Forensic Analysis of TLE Satellite Data

Nitzan Schwartz

Supervised by: Dr. Vladimir Martinusi

Contents

Introduction

The Two-Line Element (TLE) data format is instrumental in meticulously tracking the positions and trajectories of satellites and other celestial entities orbiting Earth. It provides a structured set of parameters crucial for precise tracking and prediction, including orbital elements, epoch time, security classification, and drag term. TLEs are available in two varieties: comprehensive lists of diverse satellites and their corresponding parameters, and historical TLEs, which depict the trajectory of a single satellite over time. Due to the significant volume of data in these lists, extracting specific information about a particular space object can be challenging. However, the insights gleaned from these data sets are invaluable, as historical TLEs can reveal the trajectory types and behaviors of space bodies, enabling the identification of patterns, maneuvers, and potentially suspicious activities.

This report focuses on the analysis of two NAVSTAR GPS satellites, which serve as exemplars for extracting insights into satellite maneuvers using TLE data.

Analysis Method and Theory

As we analyze a satellite's TLE data, we want to look for any hints for maneuvers. The orbital parameter which is the primary indicator of a maneuver is the semi – major axis. As this increases or decreases, it can be concluded that the satellite's altitude has been shifted accordingly.

Once spotting a drastic change in the semi-major axis, we would like to validate the likelihood of a maneuver by analyzing the other orbital elements of the satellite. As the semi-major axis shifts, we would expect that the other 5 orbital elements- Eccentricity, Inclination, Right Ascension of the Ascending Node (RAAN), Argument of the Perigee and the Mean Anomaly - would shift too as they are dependent on one another. That dependency can be seen in the following equations:

The specific angular momentum is defined by:

$$h = \vec{r} \times \vec{v}$$

The specific mechanical energy:

$$\epsilon = \frac{v^2}{2} - \frac{\mu}{r}$$

Where: $\mu = 3.986 \cdot 10^5 \left[\frac{km^3}{s^2}\right]$.

Eccentricity:

$$e = \sqrt{1 + \frac{2\epsilon h^2}{\mu^2}}$$

Inclination:

$$\cos(i) = \frac{h_z}{h}$$

The RAAN and argument of the perigee are defined as the following:

$$\cos(\Omega) = \frac{n_x}{n}$$

$$\cos(\omega) = \frac{\vec{n} \cdot \vec{e}}{ne}$$

Where: $\vec{n} = \hat{z} \times \vec{h}$

($\hat{z}$ is a unit vector in the inertial z direction.)

The Mean Anomaly can be calculated iteratively using the following equations:

$$M = E - esin(E)$$

$$\cos(E) = \frac{e + \cos(f)}{1 + ecos(f)}$$

$$\cos(f) = \frac{\vec{r} \cdot \vec{e}}{re}$$

Where $\vec{r}$ is the position vector of the satellite.

The analysis involves detecting simultaneous changes in all orbital parameters, indicating potential maneuvers. By scrutinizing these parameters collectively, the likelihood of a maneuver can be validated in our quest to detect one.

MATLAB Code

Two main codes were developed for this project to facilitate the desired analysis. The first code sifts through TLEs of different satellites based on user-defined parameters, enabling the selection of satellites with trajectories typical of GPS satellites. The output of this code is a list of satellites that meet the specified criteria. Subsequently, an intermediate Python code processes this list, accepting satellite designator numbers as input and generating a text document containing historic TLE data for each satellite. The second MATLAB code is then employed, taking as input any historic TLE text file. It parses the data to make it accessible for analysis.

Specifically, for this project, the input ranges we were interested in for the first script were the following:

$$20{,}000\ [Km] < a < 20{,}800\ [Km]$$

$$50^\circ < i < 65^\circ$$

$$0 < e < 0.01$$

These ranges are typical for GPS satellites, therefore suitable for our project.

From the compiled list of satellites, our focus was directed towards those displaying substantial alterations in the semi-major axis. Among these candidates, we specifically identified satellites where concurrent changes occurred in other orbital parameters for the same time as the change in the semi-major axis. The output generated by the second code facilitated the plotting of orbital elements over time (Epoch) for further analysis.

These MATLAB scripts can be viewed at the appendix.

Results

The results of the TLE forensic analysis on two NAVSTAR GPS satellites, NAVSTAR 47 and NAVSTAR 43, reveal suspicious potential maneuvers. Both satellites of a nominal semi-major axis of $26{,}560\ [Km]$. It is important to note that we only portrayed the orbital elements which values differed at a close time to the differ in the semi-major axis, which as stated, is the primary indicator of a maneuver.

NAVSTAR 47

NAVSTAR 47, the latest addition to the American GPS fleet of navigation satellites, launched in May 2000. It's trajectory is tracked from March 24, 2012, to December 11, 2023. An analysis of its TLE data reveals several notable changes in the semi-major axis, suggestive of potential maneuvers.

Changes in the Semi-Major Axis

Visual examination of **Error! Reference source not found.** depicts sudden shifts in the semi-major axis over time, indicating potential maneuvers.

