
Units: n/a
Default: '/IMPRESS'

Output device for EZPLOT output file. Only used when PLOT = .TRUE. Any device which is acceptable to PGPLOT may be used.

OFILE

Dimensions: 1
Type: C*80
Units: n/a
Default: 'OUTPUT.LIS'

Name of the standard GTARG output file.

ORBIT

Dimensions: 6
Type: DP
Units: km, degrees
Default: all 0.0

The input keplerian state vector, valid at DATE.

ORBIT(1) is the semi-major axis a in km.
ORBIT(2) is the eccentricity e.
ORBIT(3) is the inclination i in degrees.
ORBIT(4) is the right ascension of ascending node Ω in degrees.
ORBIT(5) is the argument of perigee ω in degrees.
ORBIT(6) is mean anomaly M in degrees.

ORBITS

Dimensions: 1
Type: Integer
Units: n/a
Default: none

ORBITS gives the number of reference equator crossings in the array XINGS.

PITCH

Dimensions: 1
Type: DP
Units: degrees
Default: 0˚

Pitch angle at which the maneuver is to be applied. PITCH = 0 and YAW = 0 corresponds to along-track with a positive ΔV along the velocity direction.

PLOT

Dimensions: 1
Type: logical
Units: n/a
Default: .TRUE.

If PLOT = .TRUE. then GTARG will generate an EZPLOT input file ZFILE with the ground track.

PLOTBOOST

Dimensions: 1
Type: logical

Units: n/a
Default: .FALSE.

If `PLOTBOOST = .TRUE.` and `PLOT = .TRUE.` then the extra da/dt in `DSMADT_DATA` will be plotted along with the ground track in the EZPLOT output file `ZFILE`.

PLOT_COMPONENTS

Dimensions: 1
Type: logical
Units: n/a
Default: none

If `PLOT_COMPONENTS = .TRUE.` then an EZPLOT input file `ZFILE2` showing the error envelope for each component will be generated. The primary EZPLOT file `ZFILE` only contains the RSS envelope.

PLOTCYCLE

Dimensions: 1
Type: logical
Units: n/a
Default: .TRUE.

If `PLOTCYCLE = .TRUE.` and `PLOT = .TRUE.` then the cycle boundaries will be annotated in the EZPLOT output file `ZFILE`.

PLOTDATE

Dimensions: 1
Type: C*25
Units: n/a
Default: `DATE`

Specifies the origin of the y axis in standard format (`DD-MMM-YYYY hh:mm:ss.ffff`) for the EZPLOT output file. Only valid when `PLOT = .TRUE.`

PLOTSITE

Dimensions: 1
Type: logical
Units: n/a
Default: `.FALSE.`

If `PLOTSITE=.TRUE.`, then the site overflight offsets will be plotted on the EZPLOT output. Also requires `PLOT = .TRUE.` and `SITES = .TRUE.`

PLOTTITLE

Dimensions: 1
Type: C*50
Units: n/a
Default: none

Title of the EZPLOT graph.

REV

Dimensions: 1
Type: Integer
Units: orbits
Default: 1

Revolution number of the input state `ORBIT`. Used to label the output report file.

SID_DAY

Dimensions: 1
Type: DP
Units: sec
Default: 86164.09055 sec

The length of the sidereal day in seconds. `SID_DAY` is used to determine the earth's sidereal rate ω_e.

SIGMA_DV_FIXED

Dimensions: 1
Type: DP
Units: mm/sec
Default: 0.0

Gives the 1σ fixed uncertainty in the maneuver magnitude `DV`. The total maneuver execution uncertainty is found by root-sum-squaring the errors in ΔV units of `SIGMA_DV_FIXED` and `SIGMA_DV_PROP`.

SIGMA_DV_PROP

Dimensions: 1
Type: DP
Units: none
Default: 0.0

Gives the 1σ proportional error in the maneuver magnitude `DV`, specified as a proportion of the total maneuver magnitude. The total maneuver execution uncertainty is found by root-sum-squaring the errors in ΔV units of `SIGMA_DV_FIXED` and `SIGMA_DV_PROP`,

$$\sigma_{\Delta V} = \sqrt{\sigma_{fixed}^2 + \left(\sigma_{proportional} \cdot \Delta V\right)^2}$$

The total ΔV execution error is used to determine the contribution to the ground track error envelope due to ΔV errors.

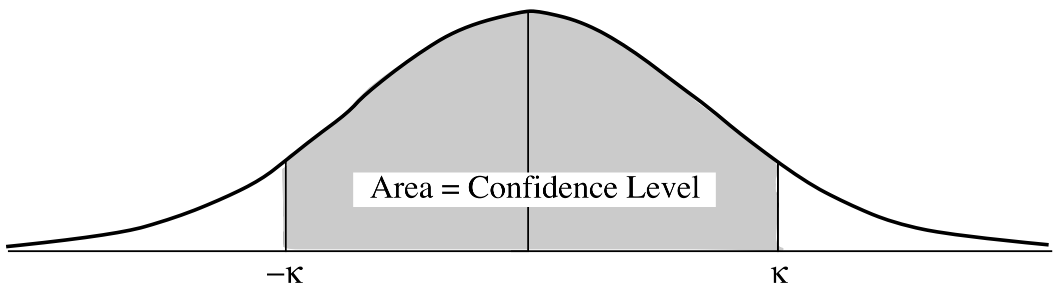
SIGMA_SF_BOOST
SIGMA_SF_DRAG
SIGMA_SF_DVOD

