TRACKING OF SPACE DEBRIS FROM PUBLICLY AVAILABLE DATA

THINK OF A BETTER TITLE

Philip Hoddinott

Submitted in Partial Fulfillment of the Requirements

for the Degree of

MASTER OF SCIENCE

Approved by: People

Department of Mechanical, Aerospace, and Nuclear Engineering

Rensselaer Polytechnic Institute

Troy, New York

August 2018

August 7, 2018

Contents

List of Tables

List of Figures

0.1 Acknowledgments

Thank lots of people here

Mom, dad, David, P Anderson John B for git and PHP Paul McKee for latex

0.2 Abstract

The purpose of this project is to acquire information about earth orbiting debris, turn the information into a usable format, and develop a targeting algorithm for a debris removal satellite. The information is things such as orbital elements, debris size, and sping if i can find htis.

Should this be accomplished in a timely manner more work will be done on the orbital dynamics of OSCAR getting to said debris.

0.3 Introduction

Talk more about project. one or two paragraph here

0.3.1 Space Debris

Todo

0.3.2 CubeSats

Todo

0.3.3 OSCAR

Unsure if talking about oscar? Yes not

0.3.4 Orbital Elements

Talk about orbital elements, define them.

0.4 Data Sources and Formats

First data sources for space debris must be found, and the data formats must be understood. The most common data format for orbital elements is the Two Line Element Set (TLE). Despite the large number of objects in space, real time data is surprisingly hard to find. While there are various websites and programs to track the location of objects such as the International Space Station, debris from old spacecraft is not quite as popular. The website space-track.org publishes TLEs for public use. First the TLE format will be explained then their acquisition will be discussed.

0.4.1 The Two Line Element Format

Orbital elements provide the means to determine a theoretical orbit. Since spacecraft are constantly experiencing forces such as atmospheric drag or solar wind their orbital elements are constantly changing.

A Two Line Element (TLE) is a data format that encodes a list of orbital elements for an Earth-orbiting object for a given point in time [Re do this]

An example shows how orbital infomration is derived from a Two Line Element [1].

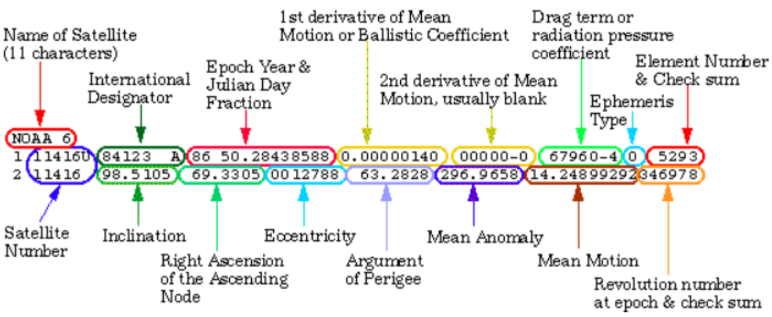

Figure 1: TLE with descriptions

1. **Name of Satellite:** NOAA 6

This is the name of the satellite. In this example it is NOAA 6.

2. **International Designator:** 84123 A

   This number shows the last two digets of the launch year and what number launch of that year it was. In this example it is 84 123 A. The satellite was launched in the year 1984 and it was the 124th launch of the year. The A means that it was the first object that came from the launch. Check this is right

3. **Epoch Date and Julian Date Fraction:** 86 50.28438588

   The Epoch date shows the year this TLE was made and the Julian Date fraction is the number of days in said year.

4. **First Derivative of Mean Motion:** 0.00000140

   This is the daily rate of change of number of revolutions the object completes per day divided by two. This is also called the ballistic coefficient.[1] It's units are $\frac{rev}{day^2}$.

5. **Second Derivative of Mean Motion:** 00000-0

   This is the second derivative of mean motion divided by six, with units of $\frac{rev}{day^3}$.

6. **Drag term or B* Term:** 6970-4

   Of the terms given here, the B* term is the least heard of. It is a way of modeling drag on orbiting objects in propagation models. Aerodynamic drag is given by the following equation:

$$F_D = \frac{1}{2}\rho C_d A v^2 \tag{1}$$

   Where A is the area, C_d is the drag coefficient, v the velocity, and ρ is the fluid density. From Newton's second law

$$F = m \times a \tag{2}$$

the accleration due to the force of drag is

$$a_D = \frac{F_D}{m} = \frac{\rho C_d A v^2}{2m} = \frac{C_d A}{m} \times \frac{\rho v^2}{2} \tag{3}$$

The ballistic coefficient is given by the following equation:

$$B = \frac{C_d A}{m} \tag{4}$$

The starred ballistic coefficient is then

$$B^* = \frac{\rho_0 B}{2} = \frac{\rho_0 C_d A}{2m} \tag{5}$$

This turns the equation for acceleration due to drag into [2][3]

$$a_D = \frac{\rho}{\rho_0} B^* v^2 \tag{6}$$

Possibly not needed maybe take out.