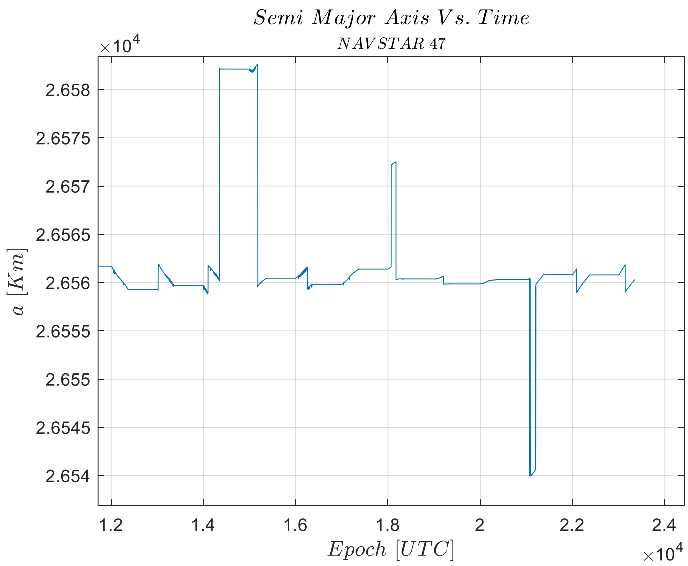

Figure 1 - NAVSTAR 47 Semi - Major Axis as a Function of Time

The shifts details listed in chronological order:

1. From December 17, 2014, to June 26, 2015: An increase of 20 km.
2. From March 10, 2018, to June 22, 2018: An increase of 12 km.
3. From March 18, 2021, to July 25, 2021: A decrease of 20 km.

It is noteworthy that after each maneuver the satellite returns to its nominal semi-major axis value of $26{,}560\ [km]$.

Validation of Maneuvers

To validate the likelihood of maneuvers, the analysis extends to other orbital elements, notably in our case, the argument of the perigee (ω), and the eccentricity. As illustrated in Figure 2 and in Figure 3, changes in ω coincide with the observed shifts in the semi-major axis:

- On December 17th, 2014, a slight decrease in ω is noted, where $\Delta\omega = 76° - 70° = 6°$, corresponding to the first change in the semi-major axis.
- A similar pattern is observed on June 26, 2015, with another sudden decrease in ω , where $\Delta\omega = 69.5° - 65.7° = 3.8°$. Subsequently, the value of ω gradually increases until the end of the data period in 2023.

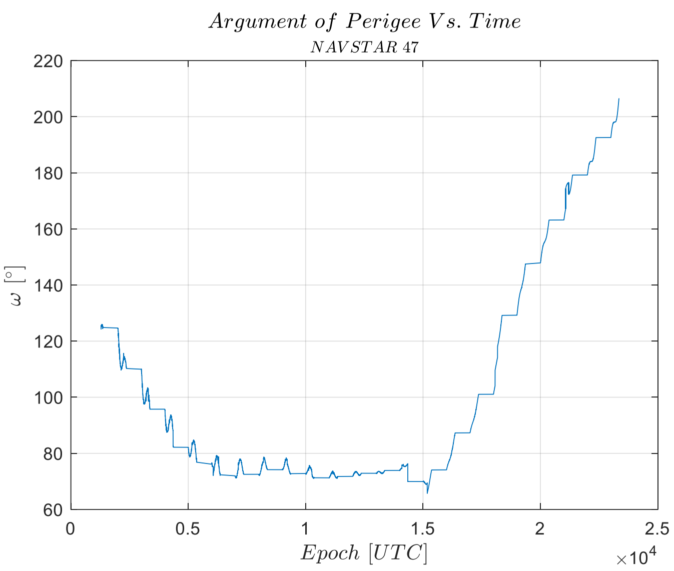

Figure 2 - NAVSTAR 47 - argument of perigee as a Function of Time

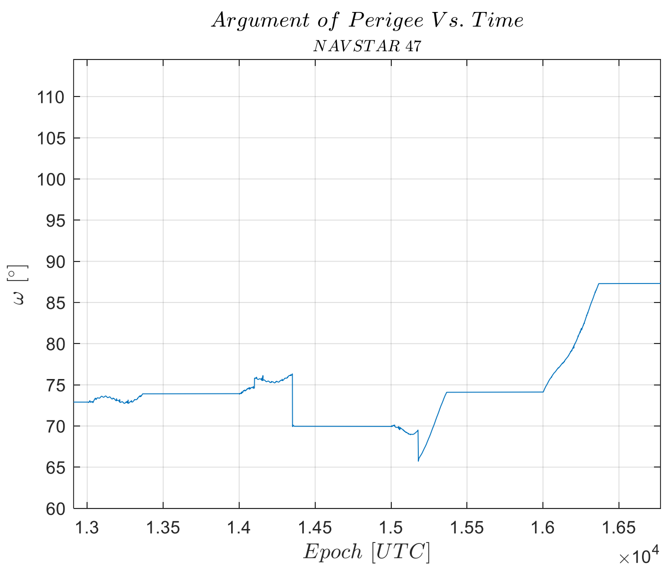

Figure 3 - NAVSTAR 47 - Argument of Perigee Close Up

Further analysis reveals changes in eccentricity, as depicted in Figure 5 and in Figure 4. Between June 5, 2014, and December 16, 2014, there is an increase in eccentricity: $\Delta e = 0.0061 - 0.0055 = 6 \cdot 10^{-4}$. This additional observation reinforces the presence of significant orbital dynamics during the identified time periods.

