Simple Orbit Transfer

Audience	Beginner
Length	30 minutes
Prerequisites	Complete *Simulating an Orbit*
Script File	`Tut_SimpleOrbitTransfer.script`

Objective and Overview

> **Note**
>
> One of the most common problems in space mission design is to design a transfer from one circular orbit to another circular orbit that lie within the same orbital plane. Circular coplanar transfers are used to raise low-Earth orbits that have degraded due to the effects of atmospheric drag. They are also used to transfer from a low-Earth orbit to a geosynchronous orbit and to send spacecraft to Mars. There is a well known sequence of maneuvers, called the Hohmann transfer, that performs a circular, coplanar transfer using the least possible amount of fuel. A Hohmann transfer employs two maneuvers. The first maneuver raises the orbital apoapsis (or lowers orbital periapsis) to the desired altitude and places the spacecraft in an elliptical transfer orbit. At the apoapsis (or periapsis) of the elliptical transfer orbit, a second maneuver is applied to circularize the orbit at the final altitude.

In this tutorial, we will use GMAT to perform a Hohmann transfer from a low-Earth parking orbit to a geosynchronous mission orbit. This requires a targeting sequence to determine the required maneuver magnitudes to achieve the desired final orbit conditions. In order to focus on the configuration of the targeter, we will make extensive use of the default configurations for spacecraft, propagators, and maneuvers.

The target sequence employs two velocity-direction maneuvers and two propagation sequences. The purpose of the first maneuver is to raise orbit apoapsis to 42,165 km, the geosynchronous radius. The purpose of the second maneuver is to nearly circularize the orbit and yield a final eccentricity of 0.005. The basic steps of this tutorial are:

1. Create and configure a **DifferentialCorrector** resource.
2. Modify the **DefaultOrbitView** to visualize the trajectory.
3. Create two **ImpulsiveBurn** resources with default settings.
4. Create a **Target** sequence to (1) raise apoapsis to geosynchronous altitude and (2) circularize the orbit.
5. Run the mission and analyze the results.

Configure Maneuvers, Differential Corrector, and Graphics

For this tutorial, you'll need GMAT open, with the default mission loaded. To load the default mission, click **New Mission** () or start a new GMAT session. We will use the default configurations for the spacecraft (**DefaultSC**), the propagator (**DefaultProp**), and the two maneuvers. **DefaultSC** is configured by default to a near-circular orbit, and **DefaultProp** is configured to use Earth as the central body with a nonspherical gravity model of degree and order 4. You may want to open the dialog boxes for these objects and inspect them more closely as we will leave them at their default settings.

Create the Differential Corrector

The **Target** sequence we will create later needs a **DifferentialCorrector** resource to operate, so let's create one now. We'll leave the settings at their defaults.

1. In the **Resource** tree, expand the **Solvers** folder if it isn't already.
2. Right-click the **Boundary Value Solvers** folder, point to **Add**, and click **DifferentialCorrector**. A new resource called **DC1** will be created.

Modify the Default Orbit View

We need to make minor modifications to **DefaultOrbitView** so that the entire final orbit will fit in the graphics window.

1. In the **Resource Tree**, double-click **DefaultOrbitView** to edit its properties.
2. Set the values shown in the table below.

Table 1. DefaultOrbitView settings

Field	Value
Solver Iterations, under **Drawing Option**	**Current**
Axis, under **View Up Defintion**	**X**
View Point Vector boxes, under **View Definition**	**`0`**, **`0`**, and **`120000`** respectively

3. Click **OK** to save these changes.

Create the Maneuvers.

We'll need two **ImpulsiveBurn** resources for this tutorial, both using default values. Below, we'll rename the default **ImpulsiveBurn** and create a new one.

1. In the **Resources** tree, right-click **DefaultIB** and click **Rename**.
2. In the **Rename** box, type **`TOI`**, an acronym for Transfer Orbit Insertion, and click **OK**.
3. Right-click the **Burns** folder, point to **Add**, and click **ImpulsiveBurn**.
4. Rename the new **ImpulsiveBurn1** resource to **`GOI`**, an acronym for Geosynchronous Orbit Insertion.

