At What True Anomaly Should I Perform a Burn If I Want To Minimize Fuel Usage?

In this exercise you will create a scenario that will help you change the orbit of a satellite that is not at the proper altitude. To do this you will use STK/Astrogator and STK/Analyzer. Upon completion, you will be able to:

- Examine the basics of an Astrogator mission control sequence.
- Adjust the timing of a satellite maneuver to minimize the fuel usage during a raise of the perigee radius (rp) of the satellite
- Use STK/Analyzer to create a carpet plot, analyzing multiple orbit variables at multiple values, at the same time

Problem Statement

A satellite has recently been launched into orbit around the Earth. The launch was not entirely successful and now the orbit is not at the right altitude. It needs to be raised. You will have to raise the perigee radius (rp) of the satellite to 10,000 km using a single burn. This is the distance from the Earth's center to the point of the closest approach to the Earth. You will want to minimize the amount of fuel burned. You will need to find the perfect time to burn while minimizing the change in velocity (ΔV). This will correlate to the amount of fuel used.

Break it Down

Here is what you know:

- You must raise the perigee radius (rp) of the satellite to 10,000 km using a single burn.
- You must minimize the amount of fuel used.
- You want to burn in the direction of the satellite's motion, in the direction of velocity.
- You have to find the True Anomaly value that tells you when the burn should occur.
- True Anomaly is measured in degrees in the direction of the satellite's motion, from the perigee radius (rp) to the position of the satellite.

Solution

Build a scenario that will help you determine when to perform the satellite burn. You will first design a maneuver that the satellite can make, adjusting its orbit from one to another. You will use *Astrogator* with the use of its new optimizing routine to determine the true anomaly (v) value to optimize your burn. You will then use *STK/Analyzer's Carpet Plot* to determine what influence small variations from the optimal will have on the fuel requirement, as well as which parameters affect the fuel usage more drastically.

Open an Existing Scenario

To speed things up and allow you to focus on the portion of this exercise that uses the *Astrogator* propagator, a partially developed scenario has been provided for you. This scenario can be found in the student files area.

1. Ensure that the *Welcome to STK!* dialog is visible in the *STK Workspace*.
2. Click the *Open a Scenario* button.
3. Locate the student files provided for this exercise (C:\Training\STK).
4. Select *WhatsGator.vdf*.

When you select an object in the *Open* dialog, information about that object will display in at the bottom of the panel.

5. Click *Open*.

When you open the scenario, a directory with the same name as the scenario will be created in the default user directory (C:\My Documents\STK 9) The scenario will not be saved automatically.

6. When the scenario loads, click *Save* ().

WHAT'S IN MY SCENARIO?

When the scenario loads, you will notice that your scenario contains one predefined satellite. You will also notice that there are two *2D Graphics* windows. One of the *2D Graphics* windows looks different from what you've seen thus far. That's because the default *2D Graphics* view uses a *Equidistant Cylindrical* projection; however, one of the windows here is using an *Orthographic* projection. An orthographic projection depicts a hemisphere of the Earth as it would appear from deep space. This is often helpful when viewing orbit segments used in *Astrogator*.

1. Open the *2D Graphics* window properties ().
2. Select the *Basic - Projection* page.
3. Ensure the *Type* is set to *Orthographic*.
4. Click *Cancel*.

The satellite that was provided for you is using an *Astrogator* propagator. Let's take a look at the *Astrogator* propagator.

5. Open *Satellite*'s () properties ().
6. Select the *Basic - Orbit* page.
7. Notice that the *Propagator* selected is an *Astrogator* propagator.

What Is Astrogator?

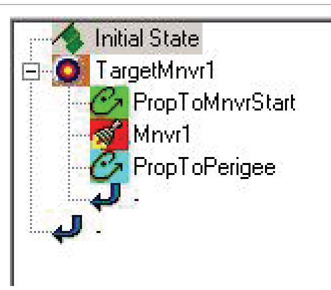
STK/Astrogator is an interactive orbit maneuver and space mission planning tool for use by spacecraft operations and mission analysis staff that offers wide flexibility through the use of customized thrust models, finite and impulsive maneuvers, and the ability to solve for solutions with a differential corrector targeter. You can use *Astrogator* for a variety of space mission analyses, such as:

- For formation flying, rendezvous planning, constellation design, space-based intercept.
- For interplanetary, lunar, and libration point trajectories.
- For GEO, LEO, HEO, Sun-Sync orbit maintenance requirements.
- For automated planning of event-driven maneuvers.
- For Monte Carlo and other script driven analyses.
- To incorporate fully customizeable force, engine, and atmospheric models.
- For high, low, and variable-thrust trajectories.

Mission Control Sequence

One of the first things that you will notice on the *Astrogator* propagator is the *Mission Control Sequence* (MCS). The *MCS* is the core of your space mission scenario. The *MCS* functions as a graphical programming language, utilizing mission segments that dictate how *Astrogator* will build the trajectory of the spacecraft.

FIGURE 1. Mission Control Sequence



By adding, removing, rearranging, and editing *MCS Segments*, you can define a mission of any desired level of complexity. The *MCS* is represented schematically by a tree structure appearing in the left pane of the *Orbit* page of the satellite's basic properties.

Let's take a closer look at the *MCS* options in your scenario. This will give you an idea of what each of the mission control segments contain.

Initial State

Initial state represents your starting position. In the *Initial State* () segment, you can select a coordinate system, enter orbital elements of several different types, and configure the spacecraft's physical values.