7. **Element Set Number and Check Sum**

8. **Satellite Number:** 11416U

   This is the Satellite Catalog Number and designation of the object. A U means unclassified.

9. **Inclination (degrees):** 98.5105

   This is one of the orbital elements.

10. **Right Ascension of the Ascending Node (degrees):** 69.3305

    This is one of the orbital elements.

11. **Eccentricity:** 0012788

    This is one of the orbital elements.

12. **Argument of Perigee (degrees):** 63.2828

    This is one of the orbital elements.

13. **Mean Anomaly (degrees):** 296.9658

    This is one of the orbital elements.

14. **Mean Motion:** 14.24899292

    This is the orbits per day the object completes.

15. **Revolution Number and Check Sum:** 346978

A second example is given below. This time with references to the position of values, for extraction. The line under the dashes is the reference number line.

```
ISS (ZARYA)
1 25544U 98067A   04236.56031392  .00020137  00000-0  16538-3 0  9993
2 25544  51.6335 344.7760 0007976 126.2523 325.9359 15.70406856328906
---------------------------------------------------------------------
1234567890123456789012345678901234567890123456789012345678901234567890
1         2         3         4         5         6         7
```

Table 1 describes the second example TLE[4].

to do here, add more details on what the checksum and stuff liek that is

The process of TLE gathering and updating is somewhat shadowy. [5] What is known is that observations are collected multiple times per day at the Joint Space Operations Center (JSPOC) which is operated by the US Air Force Space Command (AFSPC). Then the unclassified TLEs are passed on to space-track for public release.

0.4.2 Space-Track.org

Information about objects in space may be found the website space-track.org. Space-track is the main source for orbital data, though some are also avalbile from the website celestrak.com. Of the two space-track has far more information and better methods of access. "Space-Track.org is managed, maintained and administered by JFSCC"[6]. Put more information about the website here.

space-track allows information to be downloaded manually from the TLE search [4] or the satellite catalog search [7]. While these are useful tools the code needs a way to automatically download data. Fortunately space-track also has an API that allows queries.

Table 1: Description of TLE

Line 0		
Columns	**Example**	**Description**
1-24	ISS (ZARYA)	The common name for the object based on information from the Satellite Catalog.
Line 1		
Columns	**Example**	**Description**
1	1	Line Number
3-7	25544	Satellite Catalog Number
8	U	Elset Classification
10-11	98	International Designator (Last two digits of launch year)
12-14	067	International Designator (Launch number of the year)
15-17	A	International Designator (Piece of the launch)
19-32	04	Epoch Year (last two digits of year)
21-32	236.56031392	Epoch (day of the year and fractional portion of the day)
34-43	.00020137	1st Derivative of the Mean Motion with respect to Time
45-52	00000-0	2nd Derivative of the Mean Motion with respect to Time (decimal point assumed)
54-61	16538-3	B* Drag Term
63	0	Element Set Type
65-68	999	Element Number
69	3	Checksum
Line 2		
Columns	**Example**	**Description**
1	2	Line Number
3-7	25544	Satellite Catalog Number
9-16	51.6335	Orbit Inclination (degrees)
18-25	344.7760	Right Ascension of Ascending Node (degrees)
27-33	0007976	Eccentricity (decimal point assumed)
35-42	126.2523	Argument of Perigee (degrees)
44-51	325.9359	Mean Anomaly (degrees)
53-63	15.70406856	Mean Motion (revolutions/day)
64-68	32890	Revolution Number at Epoch
69	6	Checksum

0.4.3 Space-Track Query

0.4.4 SatCat?

0.5 NORAD Space-Track

Describe the site desribe the SATCAT, then the TLE query

0.6 Code Overview

The code obtains orbital data for space debris depending on what variables the user has inputted, such as the launch year of the debris or how recent the data should be. First the code ascertains what data is already available. It checks to see if there is a .mat file for the SATCAT. This is simply a file that contains the catalog ids of the debris. If this file does not exist then the code has not been run for the current parameters, and the code will be run.