Dimensions: 1
Type: DP
Units: none (standard deviations, σ)
Default: 1.0

These three parameters are the weight factors which determine the contribution of each error source in standard deviations. They are used to weight the propagated error sources to determine the width of the error envelope in ground track units. The uncertainties due to all error models are root-sum-squared in ground track units ($\Delta\lambda$) as

$$\Delta\lambda = \sqrt{\kappa_{da/dt}^2 \Delta\lambda_{da/dt}^2 + \kappa_{\Delta V\&OD}^2\left(\Delta\lambda_{\Delta V}^2 + \Delta\lambda_{OD}^2\right) + \kappa_{Drag}^2 \Delta\lambda_{Drag}^2} .$$

Figure 4.2.
Demonstration of confidence levels for error sources which are represented as random variables with a standard normal distribution. The confidence level A and scale factor k are related by $\boxed{A = \frac{1}{\sqrt{2\pi}}\int_{-\kappa}^{\kappa} e^{-z^2/2}\,dz}$.



SIGMA_SF_BOOST is $\kappa_{da/dt}$, which gives the contribution due to the unmodeled forces in DSMADT_DATA.

SIGMA_SF_DRAG is κ_{drag} , which gives the contribution due to drag prediction errors, either from DFLX, DFLXBAR, and DKP, or from GTBIAS_DRAG .

SIGMA_SF_DRAG is $\kappa_{\Delta V\&OD}$, gives the contribution due to maneuver execution errors (derived from SIGMA_DV_FIXED and SIGMA_DV_PROP) and orbit determination uncertainty (derived from DELTA_A_OD).

The values of κ are typically based upon an assumption of error sources which can be represented as random variables with a standard normal distribution function and are derived from the desired level of confidence which the error envelope is meant to represent. The relationship between the scale factors and confidence levels is illustrated in figure 4.2 and typical values are shown in figures 4.3 and 4.4.

Figure 4.3.
Relationship between scale factor and confidence level.

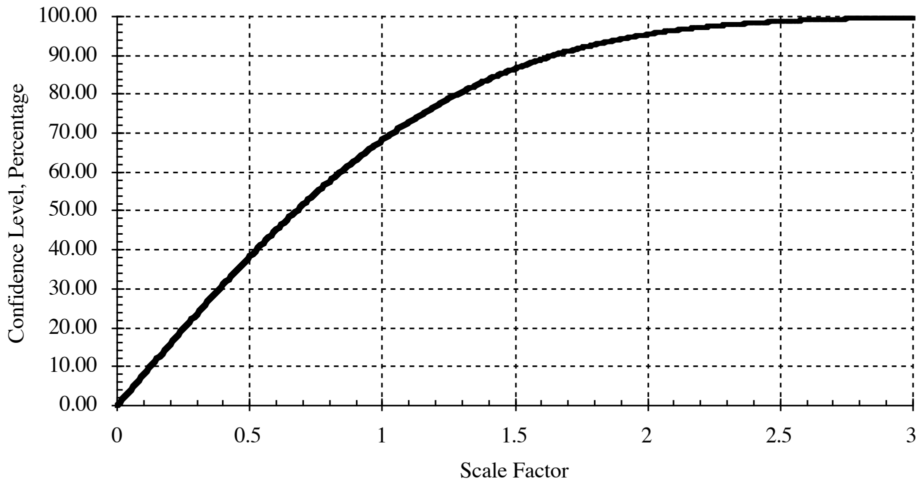
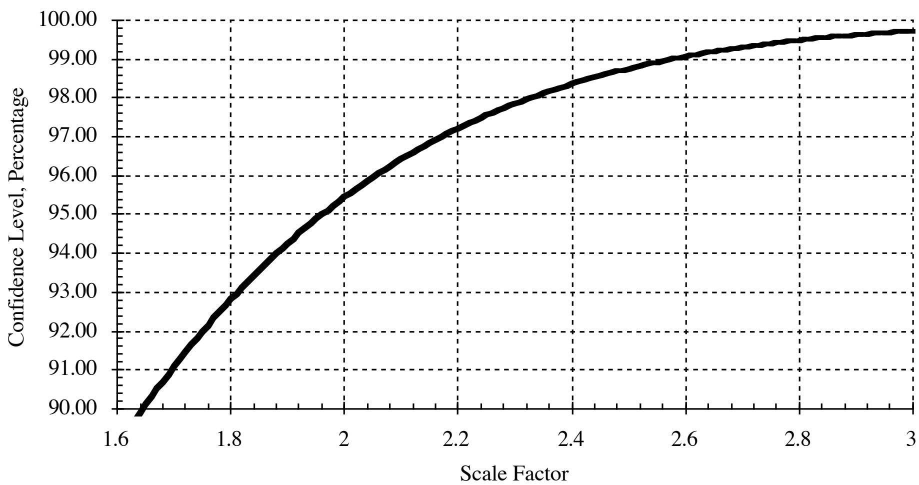


Figure 4.4.
Relationship between scale factor and confidence level for high confidence levels.



SITE_LOC

Dimensions: (2,NSITES) where NSITES ≤ 10
Type: DP
Units: degrees
Default: 239.31919, 34.4691,
12.32054, 35.54649,
remainder all 0.0d0

If SITES = .TRUE. then the longitudinal overflight distances of the sites whose locations are give by SITE_LOC will be calculated. This data will also be plotted if PLOTSITE = .TRUE., and PLOT = .TRUE.

SITE_LOC(1,I) gives the longitude of site I in degrees.

SITE_LOC(2,I) gives the latitude of site I in degrees.
The type of coordinates are specified by SITE_LOC_TYPE(I).

