

Space Object Taxonomy
Space Situational Awareness

Dhruv Jain

May 2019

This report summarizes the work done in AAE 497: Space Object Taxonomy

Supervisor Dr. Carolin Frueh

School of Aeronautics & Astronautics
Purdue University
United States

Contents

1	Introduction	1
2	Key Terms	1
3	Truth Set	2
3.1	Setup of Truth Set	3
4	Method Devised	4
4.1	Flowchart to explain classification logic (except Group 3)	7
5	Results	7
6	Conclusions	8
7	Appendix	8
7.1	Functions	8
7.1.1	Truth Set	8
7.1.2	For Method Devised	9
7.2	Workflow	10
7.2.1	Setup For Truth set:	10
7.2.2	Method Devised	10

1 Introduction

In 1957, the USSR launched the first artificial satellite, *Sputnik*. This event kickstarted the space race but no one had imagined that mere 60 years after the historic event that 'Space Debris' will be a serious cause of concern and we will have to be cautious about how satellites are populated and controlled near Earth. Due to poor satellite management in the past and testing of Anti-Satellite Missiles, there has been a tremendous increase in the number of small shrapnels and big scrap metals, which cannot be controlled. These objects revolve around Earth at speeds more than 5-6 times the speed of a bullet. They are a potential threat to all the multi-billion dollar space assets on which we heavily rely, ranging from daily communication to studying weather patterns. It is inevitable that the space debris will exponentially increase if no serious measures are taken against it.

In 1978, NASA scientist Donald J. Kessler proposed a scenario known as Kessler Syndrome. He proposed that due to the high density of objects in an orbit, there will be collisions that will cause a cascade where each collision will generate space debris that further increases the likelihood of collisions. Thus, to safely escape the near Earth orbits in the future it is important to catalog all the objects and actively think of ways to decrease space debris.

This report explains an effort to make a taxonomy of space objects in the Geosynchronous Orbit (GEO) to autonomously classify *new* space objects into one of the 8 groups established by ESA in their CLASSIFICATION OF GEOSYNCHRONOUS OBJECTS 2017 Report. This will help in characterizing the objects based on the group definitions and give a starting point to differentiate between space assets and space debris as well as different types of space debris.

2 Key Terms

1. **Two Line Element (TLE)** is a data format that is used to define orbital characteristics of Earth-orbiting artificial satellites. It consists of two lines of 70 columns that contain the satellite number, international designator, epoch, mean motion, b* drag term, and element set number. This is maintained and created by the United States Air Force. It is important to frequently know the orbital elements from TLE as the satellites are in Non-Keplerian orbits due to perturbations caused by solar radiation pressure, atmospheric drag, and the non-spherical shape of the central body. Thus, the satellites deviate from their Keplerian orbits and there is a need to accurately update the orbit's characteristics.

All the work done in this study revolves around reading the satellite data from the TLEs to compute and characterize their longitude and orbital elements.

2. **SGP4** is a simplified perturbation model. It accounts for perturbations caused by the Earth's irregular shape, drag, radiation, and gravitational effects from the Sun and Moon. The model is a powerful tool to propagate the orbital state vectors of satellites and space debris in the Earth-Centered Inertial frame. The propagated states serve as a good approximation of the future positions of space objects in their Non-Keplerian Orbit.

3. **Orbital Elements** are a group of 6 parameters derived from the state vectors that can define the orbit of a satellite. These parameters are:

- 'a' semi-major axis of an elliptical orbit.
- 'e' eccentricity defines how elliptical or circular the orbit is.
- 'i' inclination angle is the angle between the Earth's equatorial plane and the orbital plane.
- ' Ω ' right ascension of ascending node (RAAN) is the angle between the vernal equinox and the line of the ascending node
- ' ω ' argument of periapsis is the angle from the line of ascending node to the eccentricity vector.
- ' ν ' true anomaly is the angle from the eccentricity vector to the position vector of the satellite in the orbital plane.

It is important to keep track of the first 5 of the 6 as the satellites are in Non-Keplerian Orbit.

4. **Vernal Equinox** is the vector that forms at the intersection of the Ecliptic and Equatorial plane made with the Sun on the vernal equinox originating from the center of Earth and pointing towards the Sun.
5. **Ascending node** is the point at which the satellite in the orbital plane crosses the Earth's equator and transitions from the southern hemisphere to the northern hemisphere.

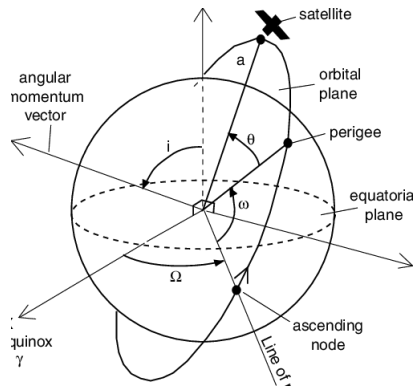


Figure 1: Orbital Elements

3 Truth Set

The space object classification defined in the ESA CLASSIFICATION OF GEOSYNCHRONOUS OBJECTS Report, issue 19 published on 21 April 2017, was recreated to get a truth set. This was achieved by obtaining the TLE for the objects listed in the report and segregating them into groups as they are presented in the ESA report.