Configure the Mission Sequence

Now we will configure a **Target** sequence to solve for the maneuver values required to raise the orbit to geosynchronous altitude and circularize the orbit. We'll begin by creating an initial **Propagate** command, then the **Target** sequence itself, then the final **Propagate** command. To allow us to focus on the **Target** sequence, we'll assume you have already learned how to propagate an orbit to a desired condition by working through the *Simulating an Orbit* tutorial.

Configure the Initial Propagate Command

1. Click on the **Mission** tab to show the **Mission** tree.
2. Configure **Propagate1** to propagate to **`DefaultSC.Earth.Periapsis`**.
3. Rename **Propagate1** to **`Prop To Periapsis`**.

Create the Target Sequence

Now create the commands necessary to perform the **Target** sequence. Figure 1 illustrates the configuration of the **Mission** tree after you have completed the steps in this section. We'll discuss the **Target** sequence after it has been created.

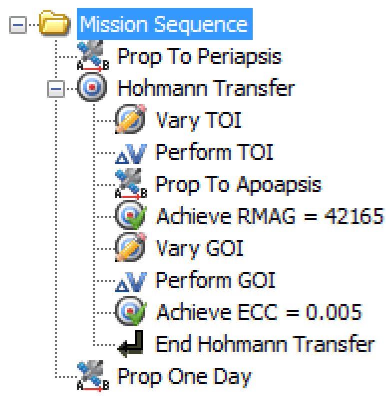

Figure 1. Final Mission Sequence for the Hohmann Transfer

To create the **Target** sequence:

1. In the **Mission** tree, right-click **Prop To Periapsis**, point to **Insert After**, and click **Target**. This will insert two separate commands: **Target1** and **EndTarget1**.
2. Right-click **Target1** and click **Rename**.
3. Type **`Hohmann Transfer`** and click **OK**.
4. Right-click **Hohmann Transfer**, point to **Append**, and click **Vary**.
5. Rename **Vary1** to **`Vary TOI`**.
6. Complete the **Target** sequence by appending the commands in Table 2.

Table 2. Additional Target Sequence Commands

Command	Name
Maneuver	`Perform TOI`
Propagate	`Prop To Apoapsis`
Achieve	`Achieve RMAG = 42165`
Vary	`Vary GOI`
Maneuver	`Perform GOI`
Achieve	`Achieve ECC = 0.005`

Note

Let's discuss what the **Target** sequence does. We know that two maneuvers are required to perform the Hohmann transfer. We also know that for our current mission, the final orbit radius must be 42,165 km and the final orbital eccentricity must be 0.005. However, we don't know the size (or ΔV magnitudes) of the maneuvers that precisely achieve the desired orbital conditions. You use the **Target** sequence to solve for those precise maneuver values. You must tell GMAT what controls are available (in this case, two maneuvers) and what conditions must be satisfied (in this case, a specific orbital radius and eccentricity). You accomplish this using the **Vary** and **Achieve** commands. Using the **Vary** command, you tell GMAT what to solve for—in this case, the ΔV values for **TOI** and **GOI**. You use the **Achieve** command to tell GMAT what conditions the solution must satisfy—in this case, the final orbital conditions.

Create the Final Propagate Command

We need a **Propagate** command after the **Target** sequence so that we can see our final orbit.

1. In the **Mission** tree, right-click **End Hohmann Transfer**, point to **Insert After**, and click **Propagate**. A new **Propagate3** command will appear.
2. Rename **Propagate3** to **`Prop One Day`**.
3. Double-click **Prop One Day** to edit its properties.
4. Under **Condition**, replace the value **`12000.0`** with **`86400`**, the number of seconds in one day.
5. Click **OK** to save these changes.