SITE_LOC_TYPE

Dimensions: 1
Type: C*10
Units: n/a
Default: 'GEODETIC'

Specifies the type of coordinates which are give in SITE_LOC. Options are 'GEODETIC' and 'GEOCENTRIC'.

SITE_NAME

Dimensions: NSITES where NSITES ≤ 10
Type: C*10
Units: n/a
Default: 'NASA', 'CNES', remainder all ' '

SITE_NAME(I) gives the name of the site whose location is give in SITE_LOC(I).

SITE_NODE

Dimensions: NSITES, where NSITES ≤ 10
Type: I
Units: n/a
Default: 22, 111, remainder all 0

SITE_NODE(I) gives the rev # (in XINGS) of the orbit with an ascending node just prior to the overflight of sight I.

SITE_STRAT

Dimensions: 1
Type: C*6
Units: n/a
Default: 'PROP'

Determines the strategy to be used in calculating the longitudinal site overflight distances. The propagation used from the node to the verification site is independent of the propagation used by the rest of GTARG, and hence the input to this parameter will only affect the site offset calculation.

SITE_STRAT = 'KEPLER' means that a Keplerian approximation will be used to propagate the orbit from the node to the site. This is faster but less accurate than 'PROP'

SITE_STRAT = 'PROP' means that the full GTARG model (including all perturbations) will be used to propagate the orbit from the node to the site.

SITE_SYM

Dimensions: 10
Type: I
Units: n/a
Default: 0,7,5,2,4,6,3,12,8,9
(box, triangle, x, +, circle, diamond, asterisk, star, + in circle, dot in circle)

SITE_SYM(I) is the PGPLOT plotting symbol which will be used to plot the longitudinal site overflight offset distance for site I (SITE_NAME(I), SITE_LOC(I,-)) on the EZPLOT output file. The plotting of site data requires SITES=.TRUE., PLOTSITE = .TRUE., and PLOT = .TRUE. Any valid PGPLOT plotting symbol may be used.[1]

SITES

Dimensions: 1
Type: I
Units: n/a
Default: .FALSE.

If .TRUE., will calculate and print out the longitudinal overflight distances for the sites specified in SITE_LOC. The data will not be plotted unless SITES=.TRUE. and PLOT = .TRUE.

STRAT

Dimensions: 1
Type: C*6

[1]Pearson, 1989.

Units: n/a
Default: 'RUNOUT'

Selects the targeting strategy.

STRAT = 'RUNOUT' - ground track propagation without targeting.

STRAT = 'LONG' - longitude targeting. The size of the control band is given by BOUNDS.

STRAT = 'EAST' - time targeting to the eastern boundary. The size of the control band is given by BOUNDS and the desired time of the next maneuver by TIMTGT.

STRAT = 'WEST' - time targeting to the western boundary. The size of the control band is given by BOUNDS and the desired time of the next maneuver by TIMTGT.

TARGET_STRAT

Dimensions: 1
Type: C*8
Units: n/a
Default: 'UNBIASED'

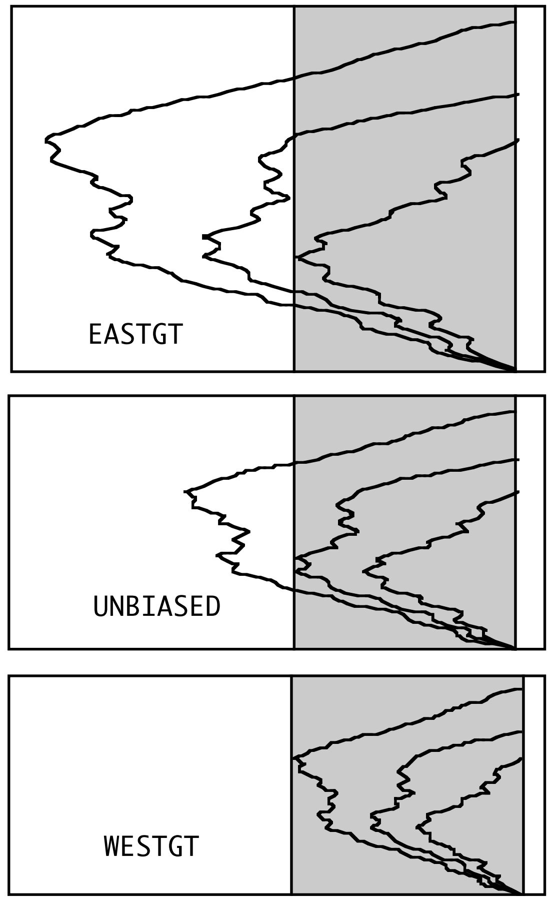
For STRAT = 'LONG', 'WEST', 'EAST', selects which ground track to apply the targeting to. TARGET_STRAT is ignored if STRAT = 'RUNOUT'

TARGET_STRAT = 'UNBIASED'- the unbiased ground track is used.

TARGET_STRAT = 'EASTGT'- the eastern error envelope is used.

TARGET_STRAT = 'WESTGT'- the eastern error envelope is used.

Figure 4.5
Use of TARGET_STRAT in longitude targeting. The control band is shaded.



TEXT

Dimensions: 1
Type: C*80
Units: n/a
Default: ' '
Namelist: FLXKP

Text used for labeling solar flux data. Not used by this version of GTARG.

TERMINAL

Dimensions: 1
Type: C*10
Units: n/a
Default: 'VT100'

If `TERMINAL = 'VT100'` the screen attributes of the VT-series terminals are utilized for the screen output. Otherwise, no special attributes are assumed.