1. Locate the *MCS* tree under the *MCS* controls.
2. Select the *Initial State* () segment.

The name of the currently selected coordinate system is displayed in a read-only field. The *Initial State* segment lets you select from among the following element types:

TABLE 1. Coordinate systems

TYPE	DESCRIPTION
Cartesian	Specifying an orbit by three position elements and three velocity elements in a rectangular coordinate system.
Keplerian	The classical system, specifying an orbit by six elements describing its size, shape and three-dimensional orientation in space.
Modified Keplerian	A modification of the classical system, using radius of periapsis instead of semimajor axis. Recommended for very elliptic or hyperbolic orbits.
Spherical	A system in which positions are specified as a radial distance form the origin and two angles relative to a fundamental plane.
Target Vector Incoming Asymptote	Used for hyperbolic arrival trajectories.
Target Vector Outgoing Asymptote	Used for hyperbolic departure trajectories.

You know that you want to raise the periapsis to 10,000 km.

- Is the current periapsis less than 10,000 km?

FUEL MASS

Let's take a look at the fuel mass before the maneuver.

1. Select the *Fuel Tank* tab.
2. Make a note of the *Fuel Mass* value.

This value represents the original fuel mass. The fuel mass will change after the maneuver based on the fuel used for the maneuver.

Target Sequence

Now that our initial state is clear to us, let's look at what we're asking *Astrogator* to do. The *Target Sequence* ( ) segment provided here gives Astrogator the "target" or goal of a sequence of events for a satellite.Within this Target, you can set the goal as a value condition or an Optimization.

1. Expand(+) *TargetMnvr1* ( ).

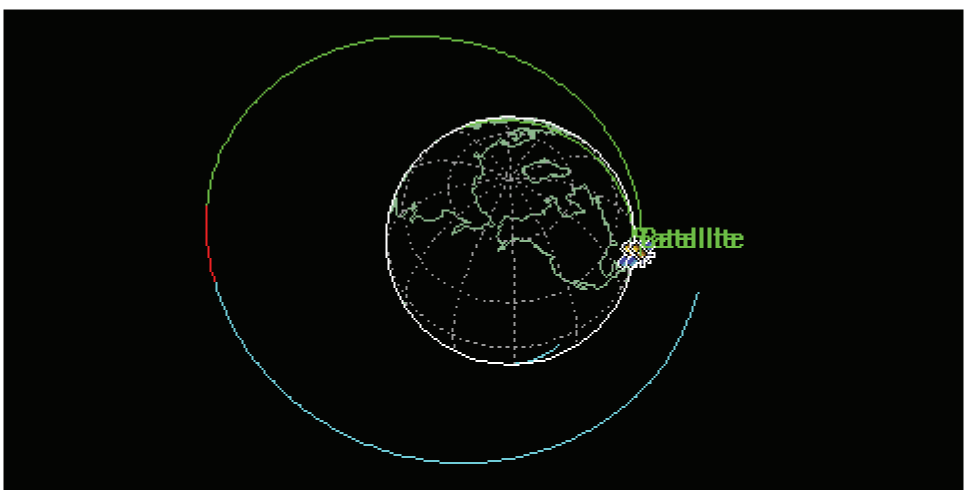
TargetMnvr1, is made up of three separate components: a propagate segment *ProptoMnvrStart* ( ), a maneuver segment *Mnvr1* ( ) and a second propagate segment *PropToPerigee* ( ).

1. Take a look at each segment.

- What color is the background of each segment's icon?

2. Bring the *Orthographic 2D Graphics* window to the front.

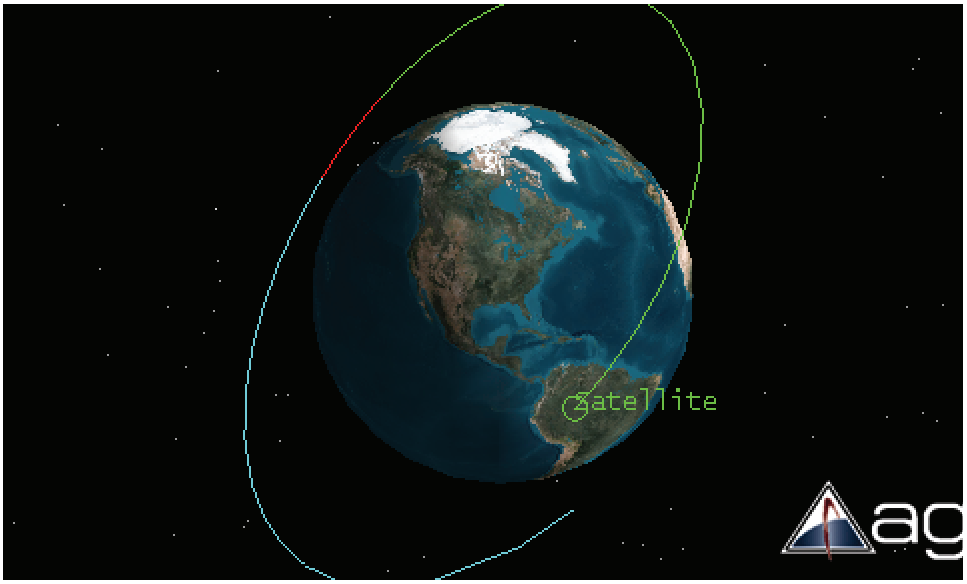
FIGURE 2. 2D View: Mission control sequence display



The color of each segment of the orbit coincides with the color of the segment in the *Target Sequence* propagator.

3. Bring the *3D Graphics* window to the front.

FIGURE 3. 3D View: Mission control sequence display



Notice that in 3D the orbit is also colored to match the segments in the *Mission Control Sequence.*

Let's look at each of the three segments that make up the *TargetMnvr1* (◉) sequence provided here, in more detail.