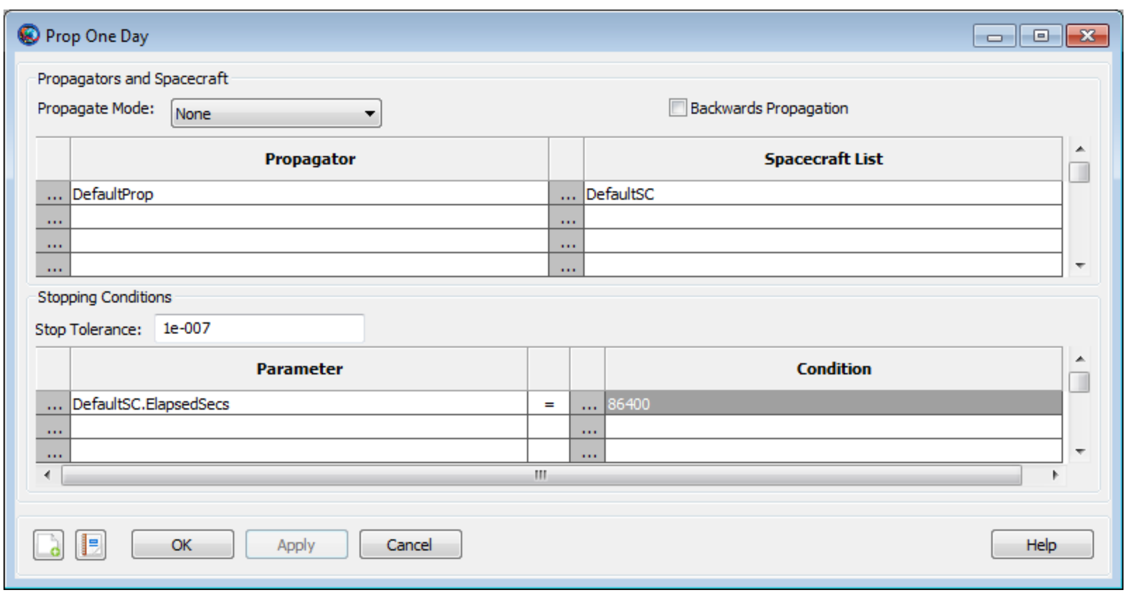

Figure 2. Prop One Day Command Configuration

Configure the Target Sequence

Now that the structure is created, we need to configure the various parts of the **Target** sequence to do what we want.

Configure the Vary TOI Command

1. Double-click **Vary TOI** to edit its properties. Notice that the variable in the **Variable** box is **`TOI.Element1`**, which by default is the velocity component of TOI in the local Velocity-Normal-Binormal (VNB) coordinate system. That's what we need, so we'll keep it.
2. In the **Initial Value** box, type **`1.0`**.
3. In the **Max Step** box, type **`0.5`**.
4. Click **OK** to save these changes.