Thereafter, the patterns between the objects of each group were studied to come up with a method to automate the addition of *new* objects to the taxonomy. After the object is added to a group then it can be studied based on the properties of the group instead of the need to repropagate to understand its characteristics.

The 8 groups defined in the ESA report are as follows:

- **Group 1** - Under Longitude and Inclination Control(E-W abd N-S Control). This group consists of artificial satellites that are active and can change their position by changing their inclination to move up or down relative to the equator, as well as changing the longitude to correct for longitudinal drift.
- **Group 2** - Under Longitude Control (E-W Control). This group consists of active artificial satellites that have the capability to correct for longitudinal drift or change their longitude but they cannot change their inclination as the artificial satellites in Group 1 can.
- **Group 3** - Controlled Drift Orbit. This group consists of artificial satellites that are in drift orbit but they are active so they can make a maneuver to change their longitudinal drift rate.
- **Group 4** - Drift Orbit. This group consists of artificial satellites that are in drift orbit and all are inactive, thus all the artificial satellites in this group are space debris.
- **Group 5** - Libration Orbit around the Eastern Libration Point. This group consists of artificial satellites that orbit around 75° East. This is because there exists a high mass density that locks the artificial satellites in an orbit around them. Most artificial satellites in this group are space debris.
- **Group 6** - Libration Orbit around the Western Libration Point. This group consists of artificial satellites that orbit around 105° West. This is because there exists a high mass density that locks the artificial satellites in an orbit around them. Most artificial satellites in this group are space debris.
- **Group 7** - Libration Orbit between the two Libration Point. This group consists of artificial satellites that orbit in between the two libration points at 75° East and 105° West. Most artificial satellites in this group are space debris.
- **Group 8** - Highly Inclined Orbit. This group consists of artificial satellites that are in highly inclined orbits.

Objects in Groups 1, 2, 3, and 8 are active artificial satellites that can maneuver as described above. The artificial satellites drift due to multiple perturbations so they require maneuvering to stay at the required longitude. It is important for artificial satellites to maintain their longitude because they serve a particular purpose for a specific region on Earth so they need to stay close to the region and not leave the line of sight. Objects in Groups 4, 5, 6, and 7 are space debris and uncontrollably drifting.

3.1 Setup of Truth Set

The truth set was constructed by first creating 8 text files where each file consisted of the COSPAR number of the objects listed in the ESA report. The aim was to obtain the TLEs of the listed objects but since TLEs do not have COSPAR numbers, the numbers had to be changed to their international designator. Then these numbers were used to obtain two different TLEs, which were 55 days apart, for all the objects through the CelesTrak database.

The SGP4 model in Python was used to propagate all the TLE data to a common epoch. The propagated data was saved by the group to serve as a truth set for each group. Each group's objects were plotted to see any visual similarities in their orbits that can act as an easy method of differentiation between the group. The plots did not serve as a good point of differentiation but they provided a way to convert the raw data into a format that is more intuitive. Below are the example plots of objects of the 8 groups that were made in MATLAB:

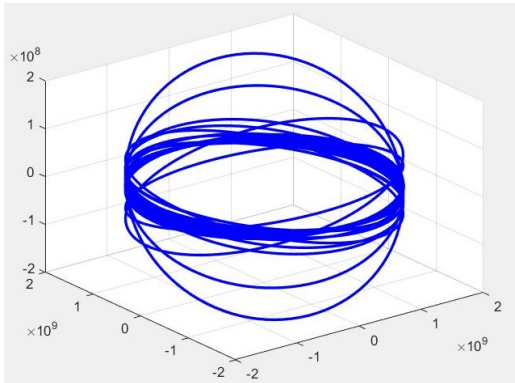


Figure 2: Group 1 objects

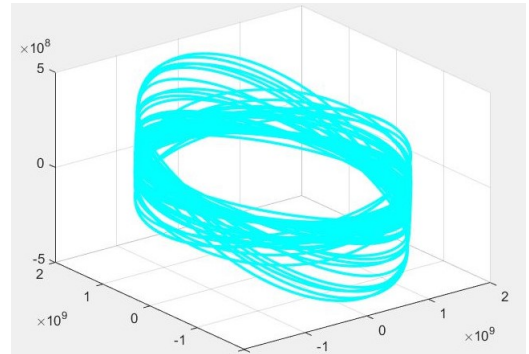


Figure 3: Group 2 objects

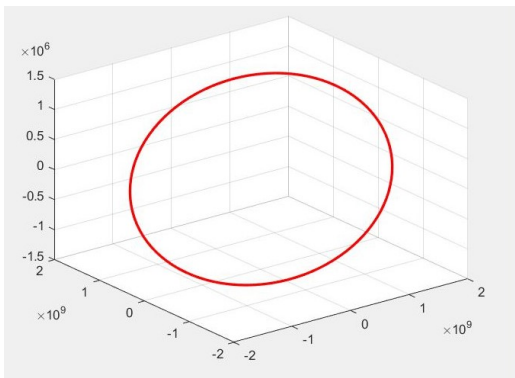


Figure 4: Group 3 objects

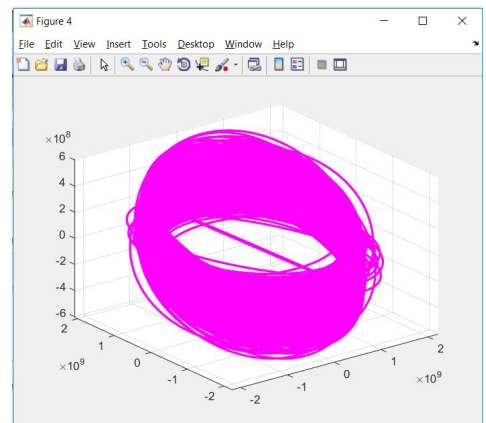


Figure 5: Group 4 objects

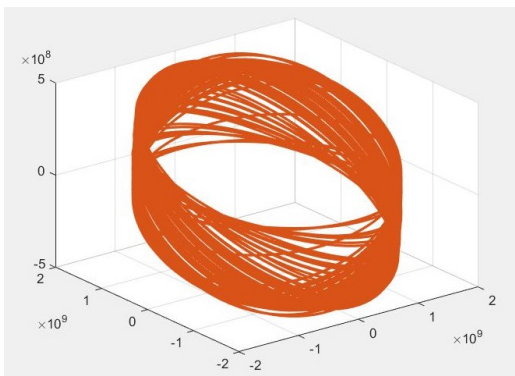


Figure 6: Group 5 objects

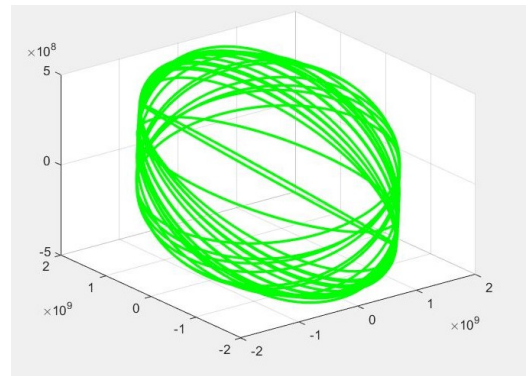


Figure 7: Group 6 objects

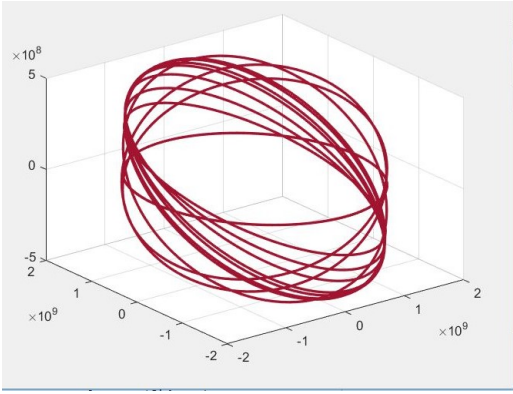


Figure 8: Group 7 objects

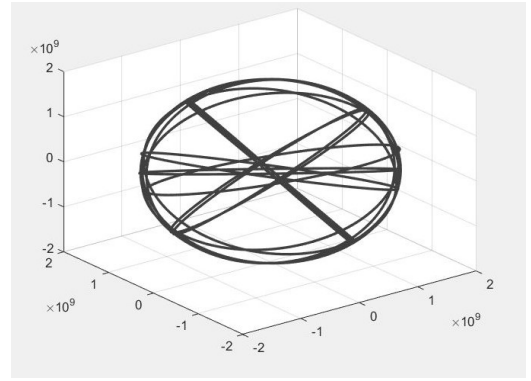


Figure 9: Group 8 objects

4 Method Devised

The first approach was centered around studying the longitudinal drift of the satellites. To do so a function in Python was created to read a TLE file and compute the longitude for a specified epoch.

Consider an epoch denoted by '0-day'. In order to analyze the drift, the longitude of objects using their TLEs at various epochs from the 0-day: 6 hours, 12 hours, 1 sidereal day, 5.75 days, 10.5 days, 15 days, 1 sidereal month, and 55 days were computed. Thereafter, the 0-day TLEs states were propagated to the various epochs and the longitude at the various epochs was computed. This was done to realize the difference in longitude between the propagated and actual states. In addition, how the longitude varies with epoch.

The below plots showcase the longitude of objects at different epochs sorted by their groups.