VARIABLES

The targeting profile sets up the goal (target) for the *Astrogator* maneuver. Let's see an example of one.

1. Select *TargetMnvr1* (◉).
2. Click *Properties* buton in the *Profiles* area.

At the top of the *TargetMnvr1* window there are *Control Parameters.* The selected *Control Parameter* includes a final value because this *Astrogator* run has previously been initiated. When you set up your mission control sequence, this data would not exist! It would be nice if you could see *Astrogator* actually compute the maneuver for you. You can!

3. Look at the *Equality Constraints* (*Results*) section.

A desired *RMag* value of 10,000 km is set for the *PropToPerigee* segment. By telling *Astrogator* to use this value, we're defining our desired outcome.

Astrogator will begin the maneuver at apogee, using the engine we specified and will burn until we raise our perigee radius (rp) to 10,000km. We don't know long the burn will take or what the resulting fuel used will be. *Astrogator* will determine these.

4. When you finish, click *Cancel* to dismiss the *Targeting Profile* window and return to the *Astrogator* propagator.

PROPAGATE

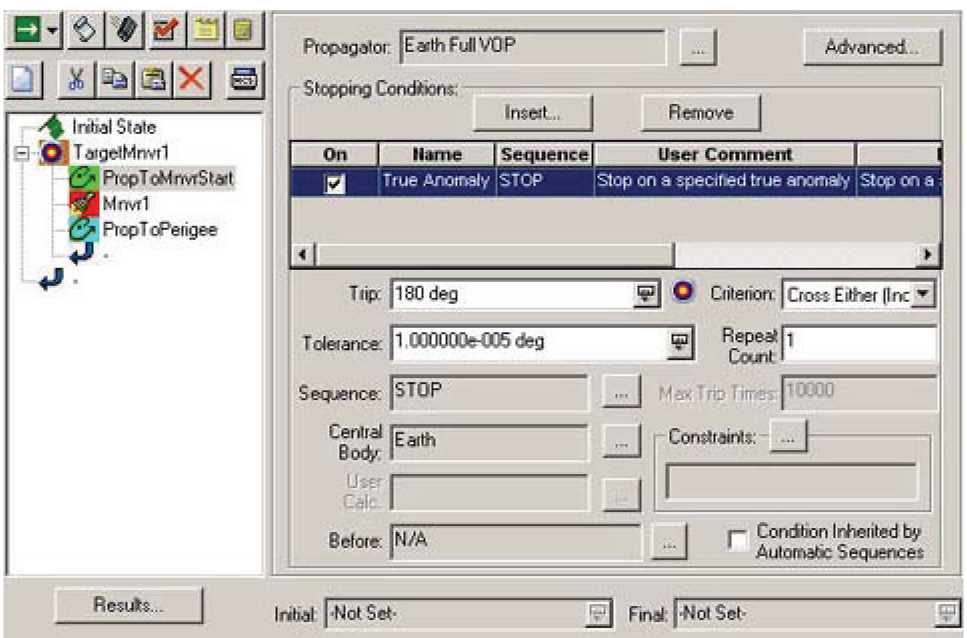
Propagation of an orbit is handled by the *Propagate* ( ) segment, the central feature of which is a mechanism for defining one or more conditions for stopping the propagation or initiating a automatic sequence. *ProptoMnvrStart* ( )(the green portion of the orbit) takes the initial orbit conditions and propagates it to the point where the maneuver begins.

Let's take a closer look at the first Propagate segment.

1. Select *ProptoMnvrStart* ( ) in the *MCS* tree.

When you select a segment in the *MCS* tree, its properties display in the panel to the right.

FIGURE 4. PropToMnvrStart properties



In this segment, the propagation is set to *STOP* at a specified value for true anomaly. The desired value for the stopping condition is specified in the *Trip* field. This tells *Astrogator* to propagate the initial orbit until the true anomaly is 180 degrees (or the satellite is at its apogee point).

2. Click the target beside the *Trip* value( ) to enable the stopping condition. Doing so will tell Astrogator to use this as a target stopping condition.
3. Take a look at the *3D Graphics* window or the *Orthograhic 2D Graphics* window.

- Does the green orbit segment end at the satellite's apogee?

Once we have arrived at 180 degrees, we will have *Astrogator* perform a maneuver for us. Let's take a look at what information will be considered when calculating this maneuver.

MANEUVER

Astrogator provides two basic types of *maneuvers* ( ) -- *impulsive* and *finite* -- for use in constructing your space mission scenario. Each maneuver segment must be defined as either an impulsive or finite maneuver. The *impulsive* maneuver segment models a maneuver as if it takes place instantaneously and without any change in the position of the spacecraft. This is the classic velocity (ΔV). In an *impulsive* maneuver, the final state vector is calculated by applying the specified velocity (ΔV) vector to the initial velocity vector (respecting the coordinate system). On the other hand, a *finite* maneuver segment takes into account changes that occur throughout the duration of the maneuver by numerically integrating the effect of the acceleration from the engine. We will be modeling a finite maneuver.