TESTCASE

Dimensions: 1

Type: C*80
Units: n/a
Default: ' '

Label to be printed on the first page of the output and the top of the screen. If TESTCASE is not specified or TESTCASE = '?' then the input in PLOTTITLE will be used, it specified.

TIMFUZ

Dimensions: 1
Type: DP
Units: Days
Default: 1.0

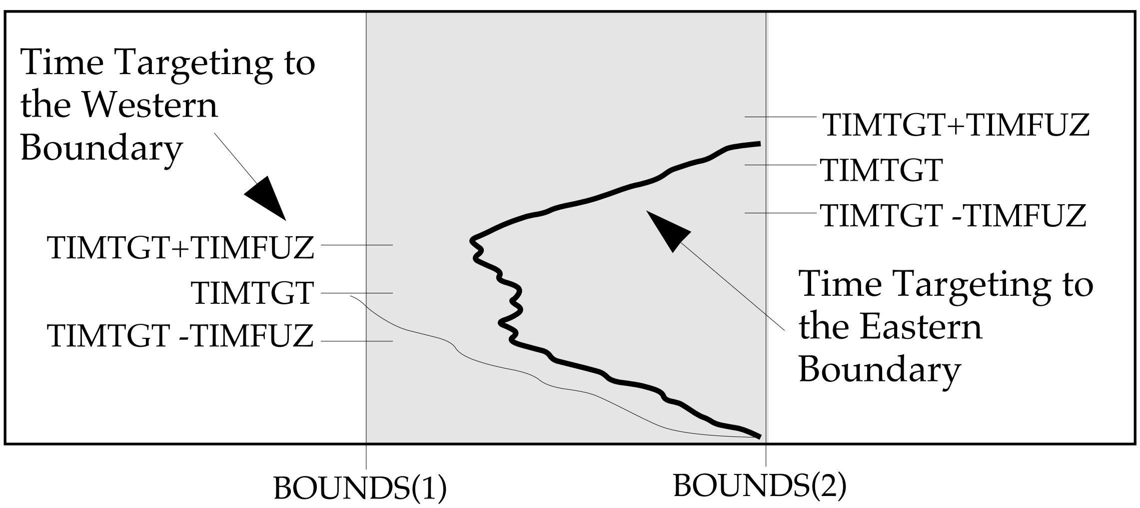
For either time targeting strategy, TIMFUZ gives the accuracy requirement for meeting the target date specified by TIMTGT (see figure 4.6)

TIMTGT

Dimensions: 1
Type: DP
Units: Days
Default: 0.0

Number of days for time targeting (see figure 4.6).

Figure 4.6.
Definition of TIMFUZ and TIMTGT shown the success criterion for time targeting. The control band is shaded.



VMATAB

Dimensions: (3, -90:90)
Type: DP
Units: degrees, m^2, m^2
Default: all 0.0

Data for TOPEX/POSEIDON Variable Mean Area (VMA) model. Used when `DRAG = .TRUE.` and `DRAGMODEL = 'VMA'`.

`VMATAB(1,I)` are the β' angles in degrees. Normally, `VMATAB(1,I) = I`°, but this is not required.

`VMATAB(2,I)` are the corresponding average drag areas for β' =`VMATAB(1,I)`.

`VMATAB(3,I)` a are the corresponding average solar radiation pressure areas for β' =`VMATAB(1,I)`. Note: This column is not used by the current version of GTARG as it does not have solar radiation pressure implemented. This is column left in for compatibility with DPTRAJ.

WATCH

Dimensions: 1
Type: Logical
Units: n/a
Default: .FALSE.

Monitor the calculations interactively on the terminal. Note: Set `WATCH = .FALSE.` when submitting batch jobs.

WATCHINT

Dimensions: 1
Type: Logical
Units: n/a
Default: .FALSE.

Save the intermediate iterations during the targeting runs.

XINGS

Dimensions: 1:200, selected by `ORBITS`
Type: DP
Units: degrees
Default: all 0.0

The equator crossing longitudes in degrees, of the reference grid. The crossings are assumed to be in time-order through a cycle.

YAW

Dimensions: 1
Type: DP
Units: degrees
Default: 0˚

Yaw angle at which the maneuver is to be applied. `PITCH = 0` and `YAW = 0` corresponds to along-track with a positive ΔV along the velocity direction.

ZFILE

Dimensions: 1
Type: C*80
Units: n/a
Default: 'GROUND_TRACK.PLOT'

The standard EZPLOT input file created when `PLOT = .TRUE.`

ZFILE2

Dimensions: 1
Type: C*80
Units: n/a
Default: 'BIAS_BREAKDOWN.PLOT'

The standard EZPLOT input file created when `PLOT_COMPONENTS = .TRUE.`

5. Sample GTARG Output

When GTARG is executed, the initialization message shown in figure 5.1 will be displayed on the screen.

Figure 5.1.
GTARG Initialization Message.

```
@GTARG TEST.DAT FLUX.TEST

****************************************************************************
* GTARG Ground Track Targeting Program Ver. 5.2 created  3-MAR-93 18:20:56 *
****************************************************************************
* Copyright (C) 1993, California Institute of Technology.                  *
* U.S. Government Sponsorship under NASA Contract NAS7-918 is acknowledged.*
****************************************************************************

Reading input namelist data ...
Initializing ...
```

If `TERMINAL = 'VT100'`, following initialization, the screen will be cleared and the display will look as shown in figure 5.2. If `WATCH = .TRUE.`, the parameters under the label `Current Iteration` will be updated continuously as they are being computed. If `WATCH = .FALSE.`, the section `Current Iteration` will be left blank.