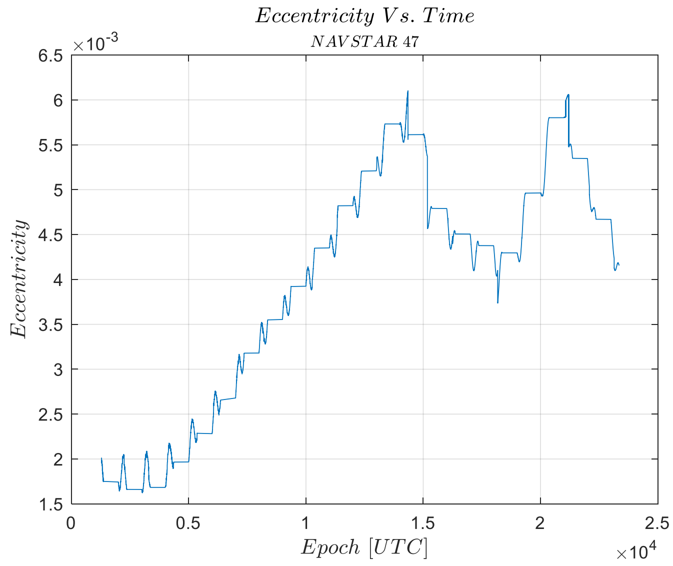

Figure 5 - NAVSTAR 47 - Eccentricity as a Function of Time

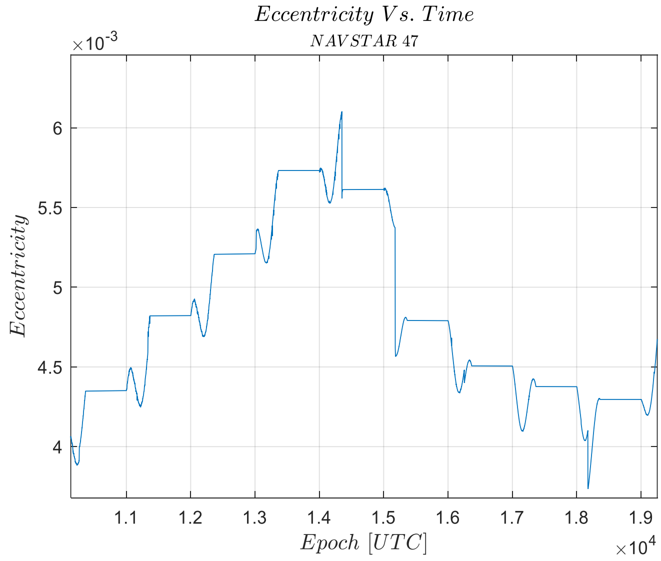

Figure 4 - NAVSTAR 47 Eccentricity Close Up

NAVSTAR 43

NAVSTAR 43, designated as #24876, is also a GPS satellite developed by the US Department of Defense. Launched on July 23rd, 1997, from Cape Canaveral, this satellite's trajectory is tracked over a ten-year period from January 13, 2013, to December 10, 2023. An analysis of its TLE data reveals significant orbital dynamics and potential maneuvers during this period.

Changes in the Semi-Major Axis

Figure 6 illustrates two apparent changes in the semi-major axis. The first change, observed from the left, appears to be a transient spike, likely attributable to measurement noise. However, the second change, spanning from September 27, 2014, to January 9, 2015, suggests a maneuver, as the satellite maintains its new altitude for the subsequent period. The magnitude of this change is calculated as: $\Delta a = 26583.1 - 26560.4 = 22.7\ [Km]$.

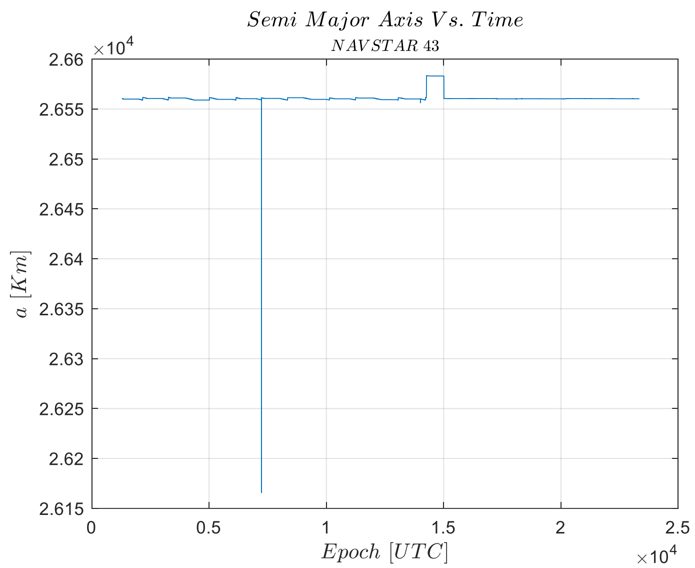

Figure 6 - NAVSTAR 43 Semi - Major Axis as a Function of Time

Validation of Maneuvers

The analysis extends to other orbital elements, one them being the argument of the perigee (ω), as depicted in Figures 6 and 5. A notable change of approximately 10° in ω is observed between February 23, 2014, and December 31, 2014. While the exact timing does not align precisely with the changes in the semi-major axis, there is some overlap, indicating potential correlations.