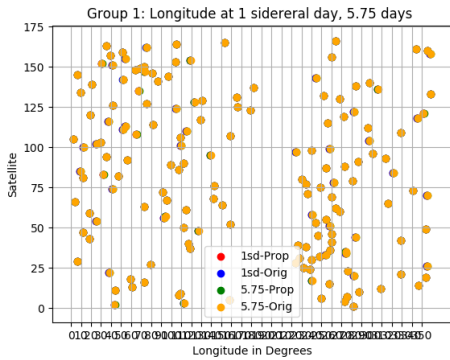


Figure 10: Gr 1

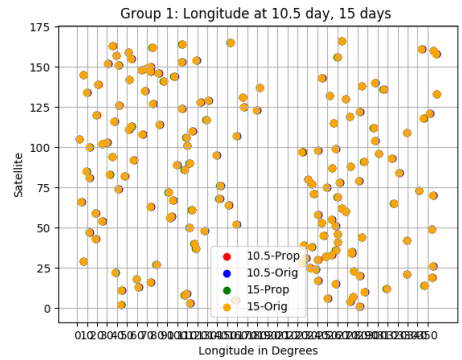


Figure 11: Gr 1

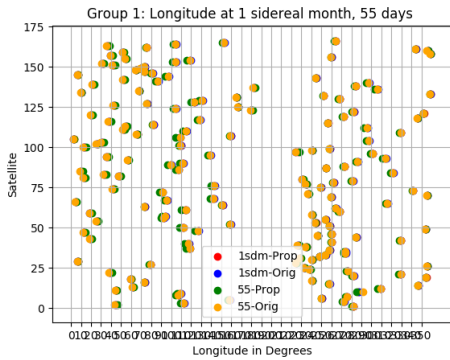


Figure 12: Gr 1

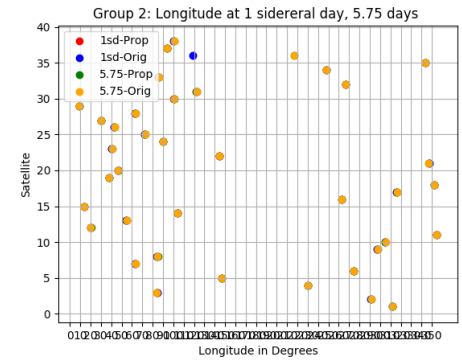


Figure 13: Gr 2

From the plots, it is evident that there is a significant difference between the propagated and actual longitude after 55 days. This highlights that a correction maneuver for some of the objects was performed. Thus, the TLE data at 0 and 55 days can be used to differentiate between active satellites and space debris, and assist in