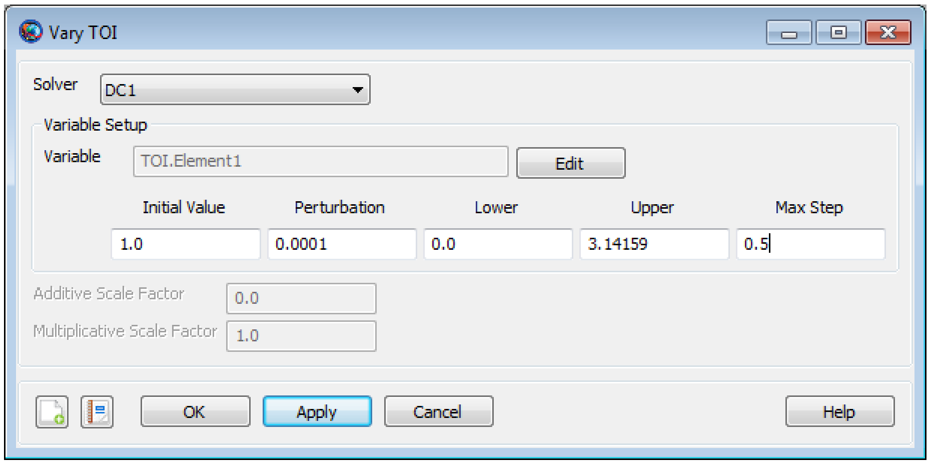

Figure 3. Vary TOI Command Configuration

Configure the Perform TOI Command

1. Double-click **Perform TOI** to edit its properties. Notice that the command is already set to apply the **TOI** burn to the **DefaultSC** spacecraft, so we don't need to change anything here.
2. Click **OK**.

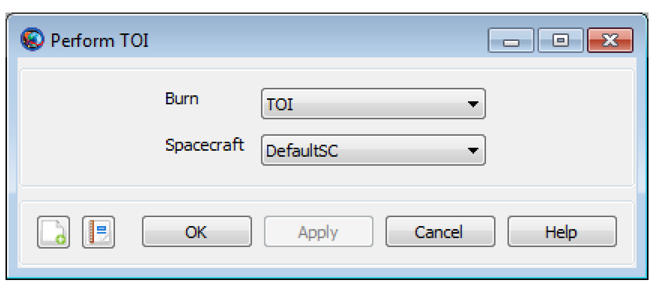

Figure 4. Perform TOI Command Configuration

Configure the Prop to Apoapsis Command

1. Double-click **Prop to Apoapsis** to edit its properties.
2. Under **Parameter**, replace **`DefaultSC.ElapsedSecs`** with **`DefaultSC.Earth.Apoapsis`**.
3. Click **OK** to save these changes.

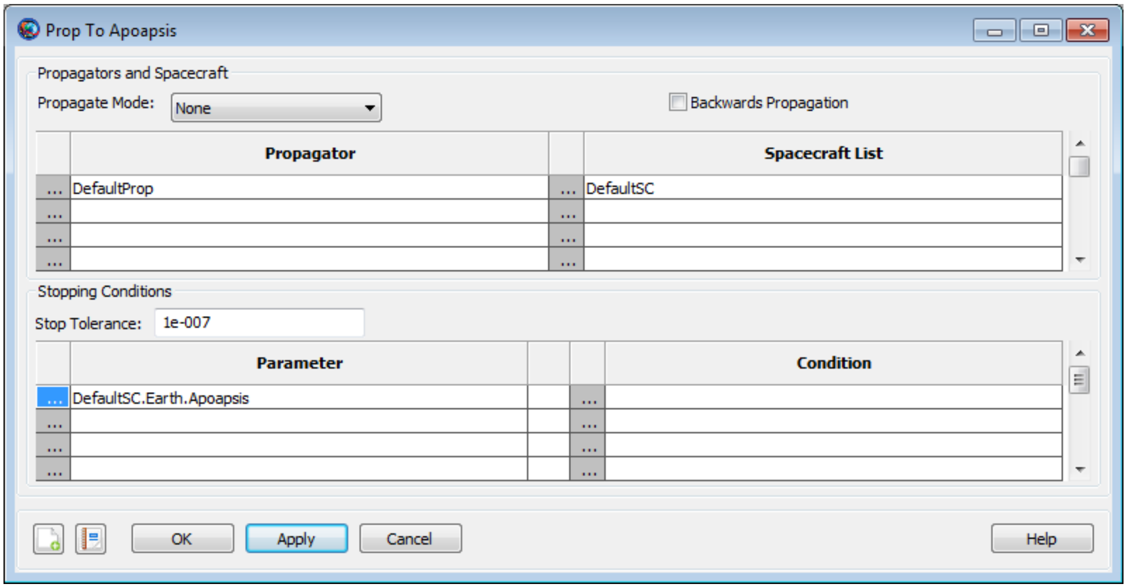

Figure 5. Prop to Apoapsis Command Configuration

Configure the Achieve RMAG = 42165 Command

1. Double-click **Achieve RMAG = 42165** to edit its properties.
2. Notice that **Goal** is set to **DefaultSC.Earth.RMAG**. This is what we need, so we make no changes here.
3. In the **Value** box, type **42164.169**, a more precise number for the radius of a geosynchronous orbit (in kilometers).
4. Click **OK** to save these changes.