The Eccentricity graph portrays a change between 2014/09/27 and 2015/01/05 which is roughly the same dates as the change in a. The difference in the eccentricity is $\Delta e = 0.00567 - 0.00507 = 6 \cdot 10^{-4}$.

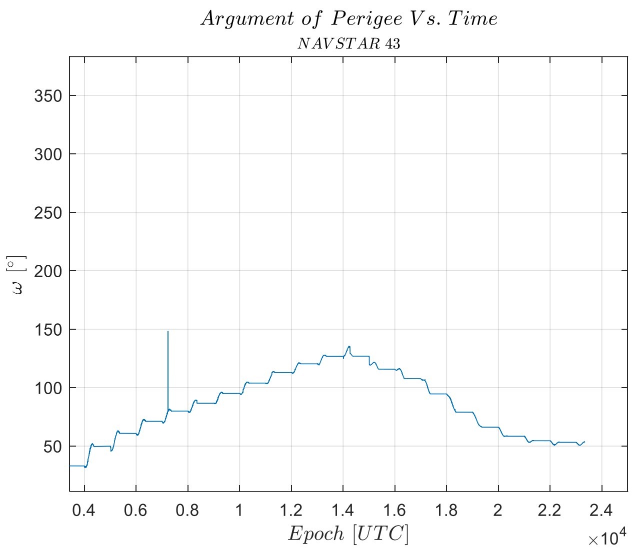

Figure 8 - NAVSTAR 43 - Argument of the Perigee as a Function of Time

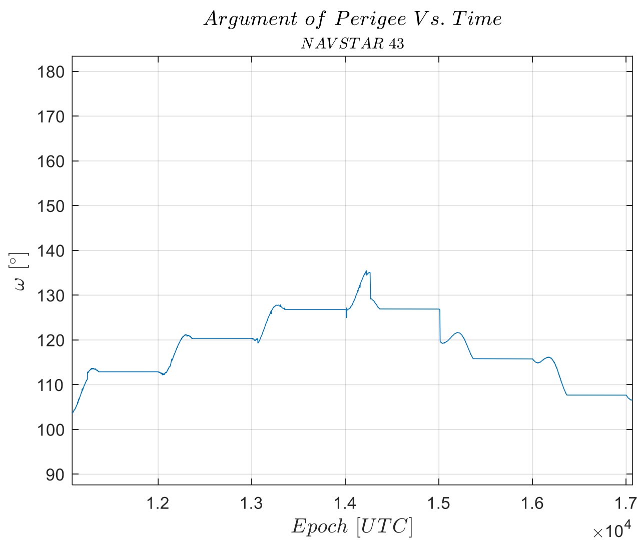

Figure 7- NAVSTAR 43 - Argument of the Perigee Close Up

Examination of the eccentricity graph portrayed in Figure 10 and in Figure 9 reveals a significant change between September 27, 2014, and January 5, 2015, coinciding closely with the observed change in the semi-major axis. The difference in eccentricity during this period is calculated as:

$\Delta e = 0.00567 - 0.00507 = 6 \cdot 10^{-4}$, further supporting the occurrence of maneuvers.