Figure 5.2.
GTARG Screen Display.

```



GTARG Ground Track Targeting Program
Copyright (C) 1993, California Institute of Technology.
U.S. Government Sponsorship under NASA Contract NAS7-918 is acknowledged.
GTARG Version 5.2 Created  3-MAR-93 18:20:56
-------------------------- GTARG    RUNOUT Parameters --------------------------
03-MAR-1993 10:00:00.000
       a 7714.43633     Iteration   1       dv    0.00   Bias mode  FLUX
       e   0.000116   Lunar/Solar   T    Quanta  0.0000   d(sma)/dt  Input Array
       i   66.04004       Drag T  2.3   Band -1.0   1.0
    Node  188.83570     Step Size  10    Bracket  0 0 1  Scale Fact 1.960  1.960
 Perigee   74.26850 Grav Fld Size  20    B Fuzz  0.0100    Sigma dv 0.004  0.010
       M   12.62016   Runout Days  60    T Fuzz  1.0000    Sigma  a 0.333
------------------------------ Current Iteration ------------------------------
15-MAR-1993 21:21:35.1665                             Days      12.4733236863
West   -0.1821429783       a    7714.4359793923       RAAN     162.9361897162
  GT   -0.1176322573       e       0.0001133521        AOP      72.9696178701
East   -0.0525263077       i      66.0406397139          M     287.0428014289
```

After the run completes, the screen will again be cleared and the summary message shown in figure 5.3 will be displayed.

Figure 5.3.
GTARG Completion Message.

```
--------------------------- Ground Track Results ----------------------------
                ---- Unbiased ----  ---- 95% West ----  ---- 95% East ----
                    Time        GT      Time        GT      Time        GT
      1st Node   0.76192  -0.34193   0.76192  -0.34193   0.76192  -0.34193
 Furthest West   0.76192  -0.34193   0.76192  -0.34193   0.76192  -0.34193
 Furthest East  60.09968   0.83954  60.09968   0.45540  60.09968   1.27052
    Final Node  60.09968   0.83954  60.09968   0.45540  60.09968   1.27052

Run completed. 5-MAR-93 20:45:38
***************************************************************************
Total CPU time =          58.3 seconds.
Elapsed time =           103.3 seconds.
CPU Utilization =         56.4 percent.
***************************************************************************
Output File: NAVDEV:[SHAPIRO.MANEUVER]TEST.GTARG;3
EZPLOT Files:NAVDEV:[SHAPIRO.MANEUVER]TEST.PLOT;3
             NAVDEV:[SHAPIRO.MANEUVER]TEST.COMPPLOT;3
Boot File:   NAVDEV:[SHAPIRO.MANEUVER]TEST.BOOT;3
***************************************************************************
Thank you for using GTARG!
$
```

Every GTARG run will produce a standard output listing file. A typical output file, `EXAMPLE.REPORT`, is shown in figure 5.4.. The namelist input `EXAMPLE.DAT` used for this run is given in figure 5.5. The flux data file `GFLUX.NML` is given in figure 5.6. Portions of these files were omitted for brevity where the nature or format of the data was completely clear. The output plot file is in the format of an EZPLOT namelist file. (see Appendix A). Since EZPLOT is not part of the GTARG software release, the EZPLOT output is not shown. However, it is straightforward to plot the results of a GTARG run using any standard spreadsheet or graphics program. An example of this using the data in figure 4 is shown in figure 5.7 .

6. Software Installation and Modification

6.1. Installation of GTARG

To execute GTARG using the runstream in chapter 3, the following executable file is required:

GTARG.EXE

With the exception of the two user-defined data files described in chapters 3 through 5, (the input namelist file and the solar/geomagnetic data file) no other files are required to run GTARG. In lieu of creating the runstream file of figure 3.1, the following files may be installed:

GTARG.COM
GTARGJOB.COM

Instructions for executing these files may be obtained by entering the command

@GTARG ?

The sample GTARG.COM file will prompt for any missing parameters and then executed GTARG. The second file, GTARGJOB.COM, is executed by GTARG.COM. Sample data files are

EXAMPLE.FLUX
EXAMPLE.DAT

These data files correspond to the files illustrated in figures 5.5 and 5.6.

To interpret the plotting files produced by GTARG requires the files

EZPLOT.COM
EZPLOT.EXE

Instructions for the execution of EZPLOT are given in chapter 7. EZPLOT requires the presence of the PGPLOT real-time graphics library.[1]

---

[1]PGPLOT is written by the California Institute of Technology Astronomy Department. It is not included in the COSMIC GTARG package. PGPLOT is described by Pearson, 1989. If PGPLOT is not available, none of the functionality of GTARG will be lost. Since all of the ground track data which is included in the EZPLOT output file is also included in the GTARG report file in a tabular form, the data may always be plotted in any standard spreadsheet. The user may also write a graphics interpreter for whatever system is available using the information in chapter 7, and then plot the EZPLOT files directly.

All of the remaining files on the COSMIC tape contain the source code and command files for re-compilation and re-linkage of GTARG, and are not required unless the software is being modified.