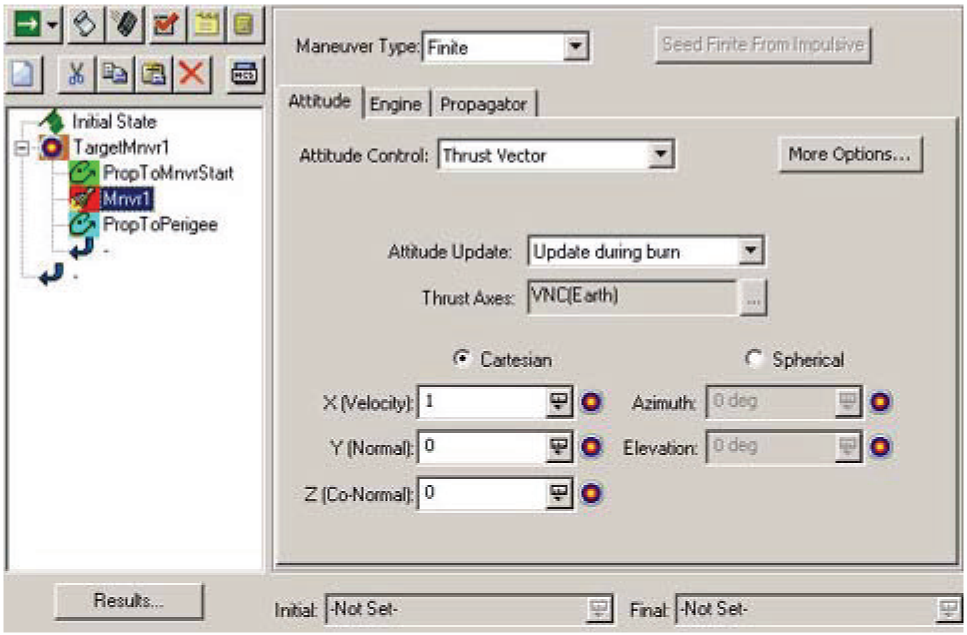
1. Select *Mnvr1* ( ).

Notice there are three properties categories associated with this segment: *Attitude*, *Engine*, and *Propagator*. Let's look at the *Attitude* properties first.

2. Ensure that the *Attitude* tab is selected.

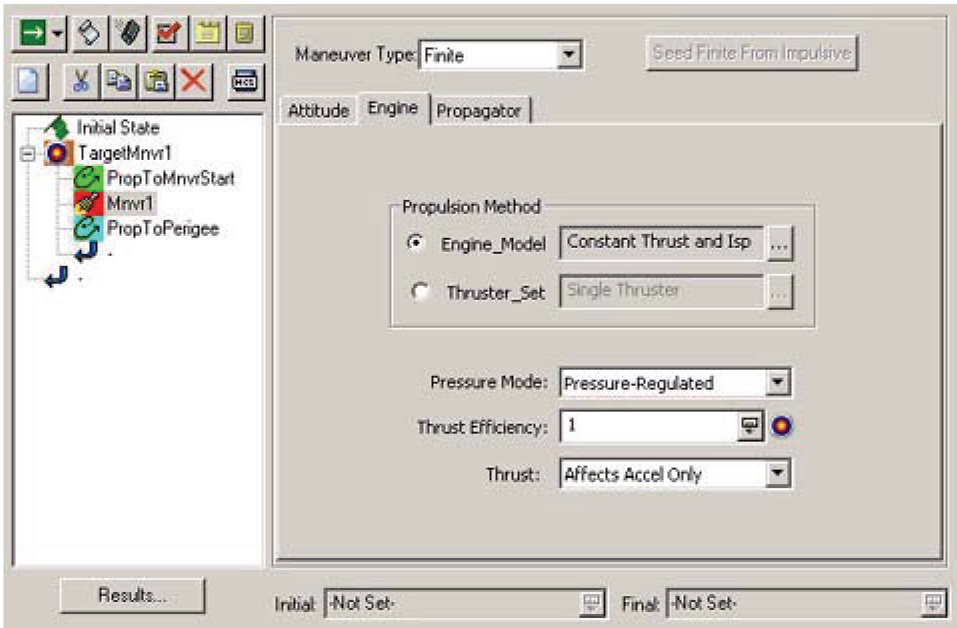
The options on the *Attitude* tab are specific to the *Attitude Control* selected, so the attitude parameters that you must specify will depend on the *Attitude Control* selected. The *Thrust Vector Attitude Control* is selected because in this exercise, you need to perform the burn in the direction of the satellite's motion. You can select the total thrust vector in cartesian or spherical form with respect to the thrust axes. The *Thrust Vector* for this maneuver is along the satellites's X (Velocity) vector (indicated by the number one beside that option).

FIGURE 5. Mnvr1 Attitude properties



3. Now, select the *Engine* tab.

FIGURE 6. Mnvr1 Engine properties



The options on the Engine tab allow you to select an engine model or a thruster set. This maneuver uses an engine model with a constant thrust and specific impulse (*Constant Thrust and Isp*) for this maneuver.

- But what are the thrust and specific impulse values that will be used?

The specific values used to define components in the *Mission Control Sequence* can be found in the *Astrogator Component Browser*.

THE ASTROGATOR COMPONENT BROWSER

The *Astrogator Component Browser* is a powerful tool that enables you to redefine components of your space mission analysis and create new ones. The components are organized into groups listed in a tree structure.

1. Select the *Component Browser* option from the *Utilities* menu in the *STK Workspace* to open the component browser.
2. Take a look at the components in the *Component Browser*.