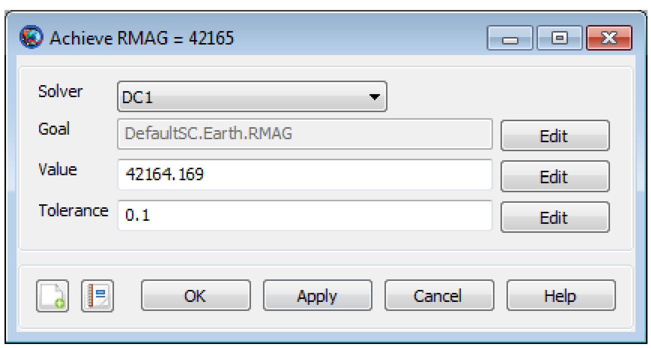

Figure 6. Achieve RMAG = 42165 Command Configuration

Configure the Vary GOI Command

1. Double-click **Vary GOI** to edit its properties.
2. Next to **Variable**, click the **Edit** button.
3. Under **Object List**, click **GOI**.
4. In the **Object Properties** list, double-click **Element1** to move it to the **Selected Value(s)** list. See the image below for results.

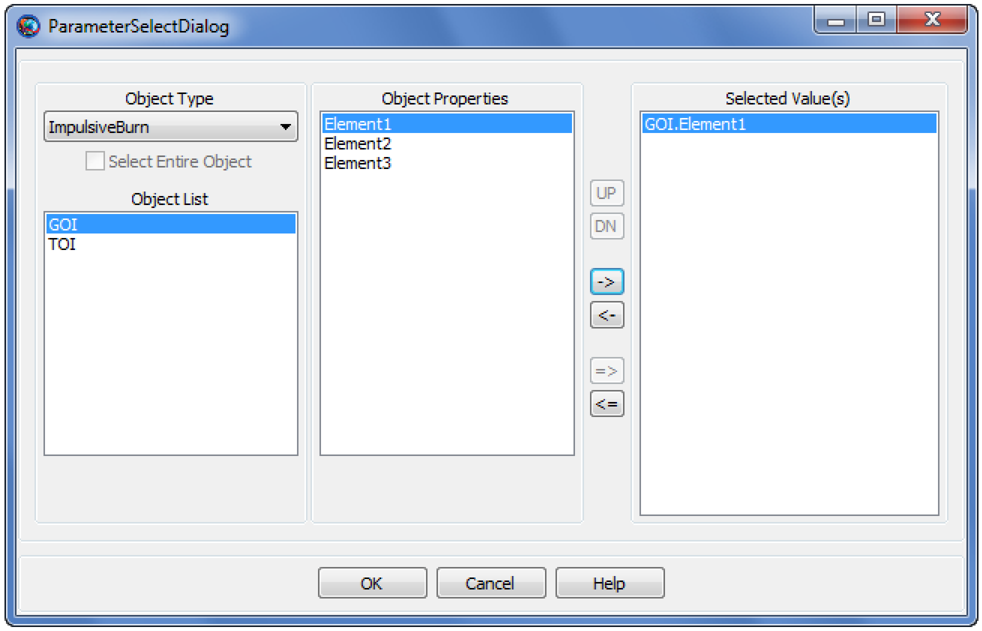

Figure 7. Vary GOI Parameter Selection

5. Click **OK** to close the **ParameterSelectDialog** window.
6. In the **Initial Value** box, type **`1.0`**.
7. In the **MaxStep** text box, type **`0.2`**.
8. Click **OK** to save these changes.