6.2. Compiling and Linking GTARG

To transport GTARG to another platform or to supply a user-defined density function, the executable will need to be rebuilt. The GTARG executable can be rebuilt from the original VAX-FORTRAN by executing command procedure MAKEGTARG.COM. An object library GTARG.OLB must exist for MAKEGTARG.COM to run without error. This library can be created from any .OBJ module in GTARG. If no .OBJ files are available, compile GTARG first and then create the library:

```
$FOR GTARG
$LIB/CREATE GTARG GTARG
$@MAKEGTARG ALL
```

To compile all modules in GTARG and link the executable,

```
$@MAKEGTARG ALL
```

To compile only a single module and then re-link the executable,

```
$@MAKEGTARG module_name
```

To compile without linking,

```
$@MAKEGTARG module_name NOLINK
```

To transport to another platform which does not support VAX-FORTRAN, substantial code modification may be required.

6.3. User-Defined Density Function

The function USER_DENSITY contains a template for the user-supplied density function. Since the density is applied at most once per orbit, function USER_DENSITY must provide an orbital average density in kg/km^3. The correct calling sequence is

```
DOUBLE PRECISION RHO, USER_DENSITY, TIME, FLUX, FLUXBAR, KP
EXTERNAL USER_DENSITY
.
.
RHO =  USER_DENSITY ( TIME, FLUX, FLUXBAR, KP )
```

where the input parameters are:

TIME = the fraction of a year of the current epoch, e.g., Jan. 1 = 0, Dec 31 = 1.0;

FLUX = F10.7 solar flux;

FLUXBAR = 81 day average of F10.7 solar flux;

KP = geomagnetic index.

The value of the function is the density in kg/km^3.

Once the code has been created it can be added to GTARG by following the procedure in section 6.2.

6.4. Software Structure

The software structure is summarized in tables 6.1 and 6.2 on the following pages. These tables are only intended to provided an overview of the software structure. Detailed descriptions of the individual modules and the interface formats are given in the comments to the code.

Table 6.1.
GTARG Subroutines.

Subroutine	Calling Module	Description
CAR2KEP	DOMNVR	Coverts Cartesian state vector to Keplerian elements.
CHECKDATES	GTARG	Checks and formats the contents of the array DSMADT_DATES.
CHECK_SUCCESS	TGTGT	Formats call to CHKSUC.
CHKDV	TGTGT	Restricts range of ΔV that bounds target.
CHKSUC	CHECK_SUCCESS	Determines if ground track has been successfully targeted.
CLEAR_SCREEN	FIND_CLASS TGTGT	Clears the screen.
COMPGT	TGTGT	Computes the ground track for a specified duration of time. Drives the propagation algorithm.
COPY84	TGTGT	Copies an array of double precision data into a single precision array.
COPY88	TGTGT	Copies an array of double precision data.
CRMGET	FIRSTGUESS PROP	Recovers the density at a given time. Entry point to subroutine CRM.
CRMINI	GTARG	Initializes the density array. Entry point to subroutine CRM.
CROSS	CAR2KEP DOMNVR	Calculates vector cross product.
DISPLAY_LIMITS	TGTGT	Displays the results of a targeting iteration on the screen.
DOMNVR	TGTGT	Adds ΔV to the state vector.
DVEAST	FNDDV	Compute ΔV for time targeting to the east.
DVLONG	FNDDV	Computes ΔV for longitude targeting.
DVMOVE	COMPGT FIRSTGUESS MNODES TGTGT	Copies a state vector.
DVWEST	FNDDV	Computes ΔV for time targeting to the west.
FIND_CLASS	TGTGT	Classifies the ground track.
FINDDATE	COMPGT PROP	Finds correct point for interpolation into DSMADT_DATA array based upon dates in DSMADT_DATES array.
FNDCLS	FIND_CLASS	Classifies the ground track.
FNDDV	TGTGT	Calculate the ΔV for the next iteration.

Table 6.1 (Page 2 of 2)

Subroutine	Calling Module	Description
FIRSTGUESS	TGTGT	Calculate the first guess at ΔV.
INIT_PARMS	GTARG	Initializes astrodynamic parameters.
INTERP_DRAG_BIAS	COMPGT	Interpolates into drag bias array, when DRAG_BIAS_MODE = 'GT'
INTERP_LINE	COMPGT	Linear interpolation.
KEP2CAR	DOMNVR	Converts Keplerian elements into a Cartesian state vector.
LUNORB	LSRGPJ	Calculates the lunar ephemeris.
LSRGPB	LSRGPJ PROP	Bulletin board used by LSRGPJ.
LSRGPJ	PROP	Orbit propagator.
MNODES	COMPGT FIRSTGUESS	Drives the orbit propagation a specified number of nodes.
NEWPAGE	GTARG TGTGT	Creates a new page in the report file.
OPSFOR	GTARG	Opens a file read only.
OPSFN	GTARG	Creates and opens a new file.
ORB2U	KEP2CAR MNODES	Calculates argument of latitude from the state vector.
ORB2LATLONG	COMPGT	Calculates the geodetic ground track latitude and longitude from the state vector.
PROP	MNODES	Drives the orbit propagation a specified amount of time.
SCREEN_HEADER	TGTGT	Writes the header to the screen during execution.
SUMMRY	TGTGT	Writes a summary of the results to the output file.
SUNORB	FIRSTGUESS LSRGPJ PROP	Calculates the solar ephemeris.
UNIT	CAR2KEP DOMNVR	Converts a vector into a unit vector.
UPCASE	GTARG LSRGPB	Converts a string into all upper case characters.
VMSDATE	GTARG	Interface with VMS Date routine.
WRITE_CURVE	WRITE_LABELED_CURVE TGTGT	Writes a curve to the EZPLOT file.
WRITE_LABELED_CURVE	TGTGT	Writes an annotated curve to the EZPLOT file.
WRITE_LIMITS	TGTGT	Writes ground track limits to the output file.
WRITE_LINE	GTARG	Writes a line to the EZPLOT file.