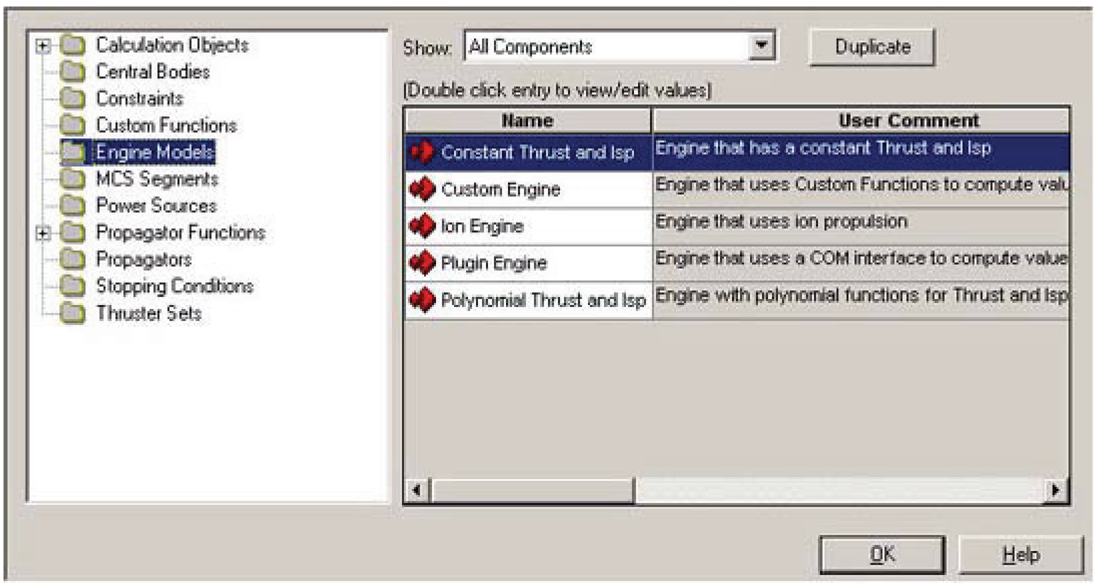
The components are organized into groups listed in a tree structure in the left pane of the component browser. Individual components in a given group or subgroup are displayed in the right pane when you click the corresponding folder or subfolder in the left pane.

The show option allows you to filter the display of components. You can restrict the display so that only user-created, hard-coded, all components.

3. Select *Engine Models* in the *Component Tree.*

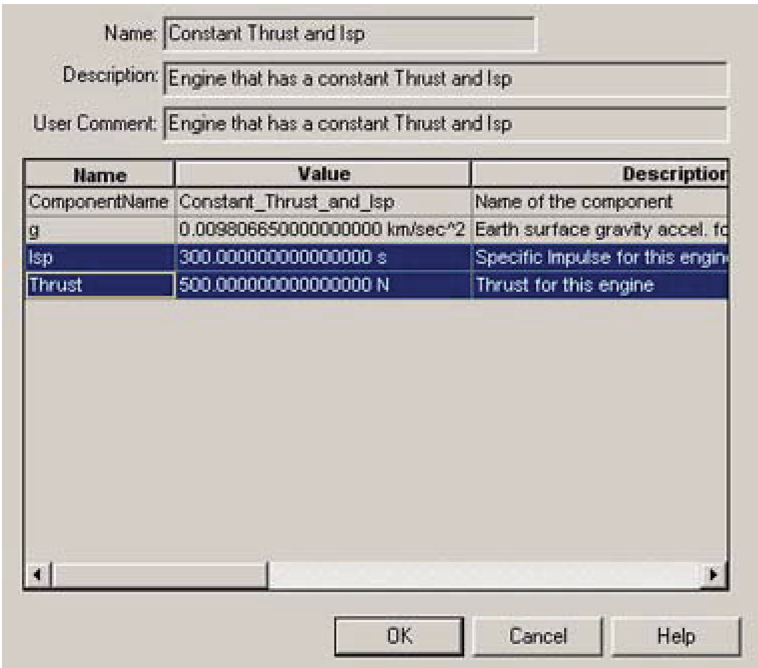
When you select *Engine Model* in the tree, all of the available components will display on the right side of the window.

FIGURE 7. Constant Thrust and Isp Engine Model components



4. Double-click the *Constant Thrust and Isp* component.

FIGURE 8. Constant Thrust and Isp Engine Model properties



The dialog that appears will outline the characteristics of the selected engine model. If the default values differ from the desired values, you can duplicate the component and adjust them.

- What is the specific impulse for this engine?
- What is the thrust used for this engine model?

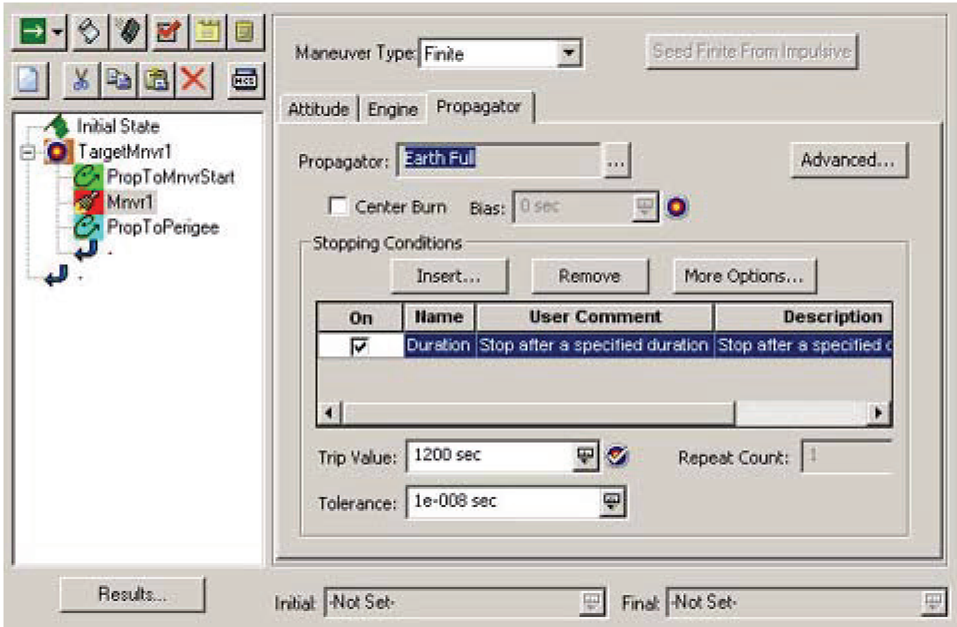
5. Click *OK* to close the characteristics of this engine model.
6. Click *OK* again to close the *Component Browser*.