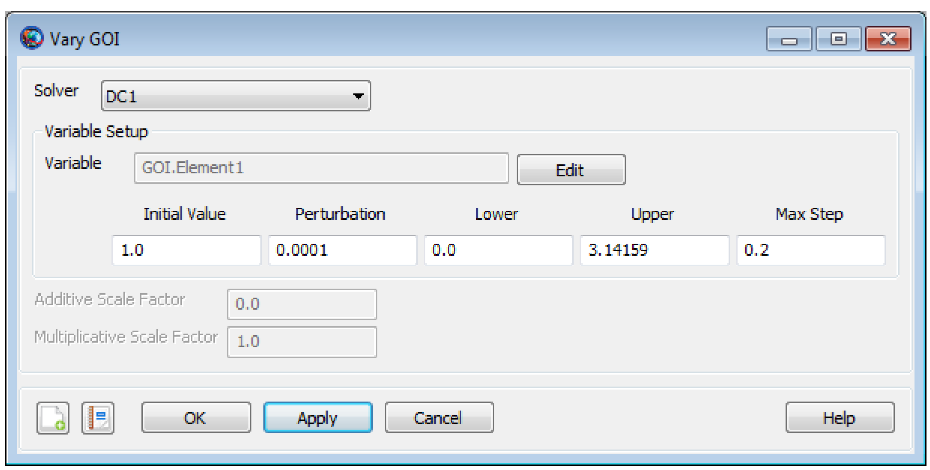

Figure 8. Vary GOI Command Configuration

Configure the Perform GOI Command

1. Double-click **Perform GOI** to edit its properties.
2. In the **Burn** list, click **GOI**.
3. Click **OK** to save these changes.

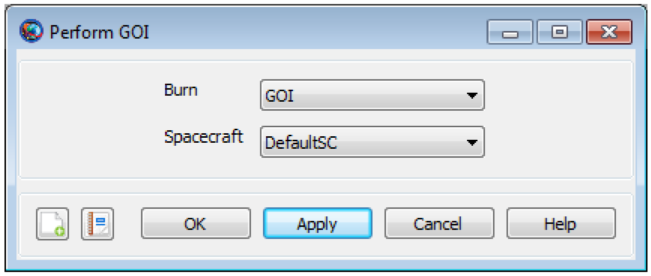

Figure 9. Perform GOI Command Configuration

Configure the Achieve ECC = 0.005 Command

1. Double-click **Achieve ECC = 0.005** to edit its properties.
2. Next to **Goal**, click the **Edit** button.
3. In the **Object Properties** list, double-click **ECC**.
4. Click **OK** to close the **ParameterSelectDialog** window.
5. In the **Value** box, type **`0.005`**.
6. In the **Tolerance** box, type **`0.0001`**.
7. Click **OK** to save these changes.

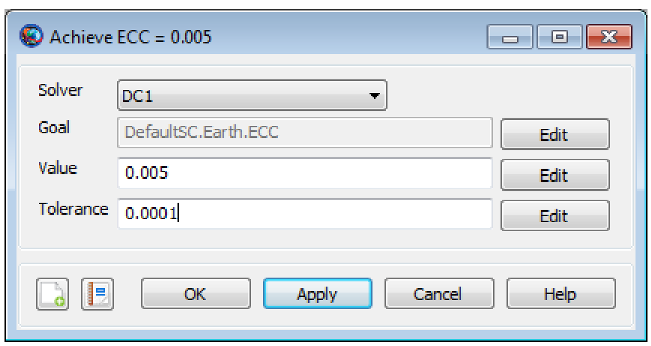

Figure 10. Achieve ECC = 0.005 Command Configuration

Run the Mission

Before running the mission, click **Save** () and save the mission to a file of your choice. Now click **Run** (). As the mission runs, you will see GMAT solve the targeting problem. Each iteration and perturbation is shown in **DefaultOrbitView** window in light blue, and the final solution is shown in red. After the mission completes, the 3D view should appear as in to the image shown below. You may want to run the mission several times to see the targeting in progress.