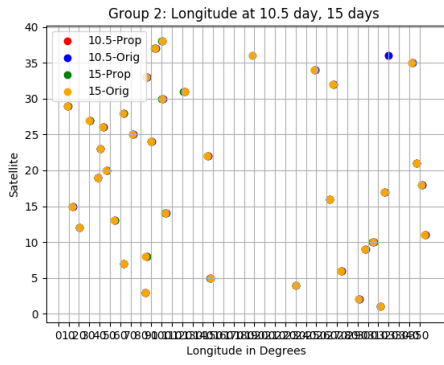


Figure 14: Gr 2

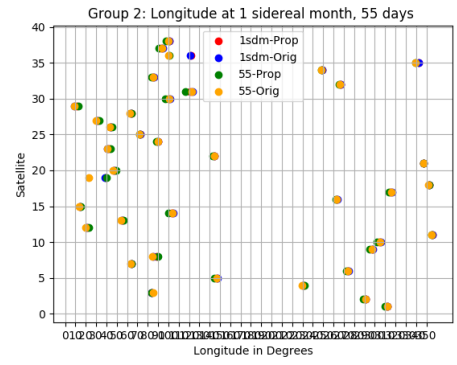


Figure 15: Gr 2

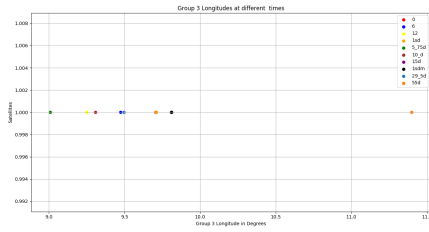


Figure 16: Gr 3

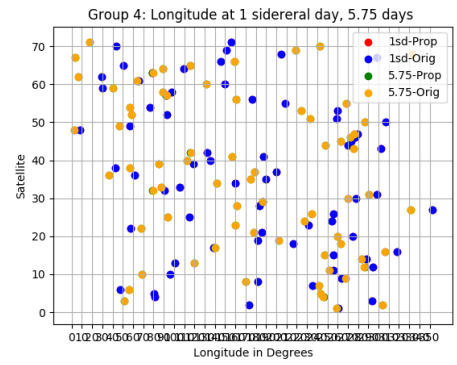


Figure 17: Gr 4

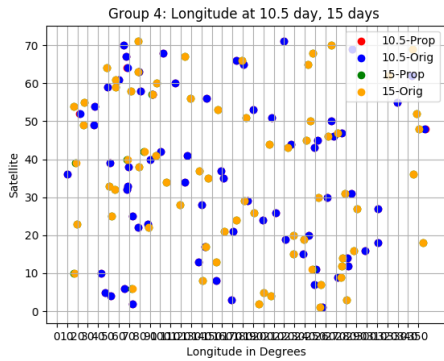


Figure 18: Gr 4

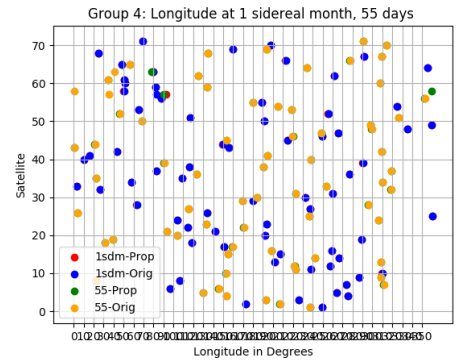


Figure 19: Gr 4

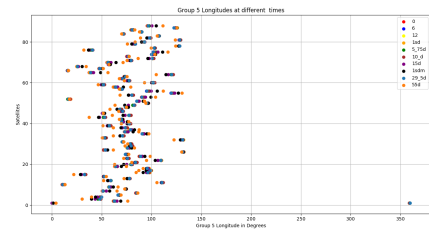


Figure 20: Gr 5

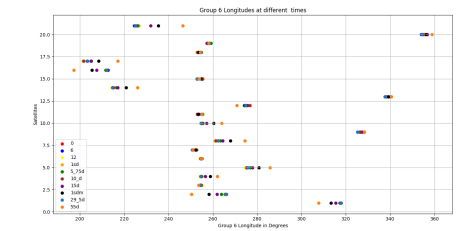


Figure 21: Gr 6

sorting the objects into the 8 groups.

However, this metric alone is not sufficient due to the unpredictable time horizon of the maneuver and lack of insight into the effect of each maneuver. This warranted investigation into other orbital characteristics of the objects, namely the orbital elements: semi-major axis, right ascension of ascending node, the argument of

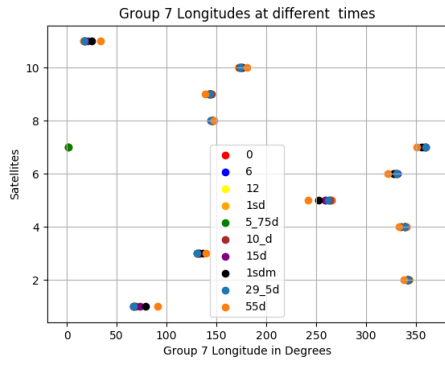


Figure 22: Gr 7

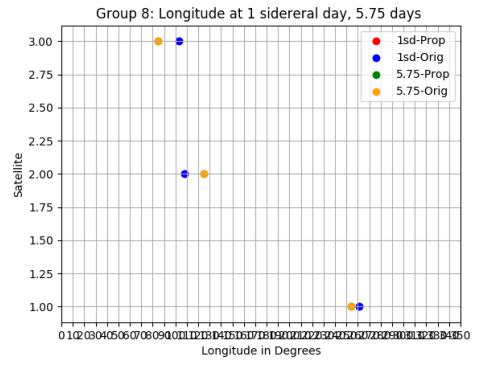


Figure 23: Gr 8

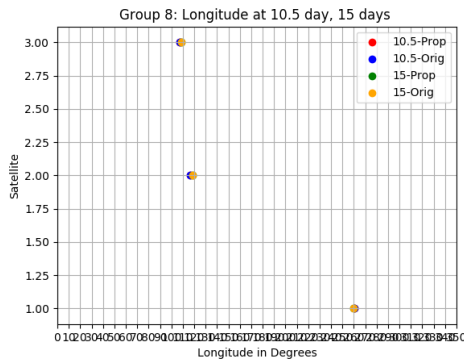


Figure 24: Gr 8

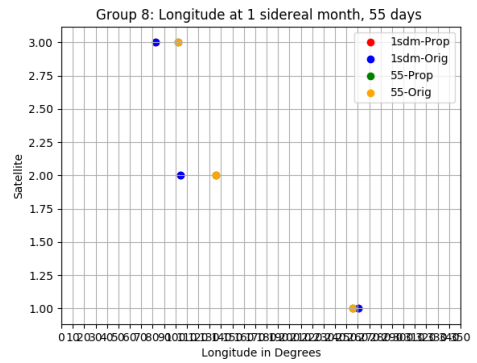


Figure 25: Gr 8

periapsis, inclination, eccentricity, and period. From the conclusion from longitudinal drift analysis, the study of orbital elements was focused on comparing data from 0-day TLE, 55-day TLE and propagated 0-day TLE to 55 days using SGP4.