PROPAGATOR

Once you define the engine characteristics and the thrust vector, you need to give *Astrogator* a starting point initial guess for how long to propagate using this maneuver model.

1. Finally, select the *Propagator* tab.

FIGURE 9. Mnvr1 Propagator properties



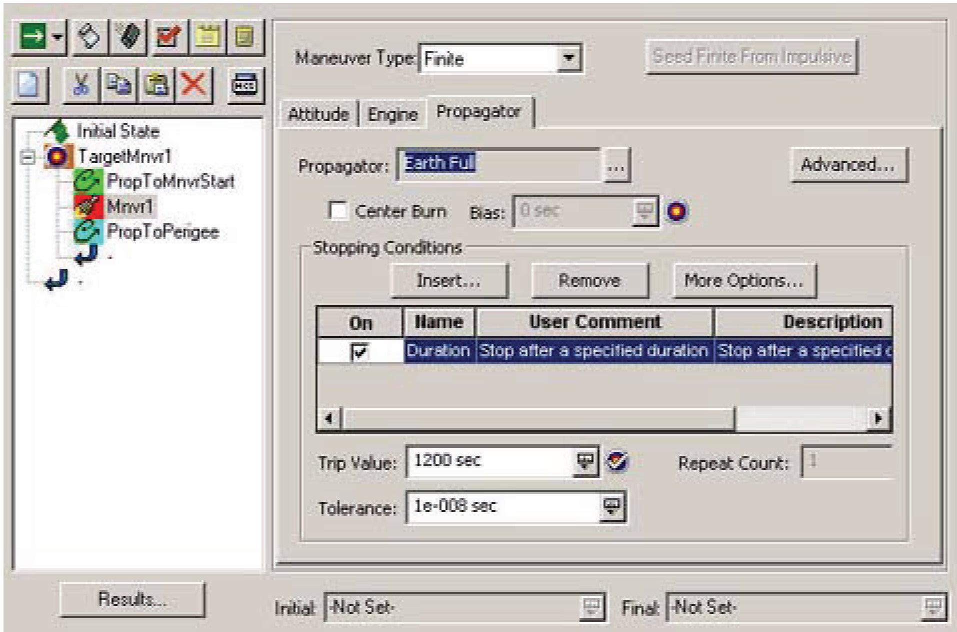
The options on the *Propagator* tab tell *Astrogator* to implement the maneuver burn for a specified time period. Let's start with the *Trip Value*. The trip value allows you to set the desired value for the selected stopping condition, i.e. the value to be achieved in order for the condition to be deemed satisfied.

- What is the current Trip Value?
- What does a trip value of 1200 seconds tell Astrogator to do?

In this case, the Trip Value indicates the duration of the burn. Setting the trip value tells *Astrogator* to burn for 1200 seconds in order to attempt to raise the perigee radius (rp) to 10,000 km. We don't know how long the burn will take. We want *Astrogator* to determine that, so we entered an initial value--a "ballpark" figure.

Notice that the target icon beside the trip value option has a check mark through it. By checking the target value, you are telling *Astrogator* that it can vary this value in the *Target Sequence* as needed in order to achieve your target, that is, in order to raise the perigee radius (rp) to 10,000 km by using the engine we have specified and by beginning the burn at apogee. In a moment we will see what value *Astrogator* computed.

FIGURE 10. Mnvr1 Propagator properties

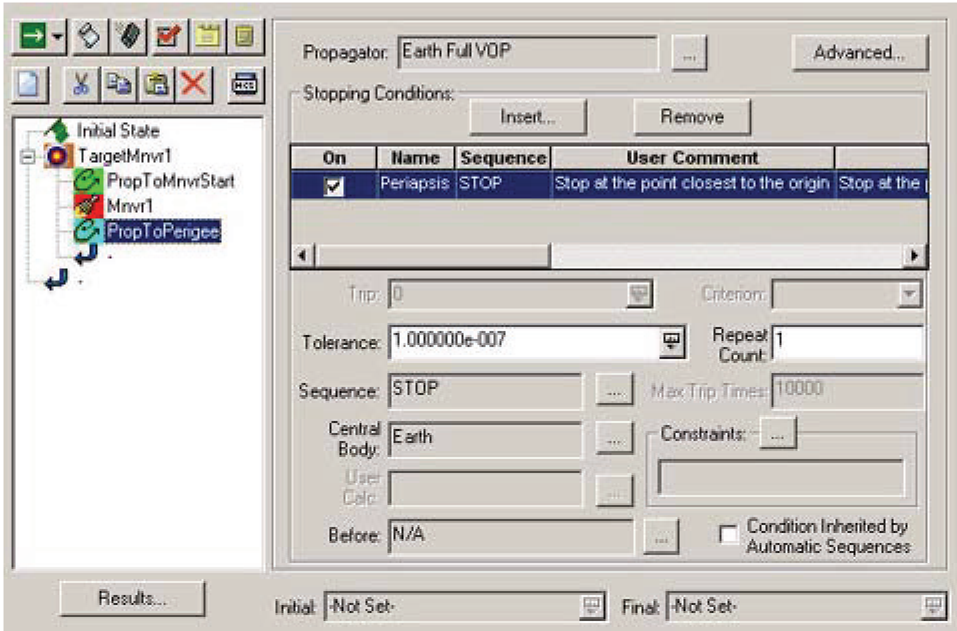


PROPAGATE

Now that we have set up the criteria for the maneuver, we need to tell STK to propagate the orbit once again following this burn. That is exactly what the last segment of the MCS, the *PropToPerigee* ( )segment, does.