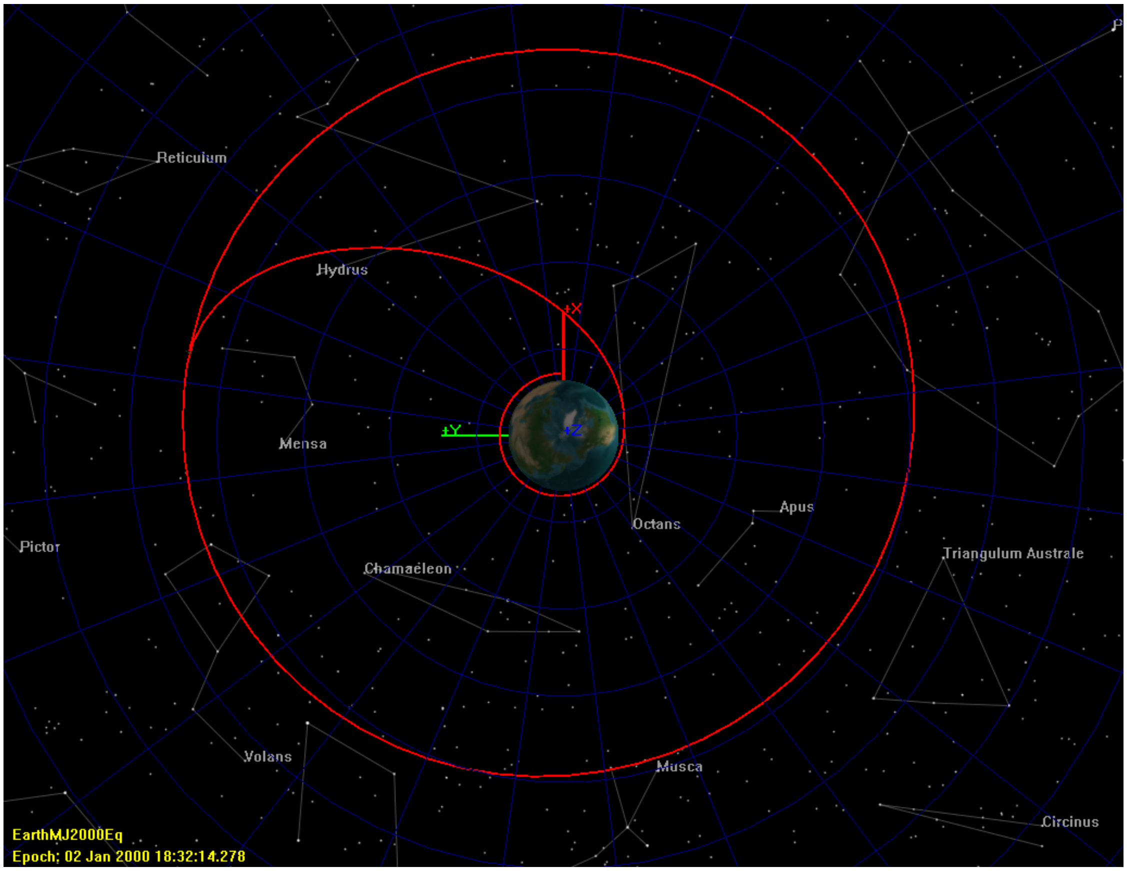

Figure 11. 3D View of Hohmann Transfer

If you were to continue developing this mission, you can store the final solution of the **Target** sequence as the initial conditions of the **TOI** and **GOI** resources themselves, so that if you make small changes, the subsequent runs will take less time. To do this, follow these steps:

1. In the **Mission** tree, double-click **Hohmann Transfer** to edit its properties.
2. Click **Apply Corrections**.
3. Now re-run the mission. If you inspect the results in the message window, you will see that the **Target** sequence converges in one iteration because you stored the solution as the initial condition.