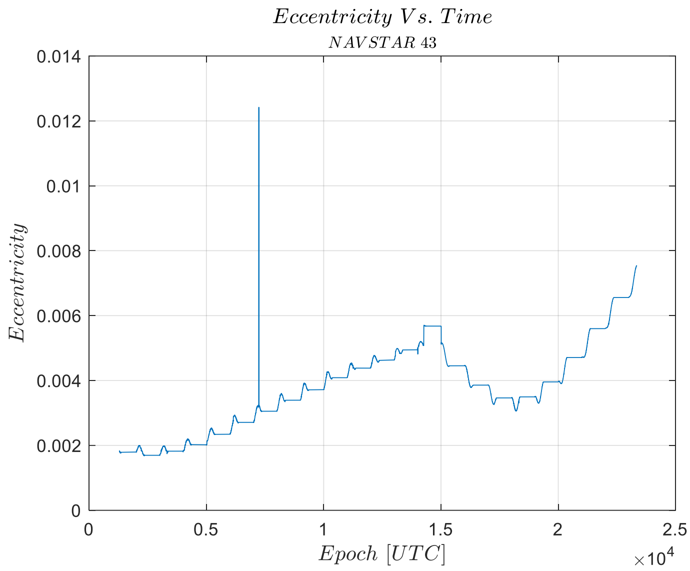

Figure 9 - NAVSTAR 43 - Eccentricity as a function of time

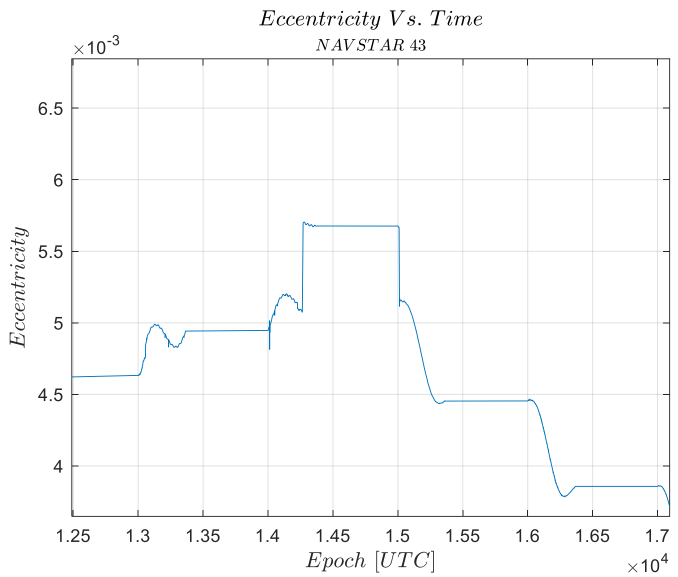

Figure 10 - NAVSTAR 43 - Eccentricity Close Up

Summery

The forensic analysis of Two-Line Element (TLE) data represents a simple approach to understanding satellite behaviors in orbit, offering valuable insights into potential maneuvers, in addition to tother valuable data. This comprehensive analysis, while immensely powerful, also underscores the complexity and challenges inherent in interpreting orbital dynamics.

One of the primary advantages of TLE forensic analysis lies in its ability to detect subtle changes in orbital parameters, such as the semi-major axis, eccentricity, inclination, and argument of perigee. These parameters serve as key indicators of satellite maneuvers, including orbital adjustments, rendezvous operations, and station-keeping maneuvers. However, it is crucial to acknowledge that detecting changes in the semi-major axis does not always correspond to observable alterations across all orbital elements.

As presented in this report, the analysis of TLE data for NAVSTAR 47 and NAVSTAR 43 reveals significant changes in the semi-major axis, indicative of potential maneuvers. However, this observed change in the semi-major axis does not necessarily translate to discernible alterations across all orbital elements. Instead, we may only observe changes in specific parameters, notably the eccentricity and argument of perigee, while others remain relatively stable.

For instance, while examining the TLE data for NAVSTAR 47, changes in the semi-major axis coincide with variations in eccentricity and, to some extent, the argument of the perigee. These alterations provide additional context and validation for the observed maneuvers. Similarly, in the case of NAVSTAR 43, changes in the semi-major axis are accompanied by noticeable shifts in eccentricity, while the correlation with the argument of perigee remains debatable due to discrepancies in timing.

This nuanced understanding underscores the intricate interplay between orbital parameters and the challenges associated with interpreting TLE data comprehensively. While changes in the semi-major axis serve as primary indicators of maneuvers, we must exercise caution and consider the broader context, including potential discrepancies or inconsistencies across other orbital elements.

Despite these challenges, TLE forensic analysis remains an invaluable tool for enhancing the tracking of trajectories for both active and inactive space bodies. Its utilization, when combined with complementary analytical techniques and algorithms, spans a wide range of applications. These applications include tracking adversary satellites, mitigating collision risks between space bodies, and ultimately yielding high-quality results that contribute to improved space situational awareness and operational safety.

Appendix – MATLAB Code

Sifting through TLE file containing different satellites in every row.

Contents

Input: Range of inclination, eccentricity and altitude.