1. Select the *PropToPerigee* ( )segment.

FIGURE 11. PropToPerigee properties



In this segment, the propagation is set to *STOP* this propagation when perigee is reached. We, of course, would like verification that our maneuver achieved its purpose--that the perigee radius (rp) is 10,000 km.

- How will you tell Astrogator that this requirement must be met?

Defining this requirement is a two part process. The first part of the process tells *Astrogator* when to stop propagating the orbit. In this case, you want to stop propagation at perigee, which would be the most logical stopping point since you want to raise your perigee radius.

The second part tells *Astrogator* what value(s) you want to report on when propagation stops. In this case, you are interested in the magnitude of the radius at the stopping point (perigee), *RMag*, and fuel used. For the first value, *RMag* you are setting the desired value. You want *Astrogator* to use this value as a factor in determining the maneuver. The second value, fuel used, is a dependant value that will depend on the maneuver that *Astrogator* computes. You're not setting a fuel use requirement. You are instead interested in the resulting value.

2. Click on the *Results*… button below the MCS tree.

The *Results*... button opens the *User-Selected Results* window where you can select components to include in the summary report ( )for the currently selected segment. These results will be displayed at the end of the summary report under the *User-selected results* header.

3. Notice that the *RMag* components is selected here.
4. Click *OK* to close the *User-Selected Results* window.

Target Maneuver Variables

Let's go back and look at the *Target Maneuver* ( ) component. We previously looked at the Targeting Profile that was not set up to run a target of 10,000 km RMag. We didn't tell if it should optimize the True Anomaly and fuel used while still hitting that goal. Let's create a new Targeting Profile to accomplish all of that.

1. Select *TargetMnvr1* ( ) in the *MCS* tree.

OPTIMIZE FOR FUEL

You want to optimize the location of your burn to optimize fuel. You will tell STK to adjust the true anomaly to optimize the fuel. Let's do that now.

1. Click *New*... in Profiles.
2. Select the *Design Explorer Optimizer* profile.
3. Click *OK*.
4. Select the *Properties* of the profile. You can choose variables that the *Astrogator Optimizer* should adjust and their ranges.
5. Enable the following stopping conditions

TABLE 2. Design explorer optimizer control parameters

CONTROL PARAMETER	STATE	LOWER BOUNDS	UPPER BOUNDS
StoppingConditions.True_Anomaly.TripValue	On	160 deg	180 deg
FiniteMnvr.StoppingConditions.Duration.TripValue	On	900 sec	1500 sec

6. Keep all other defaults.

The uppper and lower bounds for true anomaly represent a range of true anomaly values. The anticipated result of 180 degrees. The duration values represent a range around our desired stopping condition of 1200 sec.

7. Locate the Objectives and Constraints (Results) area:
8. Enable the following:
9.

TABLE 3. Design explorer optimizer control parameters

CONTROL PARAMETER	STATE	LOWER BOUNDS	UPPER BOUNDS	GOAL
Fuel Used	On	0 kg	500 kg	Minimize
R_Mag	On	10,000 km	10,000 km	Constraint

These fuel used bounds represent the minimum (0 kg) and maximum amount of fuel that can be used. You want to use fuel in this range.

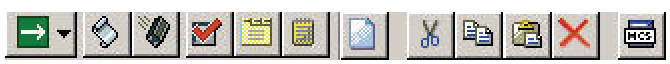
For the R-Mag, you have a specific singular goal of 10,000 km. You need to set both the lower and upper bound to 10,000 km so you reach that radius.

10. Click OK.
11. Select the first Targeting Profile.
12. Click the Remove button in the Profiles area. You no longer need it.

MCS Controls

The *Astrogator* propagator includes a full set of controls that can be used for inserting, deleting, copying, and editing segments.

FIGURE 12. Mission Control Sequence (MCS) controls



RUN!

Let's use those controls to run the *MCS* and then generate summary data.

1. Click the *Reset* button in the *Profiles and Corrections* section.
2. Click *Run* ( ).

When you run the mission control sequence, a status dialog appears. The status of each iteration of the run until covergence or failure displays in this status dialog. Use the information in the status box to answer the following questions:

- Did your solution converge?
- If so, how many iterations did it take?

Take a look at the resulting panel. Find the *True Anomaly* trip value that is optimized for this scenario.

- What is the value?
- When does this occur?
- How much fuel is used?
- Do you meet your goals?

3. When you finish, close the status dialog.

SUMMARY

The maneuver is computed and you now know where to burn to minimize fuel lost. Let's create a summary report that will give us the resulting fuel used.

1. Select *TargetMnvr1* (◎) in the *MCS* tree.
2. Click the *Summary* (◈) button.
3. Scroll down to the section titled *MCS Segment Type: Maneuver: Finite.*
4. Scroll down to *Maneuver Summary.*
5. Note the *Fuel Mass* value.

- How much fuel was used for the maneuver?
- How much fuel mass remains?