It is apparent from the definition of Group 8 that its objects are easiest to segregate as they have a high inclination, and minimal longitudinal and inclination drift. These characteristics are easy to mathematically formulate and were used to obtain the objects in the group. For objects in Group 4, since the group consists of artificial satellites in drift orbits, so a lower limit on the change in expected longitude over 55 days was identified.

The rest of the satellites were checked to see if they have a very small longitudinal drift of order 1 because if the satellites are active then they will do a maneuver to decrease the drift. It was found that if the satellites have a drift of less than 17 deg then there is a high probability that they are active so these satellites lie in either Group 1, Group 2, or Group 3. Further, since Group 1 is also capable of inclination control, it was observed that the satellites in Group 1 had a larger increase in RAAN value than inclination so if the satellites have RAAN drift of more than 2.5 then they are in Group 1 otherwise Group 2. It is extremely difficult to segregate the satellites of Group 3 as the true set contains only 1 satellite in the group thus it is difficult to predict the general trend of the group.

The leftover satellites are checked if their propagated longitudes after 55 and 400 days from the 0-day epoch TLE file remain within the range of 50 degrees on either side from the stable libration point. If they are within 50 degrees on either side from 75 degrees then it is classified as group 5, if around 255 degrees then in group 6, otherwise in group 7.

4.1 Flowchart to explain classification logic (except Group 3)

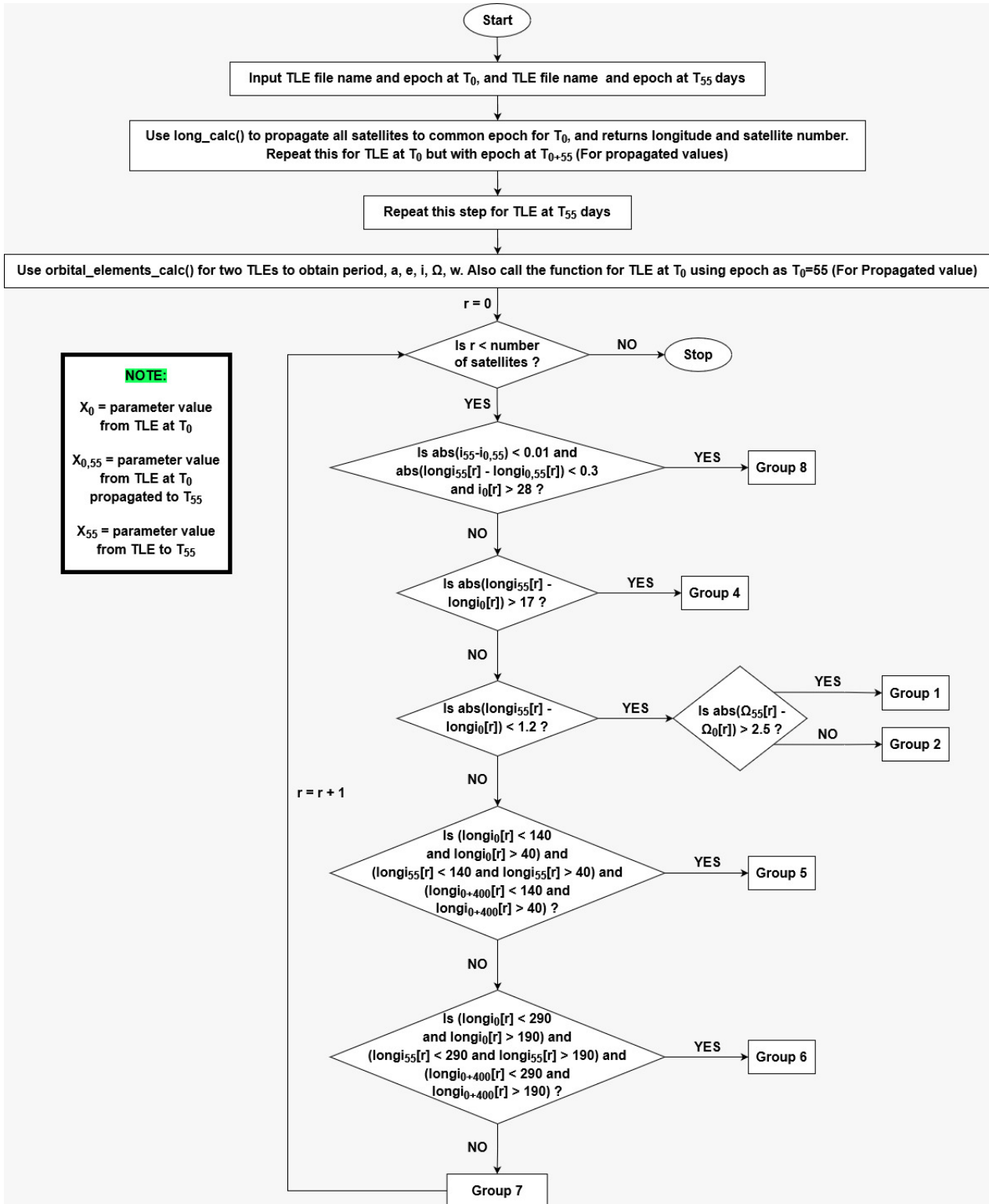


Figure 26: Classification logic

5 Results

The method devised was used to segregate the space objects in the report and then checked if they were accurately classified. It was found that the logic built the taxonomy with 95% accuracy. The major shortcoming of the method is the faulty categorization of a few of Group 1 satellites as Group 2. This is because the two