Output: The matching satellite catalogs.

```
clc
clear all
close all

% arranging the TLE data in three different vectors:
% 1. Satellite Name
% 2. First line of TLE:
% - Line Number
% - Satellite Catalog and Classification
% - International Designator
% - Epoch
% - 1st Derivative of mean motion
% - 2nd Derivative of mean motion
% - BSTAR Drag Term,Ephemeris type
% - Element number
% - Checksum
% 3. Second line of TLE:
% - Line Number
% - Satellite Catalog
% - Inclination, i [deg]
% - RAAN, Omega [deg]
% - eccentricity, e
% - Argument of Perigee, omega [deg]
% - Mean Anomaly, M [deg]
% - Mean Motion, n [rev/day]

%constants
mu = 3.986e5; %[km^3/s^2]
Re = 6378; %[km]

TLE = "celes_tle.json"; %this file needs to be downlaoded as json using a script in python
txt = fileread(TLE);
data = jsondecode(txt);

%initializing vectors
Sat_Name = string(NaN(length(data),1));
TLE_1 = string(NaN(length(data),1));
TLE_2 =  string(NaN(length(data),1));


%converting the TLE data to vector of type double
for i=1:length(data)
    Sat_Name(i,:) =  data{i, 1}.satellite_name;
    TLE_1(i,:) = data{i, 1}.tle_1;
    TLE_2(i,:) = data{i, 1}.tle_2;
    temp_string = strsplit(TLE_2(i,:));
    split_temp(i,:) = temp_string(1:8);
    split_temp(i,5) = "0." + split_temp(i,5);
    TLE2_param(i,:) = str2double(split_temp(i,:));
    n(i) = (TLE2_param(i,8))*2*pi/(24*60*60); %mean motion [1/sec]
    e(i) = TLE2_param(i,5);
    a(i) = (mu/(n(i)^2))^(1/3); %semi major axis [Km]
    h(i) = a(i)- Re; %altitude [Km]
    rp(i) = a(i)*(1-e(i)); %[Km]
end

i_min = input("Please enter min value of the inclination (in degrees) : ");
i_max = input("Please enter max value of the inclination (in degrees) : ");

e_min = input("Please enter min value of the eccentricity: ");
```

```
e_max = input("Please enter max value of the eccentricity: ");

h_min = input("Please enter min value of the altitude (in Km) : ");
h_max = input("Please enter max value of the altitude (in Km) : ");

flag = zeros(length(data),1);

% trimmed_TLE2 = TLE2_param(TLE2_param>i_min & TLE2_param<i_max & TLE2_param<e_min TLE2>e_max)
for i=1:length(data)
    if TLE2_param(i,5)>e_min && TLE2_param(i,5)<e_max && TLE2_param(i,3)>i_min && TLE2_param(i,3)<i_max && h_min<h(i) && h(i)<h_max
        flag(i) = 1;
    end
end

TLE2_param_sifted = TLE2_param(flag==1,:);
Satellite_Name = Sat_Name(flag==1);
Satellite_Catalog = TLE2_param_sifted(:,2);
Sifted_Sats = [Satellite_Name,Satellite_Catalog];

if isempty(Satellite_Catalog)
    disp('No satellites meet the specifies input conditions')
else
    fprintf('There are %d satellites that match the given parameters and they are:\n ',length(Satellite_Catalog));
    disp(Sifted_Sats)
end
```

Contents


```
function [a,e,i,Om,om,n,M,Epoch_Vec,rp] = prs_TLE (TLE_Designator)
```

Input: TLE txt file designator number as string. For example, '26360'

Output: Orbital Elements and epoch vector

```
file = strcat(TLE_Designator,'.txt');

mu = 3.986e5; %[km^3/s^2]
Re = 6378; %[km]
```

Importing the TLE txt file, dividing it into two rows, and labeling the variables

```
TLE = readtable(file,'Delimiter',' ', 'MultipleDelimsAsOne', true, 'Format','%d%s%s%f%s%s%s%f%f');

TLE1 = TLE(1:2:end,:);
TLE_1 = renamevars(TLE1,["Var2","Var3", "Var4", "Var5", "Var6", "Var7","Var8","Var9"],...
    ["Satellite_Number", "International_Designator","Epoch", "1st_Derivative_of_Mean_Motion","2nd_Derivative_of_Mean_Motion","Drag_Term_or_Radiation_Pressure_Coeffi
TLE2 = TLE(2:2:end,:);
TLE_2 = renamevars(TLE2,["Var2","Var3", "Var4", "Var5", "Var6", "Var7","Var8","Var9"],...
    ["Satellite_Number", "Inclination","RAAN", "Eccentricity","Argument_of_Perigee","Mean_Anomaly",...
    "Mean_Motion", "Revolution#_at_Epoch&Check_Sum"]);
```

Fixing the Eccentricity value and converting all relevent values to be double/string

```
TLE_2 = convertvars(TLE_2,{'Satellite_Number','Inclination','Eccentricity','Argument_of_Perigee','Mean_Anomaly'},'string');
for i=1:height(TLE_2)
    TLE_2{i,5} = strcat('0.',TLE_2{i,5});
end
TLE_2 = convertvars(TLE_2,{'Satellite_Number','Inclination','RAAN','Eccentricity','Argument_of_Perigee','Mean_Anomaly',...
    'Mean_Motion','Revolution#_at_Epoch&Check_Sum'}, 'double');

TLE_1 = convertvars(TLE_1,{'Satellite_Number', 'International_Designator','Epoch', '1st_Derivative_of_Mean_Motion',...
    '2nd_Derivative_of_Mean_Motion','Drag_Term_or_Radiation_Pressure_Coefficient','Ephemeris_Type', 'Element_Number&Check_Sum'},'string');
for i=1:height(TLE_1)
    TLE_1{i,5} = strcat('0',TLE_1{i,5});
end
TLE_1 = convertvars(TLE_1,{'Epoch', '1st_Derivative_of_Mean_Motion','2nd_Derivative_of_Mean_Motion',...
    'Drag_Term_or_Radiation_Pressure_Coefficient','Ephemeris_Type', 'Element_Number&Check_Sum'},'double');

n = ((TLE_2.Mean_Motion)./(24*60*60))*2*pi; %mean motion [1/sec]
a = (mu./(n.^2)).^(1/3); %semi major axis [Km]
%     h(i) = a(i)- Re; %altitude [Km]
rp = a.*(1-TLE_2.Eccentricity); %[Km]
M = deg2rad(TLE_2.Mean_Anomaly); %[rad]
i = deg2rad(TLE_2.Inclination); %[rad]
Om = deg2rad(TLE_2.RAAN); %[rad]
om = deg2rad(TLE_2.Argument_of_Perigee); %[rad]
e = TLE_2.Eccentricity;
Epoch_Vec = TLE_1.Epoch; %[UTC]
```