The code will also check to see if there is a TLE .mat file. If this file exists the code then checks when the file was created. Depending on user variables, the code may consider the information to be out of date, will rerun everything. However if the TLE .mat file is recent, then the code will load the current debris data into the workspace.

Assuming that the code detects that it has no data, or the data is out of date the code will work in the following steps

1. The desired NORAD satellite catalog ids will be collected with the get_SATCAT.m file. These ids are determined by user variables such as launch date. The code downloads them as a csv, strips things like quotation marks away, and sorts the ids in order.

   - if a SATCAT .mat file exists and is recent this step is skipped.

2. The code then downloads the TLEs as .txt files from the satellite ids. It does this in groups that have their size determined by the user.

3. The .txt files are then parsed, the TLEs are turned into usable orbital information, and saved as an array in MATLAB. This is done in the readTLE_txt.m file.

4. Finally the array of debris information is sorted in order of satellite id and duplicate rows are removed. It should be noted that not all TLEs requested will be provided. So if the code requests TLEs of objects with a id from 1000 to 1100 and expects 100 TLEs,

but some are classified, then duplicated TLEs will be returned to make 100 TLEs. As such these duplicates must be removed. The final array is stored in a .mat file with it's date of creation noted. The user now has a usable list of space debris and their orbital elements.

What now follows is a more in depth explanation of each file.

0.6.1 VarStore.m

This is a file where a few imporant variables are stored. Depening on how the code is reformatted, it might get deleted.COME BACK TO THIS

0.6.2 UserPass.m

at This file is where the username and password for space-track.org are stored.

0.6.3 MASTER_TLE.m

This file is the master file for the Two Line Element MATLAB files. Running it will run all of the associated MATLAB files. These MATLAB files take some time to run, so it may be convenient to toggle the files run by the MASTER_TLE.m

0.6.4 get_SATCAT.m

Get_SATCAT.m is the MATLAB file that gets the satellite catalog numbers of all orbital debris launched after a given year and with the "RCS_SIZE" value equal to "SMALL". The Launch Year can be set by the values given by the user in VarStore.m The default launch year is set to 1990. Note that an earlier launch year will provide more data, and thus it will take more time to process. This may cause a time out error. Should this happen the timeOutVal in VarStore.m should be adjusted to be longer.

Once the SATCAT csv file has been acquired the file then formats it. The quotation marks that are around every entry are removed. The debris that have already deorbited are also removed. Finally the debris is sorted by NORAD Catoluge Id, and saved as a .mat file.

0.6.5 get_Multiple_TLE_from_Id.m

This file takes the Ids given in the .mat file and then obtains the Two Line Element files associated with the debris. It does this by saving the TLEs in .txt files. The number of TLEs grabbed at a time and saved in a .txt file is set by the user. It is not advised to go above 400, as that causes time out errors.

Withing MASTER_TLE.m, this code is surrounded by a a loop and a try/catch error. The loop runs through all the different blocks of TLEs. In the case of a time out the the try/catch will attempt to download the TLEs a second time.

At this point in the code the space debris has had it's Two Line Element files saved to .txt files. This is the longest part of the code to run, but also with the .txt files saved, these functions need only be run once every few days.

0.6.6 readTLE_txt.m

This MATLAB file reads the .txt files and then converts the TLE format to the standard orbital elements stored in a .mat file.

0.6.7 check_TLE_Edit_TLE.m

0.7 Targeting

Once the data has been collected, the next step is to target possible orbits. Since OSCAR has a delta V of Find delta V inclanation and plane changes are out of the question. What is desired is a number of debris in a plane with simialr inclanations

0.8 Conclusion

Bibliography

[1] Amiko Kauderer Kim Dismukes. Definition of two-line element set coordinate system, 2011.

[2] By Dr. T.S. Kelso. Frequently asked questions: Two-line element set format, 1998.

[3] Canadian Satellite Tracking and Orbit Research (CASTOR). B-star drag term, 2010.

[4] Space Track. Basic description of the two line element (tle) format, 2013.

[5] David A Vallado and Paul J Cefola. Two-line element sets–practice and use. In *63rd International Astronautical Congress, Naples, Italy*, 2012.

[6] Space Track. Space track legend, 2018.

[7] Space Track. Space-track satellite catalog search, 2018.

Appendix 1 - MATLAB code