#=============================================================================

#

# Flight Mechanics Analysis Tools Interoperability and Component Sharing (TI-18-01313)

#

# Monte-Copernicus Interface: Use Case 3.1, a Copernicus to Monte solution transfer

#

# By: Ricardo L. Restrepo (Ricardo.l.restrepo@jpl.nasa.gov)

#

#=============================================================================

Use Case 3.1: NRHO/HLS, a Copernicus to Monte solution transfer.

monteCop Script to be used: bsp2cosmic.py

Problem Description: This use case represents the Copernicus to MONTE data transfer scenario of a Low Lunar Orbit (LLO) to a Near Rectilinear Halo Orbit (NRHO) trajectory. During the Artemis III mission, one of the phases will involve returning the astronauts from the south pole of the Moon to the Orion capsule at the NRHO Gateway orbit.

Procedure:

A low fidelity trajectory is first designed in Copernicus (gen\_LLO\_to\_NRHO\_imp\_BSP.ideck). The trajectory design is performed assuming only impulsive burns, the trajectory begins at the south pole on a low elliptical orbit (post-ascent segment), and a circularizing maneuver is applied at the north pole at apoapsis. After several revolutions in this circular orbit, a maneuver is performed to initiate the transfer to the Lunar NRHO. Once the trajectory is design, it is then saved as an SPK kernel (gen\_LLO\_to\_NRHO\_imp\_BSP.bsp). Find trajectory details at AAS paper †.

Transfer to Monte:

the trajectory is loaded as an SPK kernel into Monte and then it is converted into a Monte compatible solution (a Cosmic timeline) by using the script bsp2cosmic.py. The script scan the SPK kernel to identify the locations of the maneuvers. A δv search step of 10 seconds is used, and the velocity discontinuity over time is shown in Figure 14. The constant changes in velocity of approximately 15 m/s during the low lunar orbit segments (the first 7 hours of the orbit) are due to the rapid rotation of the low lunar orbit. Hence, a minimum ΔV detection of 20 m/s is used. The three maneuvers of the total transfer are evident in the velocity discontinuity plot. These discontinuities are used to locate and compute impulsive burns for the transfer. Control points are inserted at each location of the maneuvers, and the orbit is set to start with a fixed state at t0 and a fixed target state at apoapsis of the NRHO. In total, 5 control points with 4 discontinuity constraints and 3 maneuvers are included. The output solution (LLO\_to\_NRHO\_Cosmic.py,) is then used as an initial condition for an optimization procedure, where discontinuities, due dynamical and numerical model differences between Copernicus and Monte are removed, while granting a ∆V optimal solution (LLO\_to\_NRHO\_Cosmic\_OPT.py).

**Files Required:**

* MONTE-Copernicus interface scripts (contained in the python module)
* Initial Copernicus trajectory: gen\_LLO\_to\_NRHO\_imp\_BSP.ideck
* Copernicus generated SPK kernel: gen\_LLO\_to\_NRHO\_imp\_BSP.bsp
* Reference NRHO (15-year Gateway reference orbit produced by JSC): receding\_horiz\_3189\_1burnApo\_DiffCorr\_15yr.bsp

**Test Procedure:**

1. Start with an HLS Copernicus solution transfer trajectory (provided by Marshall Space Flight Center) LLO-to-NRHO (Figure 1).  Create an SPK file containing the transfer trajectory (assigned SPICE id as -30100). The transfer has been modified to implement only impulsive solutions (original solution is modeled with finite burns).
2. Use bsp2cosmic\_HLS.py to generate a pre-converged Cosmic Solution. This is the main script of the interface which converts SPK kernels (\*.bsp files) into MONTE/Cosmic trajectories. Run scripts as

>> bsp2cosmic\_HLS.py gen\_LLO\_to\_NRHO\_imp\_ext7d\_BSP.bsp -ov -o 3 -tl 1 -dt 10 -dv 20 -n LLO\_to\_NRHO\_Cosmic.py

* 1. Test script, explore the user input options. Test the Velocity discontinuity visualization capability, and parameter tuning (e.g. dvSearch time step, and minDV search) (See Figure 2).
  2. After tuning and finding the 3 Impulsive maneuvers for the LLO to NRHO transfer, save Cosmic solution.

1. Converge cosmic solution to obtain a continues optimized trajectory (use SNOPT or DBLSE optimizers). See Figure 3.
   1. Output should be a MONTE-python file.
   2. Generate new files with continues optimized solution: \*.spk file as LLO\_to\_NRHO\_Cosmic\_OPT.bsp (assigned SPICE id as -30100), Cosmic timeline as LLO\_to\_NRHO\_Cosmic\_OPT.py, and \*.boa file as LLO\_to\_NRHO\_Cosmic\_OPT.boa. Note: BOA files are binary files archives used by MONTE to save all information related with the trajectory, as ephemeris, trajectory itself, partials, spacecraft models, etc. The BOA file is an additional product that will allow Navigation post-processing analysis, like Orbit Determination, or Flight Path Control analysis.

**Run and Save** Cosmic Solutions:

There are different ways to run a Cosmic optimization problem and to save the converged solution.

And easy way to control the optimization is by using the cosmic iterative more

(e.g. >> cosmic.py -i yourCosmicFile.py), then, you can run the optimizer just by typing:

>> run(),

The cosmic solutions obtained from the converged Ideck, should converged easily

(specially when using SNOPT on both Monte and Copernicus). Once the solution has converged,

the user can save it in different formats:

BOA file:

Option 1: only trajectory

>> saveSol(‘myNewSolution.boa’)

Option 2: only trajectory or trajectory + Partials (Want this for OD analysis)

>> saveSol(‘myNewSolution.boa’)

SPK: Save solution as SPICE kernel

>> saveSpk(‘myNewSolution.bsp’)

New Cosmic Timeline: Save converged solution as a new Cosmic timeline

>> saveChkPt(‘myNewSolution.py’)

† “Trajectory Reverse Engineering: A General Strategy For Transferring Trajectories Between

Flight Mechanics Tools” AAS 23-312. 33rd AAS/AIAA Space Flight Mechanics Meeting, Austin, Texas, January 15-19 2023. Ricardo L. Restrepo

\*\*Note: A detailed description of a test to this Use Case, using a Windows machine while accessing the MONTE toolkit through a Docker Container, can be found at: *‘/monteCop/doc/NESC\_tool\_test\_summary.docx’*