Contents


```
clc;close all;
```

Calculating Orbital Parameters of Desired Satellites

all angles in radians and distance in [Km]

```
[a_NS47,e_NS47,i_NS47,Om_NS47,om_NS47,n_NS47,...
    M_NS47,Epoch_NS47,rp_NS47] = prs_TLE ('26360'); %NAVSTAR 47 - Active, latest addition to the GPS fleet)

% NAVSTAR 47 Analysis

figure
plot(Epoch_NS47(300:end),a_NS47(300:end))
xlabel('$Epoch\ [UTC]$','interpreter','latex','FontSize',12')
ylabel('$a\ [Km]$','interpreter','latex','FontSize',12')
title('$Semi\ Major\ Axis\ Vs.\ Time$','interpreter','latex','FontSize',12')
subtitle('$NAVSTAR\ 47$' ,'interpreter','latex','FontSize',9)
grid on

figure
plot(Epoch_NS47(300:end),n_NS47(300:end))
xlabel('$Epoch\ [UTC]$','interpreter','latex','FontSize',12')
ylabel('$n\ [rad/sec]$','interpreter','latex','FontSize',12')
title('$Mean \ Motion\ Vs.\ Time$','interpreter','latex','FontSize',12')
subtitle('$NAVSTAR\ 47$' ,'interpreter','latex','FontSize',9)
grid on


figure
plot(Epoch_NS47(300:end),rad2deg(om_NS47(300:end)))
xlabel('$Epoch\ [UTC]$','interpreter','latex','FontSize',12')
ylabel('$\omega\ [{^\circ}]$','interpreter','latex','FontSize',12')
title('$Argument \ of\ Perigee\ Vs.\ Time$','interpreter','latex','FontSize',12')
subtitle('$NAVSTAR\ 47$' ,'interpreter','latex','FontSize',9)
grid on

figure
plot(Epoch_NS47(300:end),rad2deg(Om_NS47(300:end)))
xlabel('$Epoch\ [UTC]$','interpreter','latex','FontSize',12')
ylabel('$\Omega\ [{^\circ}]$','interpreter','latex','FontSize',12')
title('$RAAN\ Vs.\ Time$','interpreter','latex','FontSize',12')
subtitle('$NAVSTAR\ 47$' ,'interpreter','latex','FontSize',9)
grid on

figure
plot(Epoch_NS47(300:end),e_NS47(300:end))
xlabel('$Epoch\ [UTC]$','interpreter','latex','FontSize',12')
ylabel('$Eccentricity$','interpreter','latex','FontSize',12')
title('$Eccentricity \ Vs.\ Time$','interpreter','latex','FontSize',12')
subtitle('$NAVSTAR\ 47$' ,'interpreter','latex','FontSize',9)
grid on

figure
plot(Epoch_NS47(300:end),rad2deg(i_NS47(300:end)))
xlabel('$Epoch\ [UTC]$','interpreter','latex','FontSize',12')
ylabel('$i\ [{^\circ}]$','interpreter','latex','FontSize',12')
title('$Inclination \ Vs.\ Time$','interpreter','latex','FontSize',12')
subtitle('$NAVSTAR\ 47$' ,'interpreter','latex','FontSize',9)
grid on

figure
plot(Epoch_NS47(300:end),rad2deg(M_NS47(300:end)))
xlabel('$Epoch\ [UTC]$','interpreter','latex','FontSize',12')
ylabel('$M\ [{^\circ}]$','interpreter','latex','FontSize',12')
title('$Mean Anomaly \ Vs.\ Time$','interpreter','latex','FontSize',12')
subtitle('$NAVSTAR\ 47$' ,'interpreter','latex','FontSize',9)
grid on
```

