GMAT TLE Writer Documentation

The GMAT TLE Writer is distributed as a collection of GMAT scripts that receives an ephemeris file and solves for a valid TLE using the GMAT Yukon Optimizer. The TLE solution can be modified with a variety of convergence criteria, including a cost function viability constraint, convergence criteria between iterations, and the number and location of states to evaluate from the given ephemeris file.

The Use Case:

The GMAT TLE writer is designed to create a SPG4 valid TLE from a user provided ephemeris file of any GMAT accepted format (currently STK, SPK, Code500, and CCSDS-OEM). Users are responsible for ensuring the accuracy and appropriate format of these files. For users wanting to create a TLE from parameters in a defined GMAT mission scenario, GMAT offers the ability to create ephemeris files compatible with this package. The TLE and the elements will be created according to the format displayed in figure 1. The TLE writer is provided to give GMAT users the additional option to create TLEs for a craft and testing has been performed to ensure the functionality and ability of the writer to create TLEs that match the given ephemeris.

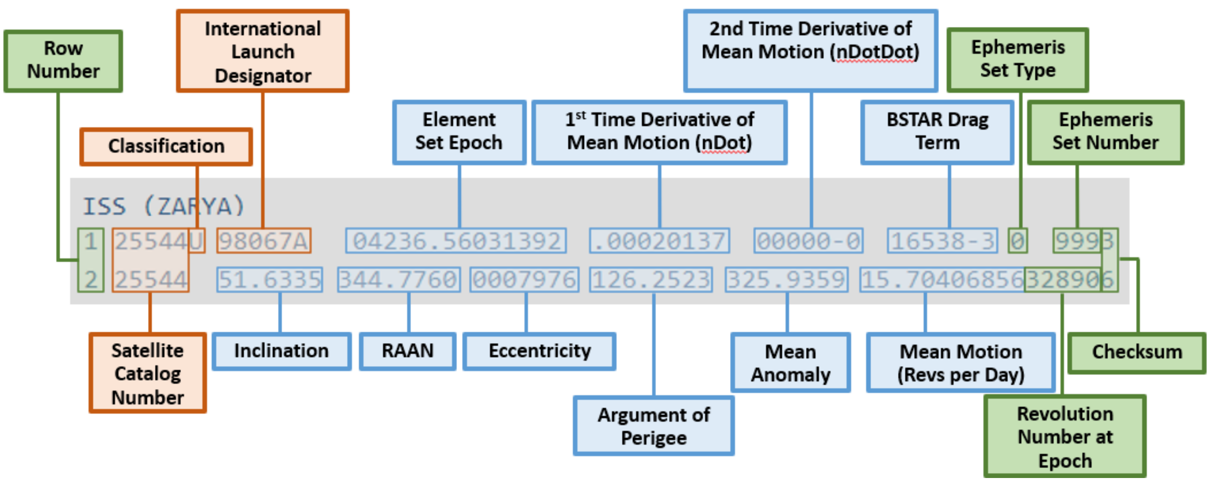

Figure 1: A TLE with each of its elements labelled

Scripts:

Within the R2022_TLEWriterSample, the user will find two top level scripts for writing a TLE. There is a generic script for Code500, CCSDS-OEM, and STK ephemeris files. A separate script specifically for SPK type files due to differences in accessing the date using GMAT. Both rely on additional scripts found within the R2022_TLEWriter/Utilities directory. Here the user will find the bulk of the scripting for the TLE writer, which the top levels scripts access through “#include” commands. Users can access these to understand the specifics of how the process is performed and make specific changes to the optimization procedure as needed. Users are ***strongly warned*** not to make edits without a thorough understanding of the algorithm and the contents of the script.

The User Inputs:

For both scripts, there are various parameters required for writing the TLE. In Figure 2, the inputs for the generic TLE solver script can be seen. The first set of parameters are the name of the object (as appearing in the line 0 of the TLE), the satellite catalog number, classification, and international launch designator. The next set covers the ballistic parameters of the object which are used in the calculation of BSTAR. If these values are not known by the user, the user may decline to include the BSTAR parameter in the TLE through the optional settings. The third set of inputs define the type ephemeris file being evaluated and the its location. Note that the type must be defined twice due to the way in which GMAT handles variable declaration and manipulation before running the mission sequence. The final set of inputs allow the user to manipulate the settings for the Yukon optimizer that is solving for the TLE. The tolerance and iteration parameters are discussed in more detail in the Yukon optimizer section of the GMAT User's Guide. The CostFunctionConstraint allows the user to define the RMS value under which convergence will be permitted, with a default value of 1 km. Finally, the user may determine how many points to evaluate with each optimization pass, with a default of 15.