Table 6.2.
GTARG Functions.

Function Name	Calling Module	Description
DCLOSE	FIRSTGUESS COMPGT	
DCPVAL	JRSMPL2	Evaluates polynomial. Interface emulates MATH77 function of same name.
DOT	CAR2KEP	Vector dot product.
GETCPU	GTARG	Determines CPU used by program.
GOTO_STRING	COMPGT TGTGT SCREEN_HEADER GTARG DISPLAY_LIMITS	Moves cursor to specified screen coordinates.
JRSMPL2	CRMINIT	Evaluates the simplified approximation to the Jacchia-Roberts density model.
LNKTIM	GTARG SCREEN_HEADER	Obtains the time at which the current program was linked.
MA2EA	ORB2U ORBBP	Converts mean anomaly to eccentric anomaly.
ORBBP	FIRSTGUESS PROP	Calculates β' angle from elements.
POLY	GTARG	Evaluates a polynomial to 5th degree.
RNG360	ORB2U GTARG FIRSTGUESS DOMNVR COMPGT	Limits an angle to the range 0° to 360°.
SETCPU	GTARG	Initializes CPU calculation.
SIDANG	COMPGT FIRSTGUESS	Calculates sidereal angle, i.e., right ascension of Greenwich.
USER_DENSITY	JRSMPL2	Template for a user-supplied density function.
VMAREA	FIRSTGUESS PROP	Calculates area using the TOPEX / POSEIDON variable mean area model.
YESNO	GTARG	Converts a logical variable to the string YES (if true) or NO (false).
YESNOSTRING	GTARG	Converts a logical variable to a string.

7. EZPLOT

EZPLOT allows the user to draw a single frame with an arbitrary number of curves on it using PGPLOT.[1] The format is

@EZPLOT *namelist input file*

The namelist input file is composed of a single $ZFRAME namelist (table 7.1) followed by an arbitrary number of $ZLINE namelists (table 7.2). $ZFRAME defines the plot parameters. Each $ZLINE defines a curve, a line, or text on the plot.

Table 7.1
$ZFRAME namelist

Name	Type	Dim	Default	Units	Description
CH	R	1	.75		Character height of PGPLOT[2] characters.
CI	I	1	2		PGPLOT[1] color index.
DEVICE	C*12	1	'/NULL'		PGPLOT[1] plot device. '/IMPRESS' - imagen printer '/TEK' - tektronics graphics monitor '/PS' - postscript printer
OFILE	C*12	1	'EZPLOT.LIS'		EZPLOT output message file.
SUMMARY	L	1	T		If true, a summary of what EZPLOT does is written to OFILE.
TITLE	C*100	1	' '		Title of plot.
VPORT	R	4	4*-1	in	PGPLOT[1] view port coordinates in inches. If not specified, a "standard" size view port is used.
WINDO	R	4	4*0		PGPLOT[1] window boundaries in world coordinates.
XOPT	C*12	1	'BCNST'		X-axis options for call to PGBOX.[1]
XSUB	I	1	0		Number of subdivisions between major ticks on the x-axis. If XSUB=0, PGPLOT[1] will calculate this.
XTICK	R	1	0		Distance between ticks on the x-axis. If XTICK = 0, PGPLOT[1] will calculate.
XTITLE	C*100	1	' '		X-axis title.
YOPT	C*12	1	'BCNST'		Y-axis options for call to PGBOX.[1]
YSUB	I	1	0		Number of subdivisions between major ticks on the y-axis. If YSUB=0, PGPLOT[1] will calculate this.
YTICK	R	1	0		Distance between ticks on the y-axis. If YTICK = 0, PGPLOT[1] will calculate.
YTITLE	C*100	1	' '		Y-axis title.

Valid output devices are described in detail in the PGPLOT users guide.[1]

If DEVICE = '/TEK', then the plot will be automatically generated on the screen. A a tektronics terminal or a terminal with tektronics graphics emulation is required.

[1]Cannell, 1990.
[2]Pearson, 1989.

If DEVICE = '/PS', EZPLOT will generate a postscript file PGPLOT.PSPLOT. If DEVICE = '/IMPRESS', EZPLOT will generate an Imagen file PGPLOT.IMPLOT. These files may then be printed directly to the appropriate graphics printer.

Table 7.2
$ZLINE Namelist

Name	Type	Dim	Default	Units	Description
ANGLE	R	1	0	deg	Angle at which to write TEXT, measured counter-clockwise from the x-axis.
CH	R	1	.75		PGPLOT[1] character height for TEXT.
CI	I	1	2		PGPLOT[1] color index for line and for TEXT.
FJUST	R	1	0		Horizontal justification for TEXT. 0.0 = left justified; 0.5 = centered; 1.0 = right justified.
KX	DP	1	1		Scale the X data by a scale constant.
KY	DP	1	1		Scale the Y data by a scale constant.
NPTS	I	1	0		Number of data points in X and Y arrays.
STYLE	I	1	1		PGPLOT[1] line style. 0 = no line, just plot points 1 = ---------------------------- 2 = - - - - - - - - - - - - - - - - - - 3 = - . - . - . - . - . - . - . - . 4 = ... 5 = - . . . - . . . - . . . - ...
SYMBOL	I	1	-2		Graphics symbol to plot at each point (X,Y)
TEXT	C*100	1	' '		Text to be written at (XTXT, YTXT).
TX	DP	1	0		Translate the x data by a scalar after multiplying by KX.
TY	DP	1	0		Translate the y data by a scalar after multiplying by KY.
X	DP	NPTS	all 0		X-axis data array in world coordinates.
XTXT	R	1	0		X-coordinates of text
Y	DP	NPTS	all 0		Y-axis data array in world coordinates
YTXT	R	1	0		Y-coordinates of text

[1]Pearson, 1989.