groups have very similar characteristics and it is until a satellite makes an inclination control, which cannot be known as it depends on the individual case, a satellite is categorized as Group 2. Thus, there are 10 satellites that were supposed to be in Group 1 but were categorized as Group 2. The only way to mitigate this error is by taking the second TLE after a longer time interval like after 70 days. The hope is that over the longer time interval, the Group 1 object will make an inclination change that can be captured in the data to be correctly categorized.

There has been no method developed to segregate Group 3 satellites because of the lack of data as there is only 1 satellite in Group 3 and that satellite has characteristics very similar to Group 2 as it drifts by a negligible amount.

Most of the Group 4 objects were correctly categorized except for 1998-057A, 25495 - EUTE 25A (HB5). It is incorrectly categorized under Group 2 because it only drifts by 0.4 deg over 55 days. As explained earlier Group 4 has a threshold of 17 deg, the threshold of 17 degrees cannot be lowered because then the other group's satellite will be listed as Group 4.

Another satellite that is not correctly segregated is 1993-013A, 22557 -Raduga 29. It should be in Group 5 but it shows an unusual behavior by drifting 17 deg.

6 Conclusions

Space Debris is a man-made disaster that needs to be dealt with very seriously to for the safe operation of space assets and to provide a safe path for rockets to escape earth's gravity. This is a small effort in the bigger picture to achieve better space traffic management.

The opportunity to work on Space Object Taxonomy has given me a better understanding of different types of space debris and satellites. I have learned how to use Latex and improved my coding skills in Python and Matlab. In addition, I have gained intuition of how the orbital elements of GEO objects vary in the Non-Keplerian model.

The method devised gives a good way to segregate satellites into the 8 groups defined by ESA in their CLASSIFICATION OF GEOSYNCHRONOUS OBJECTS Report. It was over several months of hard work and analysis of patterns of the truth set that I was able to come up with the smart conditions to make a code to autonomously segregate the satellites. However, there are a few shortcomings and outliers that have not been accounted for as explained in the results section.

Future work is to improve the conditions to segregate by testing the same set of conditions with a different issue of the ESA GEO Classification Report. This will increase the reliability of the method developed. Another important aspect that needs work is to come up with the conditions to segregate the satellites in Group 3.

7 Appendix

This section includes a brief explanation of the functions created for the analysis and the workflow to recreate the work.

7.1 Functions

7.1.1 Truth Set

- **cat1:** This is a helper function that consists of all the COSPAR satellite numbers for group 1 as listed in ISSUE 19 of the ESA GEO Classification Report. The COSPAR numbers are converted in a format similar to TLE by deleting the first two numbers and the hyphen in between the number. Similarly, **cat2,cat3...cat8** exist for group2, group3...group8 respectively. A similar version exists for ISSUE 15 Report.
- **common_elements:** This is a helper function that compares the satellite names of each group of truth set that is obtained by cat1 type function and the satellite numbers in TLEs. It passes the common satellite number and their respective TLE to 'exec.m'
- **common_elements_extra:** This function is similar to common_elements, it is used for the cases of COSPAR number which consists of 2 alphabets at the end instead of one.
- **cdm_rot:** This is a helper function that uses the cosine direction matrix to obtain the states of a 3D orbit in the ECI frame.
- **TLE_rv_plot:** Identifies the 5 orbital elements from the 2nd line of TLE. Then calls the cdm_rot to obtain various states of each orbit. Plots the orbits by the group.

- **exec:** Executive function that calls all the `cat1` type function and `common_elements.m` to obtain the satellite number and their respective TLE for each of the 8 groups. It uses `TLE_rv_plot` to plot all the satellites of each group with their respective group satellites and save each group's TLE into a different text file.
- **long_at_set_epoch:** Helper function that reads 8 TLE files that were generated using `exec.m`. Calculates the longitudes of the satellites and makes an array that stores all the file 1 satellite longitude and the number -1 at the end. Similarly, all the longitude of file 2 and -2 after that, and this goes on for the rest of the files till 8 and the last element of the array being -8. The negative numbers are used as a demarcation to know when the next file's longitude begins to appear in the same array.
- **long_at_set_epoch_for_satname:** Works similar to `long_at_set_epoch.py` but it saves the satellite number instead of longitude.
- **orbital_elements_at_set_epoch:** Works similar to `long_at_set_epoch_for_satname.py` but it creates multiple arrays to save period, semi-major axis, inclination angle, eccentricity, RAAN, and argument of periapsis.
- **exec_for_diff_epoch:** This function uses `long_at_set_epoch_for_satname.py`, `long_at_set_epoch_for_satname.py` and `orbital_elements_at_set_epoch.py` to get all the characteristics of a satellite. It is an easy-to-understand code, which is used to print different features of the orbit of a satellite.