6. When you finish, close the *Summary* report.

SWITCH PROFILES

Since you will not need the optimizer within Astrogator. Let's delete that from Astrogator and reformulate Astrogator to let us vary the radius and True Anomaly for the next round.

1. Select *TargetMnvr1* (◎) in the *MCS* tree.
2. Select the *Design Explorer Optimizer* profile.
3. Click *Remove.*
4. Click *Yes* when the confirmation message appears.
5. Click *New* to add a new profile.
6. Select Differential Corrector.
7. Click OK.

This will let you set the two variables you want to change externally.

1. Select the *Differential Corector* profile.
2. Click *Properties.*
3. Select *Finite Mnvr.Stopping.Condition.DurationTripValue* as the *Control Parameter* and the R_Mag as a constraint.
4. Set the desired value for R_Mag to 10,000 km.
5. Click OK.
6. Click OK to dismiss the *Orbit* page.

Carpet Plot

You have determined your solution for raising the perigee radius (rp) to 10,000 km, but now your curious about what the impact on fuel mass would be if you tweaked the resulting perigee radius (rp). Let's study the effects of both true anomaly and final desired radius on fuel usage. You will vary the values of the true anomaly and the radius of final orbit and see how it affects the remaining fuel using a *Carpet Plot* analysis.

This analysis can be done externally to STK with the help of STK/Analyzer.

1. Select the scenario in the *Object Browser.*
2. Select *Analyzer* from the *Analysis* menu.
3. Click the *Carpet Plot* link.

Now let's set up your carpet plot. We will be varying two variables, true anomaly and perigee radius. Let's set those up as the Input Variables first.

Input Variables

Select your first input variable, true anomaly.

1. Expand the component tree as follows:

   - ... *Scenario*
     - ... *Satellite*
       - ... *Propagator*
         - ... *TargetMnvr1*
           - ... *SegmentList*
             - ... *PropToMnvrStart*
               - ... *StoppingConditions*
                 - *true_Anomaly*

2. Drag the *true_Anomaly* variable and drop it in the design variable () box on the right side of the *Analyzer* window.

Your previous analysis says that the minimum fuel usage occurred at a true anomaly approximately 173 degrees. Let's make that the mid point of our value range for this analysis.

3. Set the following values below the design variables box:

TABLE 4. Semi-major axis design variable values

OPTION	VALUE
From	170
To	175
Num Steps	3
Step Size	Analyzer will calculate this value.

This will vary true anomaly over a smaller range, but you're also wondering if burning to a radius of 10,000 km is the best option. Would 9,000 km or 11,000 km be a better radius? Let's check the range between 9,000 km to 11,000 km.

4. Expand the component tree as follows:
   - ... *Scenario*
     - ... *Satellite*
       - ... *Propagator*
         - ... *TargetMnvr1*
           - ... Differential_Corrector
             - ... *PropToPerigee___R_Mag*
               - *desired*
5. Drag the desired variable and drop it in the design variable () box on the right side of the *Analyzer* window.
6. Set the following values below the design variables box:

TABLE 5. Semi-major axis design variable values

OPTION	VALUE
From	9,000
To	11,000

TABLE 5. Semi-major axis design variable values

OPTION	VALUE
Num Steps	5
Step Size	Analyzer will calculate this value

Now, let's tell Analyzer what data to return.

RESULTS

Now you need to tell *Analyzer* to tell you what the fuel mass value is at each of those steps.

7. Expand the component tree as follows:
   - ... *Scenario*
     - ... *Satellite*
       - ... *Propagator*
         - ... *TargetMnvr1*
           - ... *SegmentList*
             - ... *Mnvr1*
               - ... *Results*
                 - *fuel_Used*
8. Drag the *fuel_Used* variable and drop it in the *Responses* ( ) box on the right side of the *Analyzer* window.
9. Click Run...

Examining the Results

On the left side of the *Data Explorer* window, you will see several graphs and charts. The table page shows every point that was generated and the values generated for each. There is a general carpet plot of how the values of true anomaly and radius magnitude both increase over time. This increase causes fuel usage to increase as well.

The *Data Explorer* window also includes several 2D and 3D carpet plot graphs. Let's look at the surface plot.

1. Select the *Plot Variable:response2* graph.
2. Maximize ( ) the *Data Explorer* window.

A couple of the axes on the surface plot are very difficult to read.

Adjust the Graph Display

Let's make some adjustments to the markers along the different axes so that the graph will be more useful.

1. Click the *Basic Settings* ( ) button.
2. Click the *Axes* tab.
3. Set the following:

TABLE 6. Graph display options

Axis	Option	Value
X Axis: true_anomaly (deg)	Automatic	Off
	Increment	0.5
Z Axis: Desired (km)	Automatic	Off
	Increment	200

4. Click OK.

Now it is much easier to see all of the increments in radius increase. You can see that even setting your true anomaly optimally, the greatest effect on fuel usage is the perigee radius (rp).

Manipulate the Graph

The 3D graph display includes a set of controls that let you manipulate its display in the Data Explorer. You can use the controls to rotate, move, zoom, or even add depth to 3D graphs.

FIGURE 13. Analyzer graph controls

1. Click the *Rotate* ( ) button on the graph control toolbar.
2. Use the mouse to drag the graph in the direction that you want to rotate it.
3. Try using some of the other controls to manipulate the graph.
4. When you finish, close the *Data Explorer* (*Carpet Plot*) window.
5. Close the Carpet Plot Tool window.

Save Your Work

1. Save (💾) your work.
2. Close the scenario ( ) without saving.
3. Close *STK* ( ).