References

Bhat, R.S. , Frauenholz, R. B., and Cannell, P. E., "TOPEX/POSEIDON Orbit Maintenance Maneuver Design," AAS 89-408, *AAS/AIAA Astrodynamics Specialists Conference*, Stowe, VT, August 7-10, 1989.

Bhat R. S., "TOPEX/POSEIDON Orbit Acquisition Maneuver Design," AAS 91-514, 1991 *AAS/AIAA Astrodynamics Specialist Conference*, Durango, Colorado, August 19-22, 1991.

Cannell, Eric., Program EZPLOT, in-line documentation, April 4, 1990.

Carlisle, George, A. DiCicco, H. Harris, A. Salama, M. Vincent, *TOPEX/Poseidon Project Mission Plan*, Jet Propulsion Laboratory, JPL D-6862, rev. C, Aug. 1991 (Internal Document).

Cook, G. E., *Perturbations of Near-Circular Orbits by Earth's Gravitational Potential*, Royal Aircraft Establishment Technical Report #65252, Ministry of Aviation, Farnborough Hants, England, 1965.

Cutting, E., Born, G.H., and Frautnick, J.C., "Orbit Analysis for SEASAT," *J. Astronautical Sciences*, **24**, pp. 55-90, Jan-March 1976.

Ekelund, J.E., Sunseri, R.F., and Collier, J.B., *DPTRAJ/ODP User's Reference Manual*, Jet Propulsion Laboratory, JPL D-263, Oct. 1991 (Internal Document).

Escobal, Pedro Ramon, *Methods of Orbit Determination*, New York: Krieger, 1983.

Frauenholz, R. B., and Shapiro, B.E., "The Role of Predicted Solar Activity in TOPEX/POSEIDON Orbit Maintenance Maneuver Design," AAS 91-515, *AAS/AIAA Astrodynamics Specialists Conference*, Durango, CO, August 19-22, 1991.

Groves, G. V. "Motion of a Satellite in the Earth's Gravitational Field," *Proc. Roy. Soc.* **254**, pp. 48-65, 1960.

Guinn, J.R., "Short Period Gravitational Perturbations for Conversion Between Osculating and Mean Orbit Elements," AAS 91-430, *AAS/AIAA Astrodynamics Specialists Conference,* Durango, CO, August 19-22, 1991.

Kaula, William M., "Development of the Lunar and Solar Disturbing Function for a Close Satellite," *Astro. J.*, **67**:3, pp. 300-303, June 1962.

Kozai, Y., "The Motion of a Close Earth Satellite," *Astro. J.*, **64**, pp. 367-377, Nov. 1959.

Long, A.C., Capellari, J.O., Velez, C.E., and Fuchs, A.J., *Goddard Trajectory Determination System (GTDS) Mathematical Theory, Revision 1,* Goddard Space Flight Center Flight Dynamics Division FDD/552-89/001 and Computer Sciences Corporation Technical Report CSC/TR-89/6001, July 1989.

Merson, R.H., *The Dynamic Model of PROP, A Computer Program for the Refinement of the Orbital Parameters of an Earth Satellite*, Royal Aircraft Establishment, Technical Report #66255, Ministry of Aviation, Farnborough Hants, England, Aug. 1966.

Pearson, T.J., *PGPLOT Graphics Subroutine Library* (Users Manual), Jet Propulsion Laboratory, June 1989 (Internal Document).

Roberts, C., "An Analytical Model for Upper Atmosphere Densities Based upon Jacchia's 1970 Models," *Celestial Mechanics*, **4**, pp. 368-377, 1971.

Shapiro, B.E., *TOPEX/POSEIDON Repeat Orbit and Reference Grid Definition*, Jet Propulsion Laboratory Interoffice Memorandum No. 314.5-1642, June 22, 1992 (Internal Document).

Shapiro, B.E.: 1993A, *GTARG Error Models*, Jet Propulsion Laboratory Interoffice Memorandum No. 314.5-1691, April 22, 1993 (Internal Document).

Shapiro, B.E.: 1993B, *Prediction of Ground Tracks at Verification Sites*, Jet Propulsion Laboratory, Interoffice Memorandum No. 314.5-1693, May 5, 1993 (Internal Document).

Shapiro, B.E. and Bhat, R. S., "GTARG - The TOPEX/POSEIDON Ground Track Maintenance Maneuver Targeting Program," AIAA 93-1129, *AIAA Aerospace Design Conference*, Irvine, CA, Feb. 16-19 1993.

Spier, Gerd W., *Design and Implementation of Models for the Double Precision Trajectory Program (DPTRAJ)*, Jet Propulsion Laboratory JPL D-5901, April 1989 (Internal Document).

Squier, D. and Byers, K., *Goddard Trajectory Determination System (GTDS) User's Guide, Revision 2*, Computer Sciences Corporation Technical Report, CSC/SD-851/6738, Dec. 1987.

Vincent, M.A., "The Inclusion of Higher Degree and Order Gravity Terms in the Design of a Repeat Ground Track," AIAA-90-2899-CP, *AIAA/AAS Astrodynamics Conference*, Portland, Oregon, Aug. 1990.