7.1.2 For Method Devised

- **commonsat:** Helper function that reads through TLE files and creates 2 arrays, one consisting of all the satellite numbers and the other containing their respective TLEs.
- **sorting:** Executive function, the user has to change the two TLE file names at the required places as per need. This function obtains the satellite number and their respective TLE using `commonsat.m`, then finds the common satellite numbers in the two TLE files which are 55 days apart. It finds the common satellite number from the original array of the two TLE files and writes the respective TLE of the common satellites.
- **mathconstants:** Function defines a few of the math constants used in other functions. It was created by Prof. Frueh.
- **astrconstants:** Defines a few of the astronomy constants to be used in other functions. It was created by Prof. Frueh.
- **xvtokeplerorbit:** Converts the state vectors into the 6 keplerian orbital elements. It was created by Prof. Frueh but a similar function was created by me using slightly different formulas.
- **orbital_elements_at_set_epoch_master:** Helper function that propagates all the satellite of a TLE file to a given epoch and calculates the 'p' period, 'a' semi-major axis of the elliptical orbit of the satellite, 'e' eccentricity which defines how elliptical or circular the orbit is, 'i' inclination angle which is the angle between the earth equatorial plane and the orbital plane, 'Omega' Right ascension of ascending node is the angle between the vernal equinox and the line of the ascending node, and 'omega' Argument of periapsis is the angle from the line of ascending node to the eccentricity vector.
- **long_at_set_epoch_master:** Function propagates all the satellites to a given epoch and finds their state vectors. Thereafter, it calculates the angle between its X-Y plane projection and the vernal equinox. Then it finds the sidereal angle between the vernal equinox and GMT at the satellite epoch. In the end, it smartly takes the difference between the two angles to get the longitude.
- **exec_master_check_xxxx_yy:** Executive function that segregates the objects using the logic described earlier and it makes use of other above functions to get the required orbital elements and longitude of the satellites. `xxxx` stands for year and `yy` stands for date of TLE at T0.
- **cat1:** Consists of all the COSPAR satellite numbers for group 1 listed in ISSUE 19 of ESA GEO Classification Report. The COSPAR numbers are converted in a format similar to TLE by deleting the first two numbers and the hyphen in between the number. Similarly, `cat2,cat3...cat8` exist for group2, group3...group8 respectively. A similar version exists for ISSUE 15.
- **common_elements:** Compares the satellite names of each group truth set that is obtained by `cat1` type function and the satellite number of all the satellites in the TLE file given as input. It passes the common satellite number and their respective TLE to '`exec.m`'.

- **common_elements_extra:** Similar to common_elements, it is used for the cases of COSPAR number which consists of 2 alphabets at the end instead of one.
- **sat_raw_extract:** MATLAB function reads the text file created by exec_master_check_xxxx.yy.py and saves the satellite numbers of different groups in different rows of a cell array.
- **python_sorted_group_satid:** MATLAB function creates a cell array that contains the element of catx.m.
- **compare_master_xxxx:** Executive MATLAB program obtains 2 different cell arrays from sat_raw_extract.m and python_sorted_group_satid.m to come up with outliers and it writes the number of outliers in an excel file. xxxx is the year of TLE

7.2 Workflow

7.2.1 Setup For Truth set:

1. Download satellites TLE from spacetrack.org, log in using id, click on the TLE search tab, enter '1-40000' in entries, choose a date range, and enter the range required. Follow this step twice, once for T0 and for T55. Copy the data from API into a text file.
2. Use the Matlab function exec.m. Change the file name to the file name under which all the TLEs are saved. After running the function, 8 plots and 8 text files will be generated. Each of the plots and files will correspond to their respective group number. As explained earlier 8 functions need to be made which consist of the COSPAR number of satellites of their respective group.
3. Use exec_for_diff_prop to obtain the longitude and orbital elements of satellites at different epochs and from their propagated epoch value. It serves as a platform to compare propagated and actual parameter values of T55 days, and the actual values of T0 and T55 days.

7.2.2 Method Devised

1. Use sorting.m with TLE files at T0 and T55 as input to get two files that consist of the same set of satellites at epochs 55 days apart.
2. Use the two files generated as input for exec_master_check_xxxx.yy.py to segregate the satellites in their respective groups and a sat_sort_py.txt is created with the sorted satellites of different groups with a unique delimiter at the end of each group.
3. Run compare_master_xxxx.m with sat_sort_py.txt as the input, the program will generate all the outliers by comparing the sorted list of satellites to the ones obtained from GEO Classification Report. It writes the number of outliers to an excel file called Outlier_num.xlsx. The excel file has a table where (3,2) cell indicates the number of satellites that are listed as group 3 but should be part of group 2.