Figure 2: The parameters needed to create a TLE

Figure 3 shows the optional parameters that can influence the creation of the TLE. When analyzing an ephemeris, by default the script will evaluate it over the full span from the first to final point in the data. If the user provides modified span dates, for either the start or end point, then any provided times will be used instead. Custom times must fall within the span of the given ephemeris. Additionally, the TLE will be created by default with elements and epoch matching the information for the first available time in the ephemeris file. By specifying an anchor epoch that falls between the start and end of the ephemeris, a TLE can be created to the specified time. Additional options exist if the user wishes to have a custom filename output for the TLE, custom directory for the output TLE. Lastly, the user can influence how the drag terms are handled by the writer. Since the writer does not incorporate the effects of Ndot and Ndotdot, these will be set to zero unless the user provides a value, which will be passed through to the final TLE. Additionally, if the user does not have sufficient ballistic information for the craft, they may choose to have BSTAR set to zero.

```
---------- Optional Settings
----------------------------------------

seCustomFitSpan          = 0;                              % Flag to use modified dates. Set to 1 to use a custom StartDate, 2 for a custom EndDate, and 3 for both
itSpanStart              = '22 Jul 2014 12:28:10.811';     % UTCGregorianDate    (E.G. '04 May 2002 11:45:16.000')
itSpanEnd                = '22 Jul 2014 22:29:10.811';     % UTCGregorianDate    (E.G. '04 May 2002 11:45:16.000')

seAnchorEpoch            = 0;                              % Flag to use modified dates. Set to 1 to Specify a specific Epoch at which to generate the TLE
nchorEpoch               = '22 Jul 2014 17:28:10.811';     % UTCGregorianDate    (E.G. '04 May 2002 11:45:16.000')

UseCustomName            = 1;                              % Uncomment to use a custom file name for the TLE output
CustomTLEName            = 'SomeRandomTLESPK.txt';         % Alternate filename for TLE output. Recommended to have '.txt' suffix.
UseCustomDirectory       = 1;                              % Uncomment to use a custom diretory path for the TLE output. May be absolute or relative to the script directory
OutputDirectory          = '/Some/Full/Path/';             % Alternate path for TLE output. Path must be absolute.

Ndot                     = 0.0;                            % Uncomment to pass through a Ndot parameter. GMAT/SGP4 does not solve for or use this parameter.
Ndotdot                  = 0.0;                            % Uncomment to pass through a Ndotdot parameter. GMAT/SGP4 does not solve for or use this parameter.
```

Figure 3: The optional parameters to create a TLE

Notes on the Methodology:

Since the elements of a TLE are tied to the assumptions of the SGP4 propagator, it is necessary to solve for a valid set of elements. The process to perform this operation was originally developed in 2017 for use by SDO, but has been expanded in 2021 to for general use. GMAT first takes the script that the user provides as the input for a GMAT propagator of the given ephemeris type. This ephemeris propagator will extract the data points the optimizer will use as the truth the in the TLE optimization. By default, the full span of the ephemeris file will be used, but if the user provides modified span dates these will be used instead. Once the ephemeris file truth points are found, GMAT will create an initial guess TLE at initial epoch of the span or at an optionally defined Anchor epoch using the BrouwerLong Keplarian elements. This temporary TLE will be propagated over the same steps as the ephemeris points, and the difference between these evaluated according to the cost function. Following this, the Yukon optimizer at each iteration will vary the TLE's orbital elements and compare the propagated state against the truth. Once the optimizer has reached the convergence conditions as specified by the user, GMAT will write the TLE of the convergent solution into a text file. In the case that the solution fails to converge, the user may need to assess the settings for the optimizer, raising the convergence threshold or the CostFunctionConstraint as needed.

$$\text{Cost Function} = \sqrt{\left.\sum\left(x_{diff}^2 + y_{diff}^2 + z_{diff}^2\right)\middle/_{\#Steps}\right.}$$

Links

https://celestrak.com/columns/v04n03/

https://www.celestrak.com/columns/v04n03/

https://www.space-track.org/documentation#tle