NAVSTAR 43 Analysis

```
[a,e,i,Om,om,n...
    ,M,Epoch,rp] = prs_TLE ('24876'); %NAVSTAR 43


figure
plot(Epoch(300:end),a(300:end))
xlabel('$Epoch\ [UTC]$','interpreter','latex','FontSize',12')
ylabel('$a\ [Km]$','interpreter','latex','FontSize',12')
title('$Semi\ Major\ Axis\ Vs.\ Time$','interpreter','latex','FontSize',12')
subtitle('$NAVSTAR\ 43$' ,'interpreter','latex','FontSize',9)
grid on


figure
plot(Epoch(300:end),rad2deg(om(300:end)))
xlabel('$Epoch\ [UTC]$','interpreter','latex','FontSize',12')
ylabel('$\omega\ [{^\circ}]$','interpreter','latex','FontSize',12')
title('$Argument \ of\ Perigee\ Vs.\ Time$','interpreter','latex','FontSize',12')
subtitle('$NAVSTAR\ 43$' ,'interpreter','latex','FontSize',9)
grid on

figure
plot(Epoch(300:end),rad2deg(Om(300:end)))
xlabel('$Epoch\ [UTC]$','interpreter','latex','FontSize',12')
ylabel('$\Omega\ [{^\circ}]$','interpreter','latex','FontSize',12')
title('$RAAN\ Vs.\ Time$','interpreter','latex','FontSize',12')
subtitle('$NAVSTAR\ 43$' ,'interpreter','latex','FontSize',9)
grid on

figure
plot(Epoch(300:length(e)),e(300:end))
xlabel('$Epoch\ [UTC]$','interpreter','latex','FontSize',12')
ylabel('$Eccentricity$','interpreter','latex','FontSize',12')
title('$Eccentricity \ Vs.\ Time$','interpreter','latex','FontSize',12')
subtitle('$NAVSTAR\ 43$' ,'interpreter','latex','FontSize',9)
grid on

figure
plot(Epoch(300:end),rad2deg(i(300:end)))
xlabel('$Epoch\ [UTC]$','interpreter','latex','FontSize',12')
ylabel('$i\ [{^\circ}]$','interpreter','latex','FontSize',12')
title('$Inclination \ Vs.\ Time$','interpreter','latex','FontSize',12')
subtitle('$NAVSTAR\ 43$' ,'interpreter','latex','FontSize',9)
grid on

figure
plot(Epoch(300:end),rad2deg(M(300:end)))
xlabel('$Epoch\ [UTC]$','interpreter','latex','FontSize',12')
ylabel('$M\ [{^\circ}]$','interpreter','latex','FontSize',12')
title('$Mean Anomaly \ Vs.\ Time$','interpreter','latex','FontSize',12')
subtitle('$NAVSTAR\ 43$' ,'interpreter','latex','FontSize',9)
grid on
```

Useful code for interpolation and Epoch Merging (not used in the report)

%% Merging the Epoch vectors

merged_Epoch = zeros(1,length(Epoch_NS47)+length(Epoch_NS55)); cnt_47 = 1; cnt_55 = 1;

for i=1:length(merged_Epoch) if cnt_47>length(Epoch_NS47) merged_Epoch(i) = Epoch_NS55(cnt_55); cnt_55 = cnt_55+1; elseif cnt_55>length(Epoch_NS55) merged_Epoch(i) = Epoch_NS47(cnt_47); cnt_47 = cnt_47+1; elseif Epoch_NS47(cnt_47)<= Epoch_NS55(cnt_55) merged_Epoch(i) = Epoch_NS47(cnt_47); cnt_47 = cnt_47+1; elseif Epoch_NS55(cnt_55)<= Epoch_NS47(cnt_47) merged_Epoch(i) = Epoch_NS55(cnt_55); cnt_55 = cnt_55+1; end end

```
% [x_NS47,y_NS47,z_NS47] = orb2cart(a_NS47,e_NS47,i_NS47,Om_NS47,om_NS47,M_NS47);
% [x_NS55,y_NS55,z_NS55] = orb2cart(a_NS55,e_NS55,i_NS55,Om_NS55,om_NS55,M_NS55);

% N = round(length(merged_Epoch)/length(a_NS47));
% M = round(length(merged_Epoch)/length(a_NS55));
%
% a_NS47_interp = interp(a_NS47,N);e_NS47_interp = interp(e_NS47,N);i_NS47_interp = interp(i_NS47,N);...
%     Om_NS47_interp = interp(Om_NS47,N); om_NS47_interp = interp(om_NS47,N);M_NS47_interp = interp(M_NS47,N);
%
% [x_NS47_interp,y_NS47_interp,z_NS47_interp] = orb2cart(a_NS47_interp,e_NS47_interp,i_NS47_interp,Om_NS47_interp,om_NS47_interp,M_NS47_interp);
%
% a_NS55_interp = interp(a_NS55,M);e_NS55_interp = interp(e_NS55,M);i_NS55_interp = interp(i_NS55,M);...
%     Om_NS55_interp = interp(Om_NS55,M); om_NS55_interp = interp(om_NS55,M);M_NS55_interp = interp(M_NS55,M);
% [x_NS55_interp,y_NS55_interp,z_NS55_interp] = orb2cart(a_NS55_interp,e_NS55_interp,i_NS55_interp,Om_NS55_interp,om_NS55_interp,M_NS55_interp);
```

```
%interpolation integrity test
% figure()
% plot(Epoch_NS47,a_NS47)
% hold on
% plot(merged_Epoch,a_NS47_interp(1:length(merged_Epoch)))
% legend('initial','interpolated')
% grid on

% interp_x_NS47 = interp(x_NS47,N);interp_y_NS47 = interp(y_NS47,N);...
%     interp_z_NS47 = interp(z_NS47,N);
% interp_x_NS55 = interp(x_NS55,M);interp_y_NS55 = interp(y_NS55,M);...
%     interp_z_NS55 = interp(z_NS55,M);
% interp_x_NS47_b = interp_x_NS47(1:length(interp_x_NS55));interp_y_NS47_b = interp_y_NS47(1:length(interp_y_NS55));...
%     interp_z_NS47_b = interp_z_NS47(1:length(interp_z_NS55));
%
% rel_x = x_NS47_interp(1:length(x_NS55_interp))-x_NS55_interp;
% rel_y = y_NS47_interp(1:length(y_NS55_interp))-y_NS55_interp;
% rel_z = z_NS47_interp(1:length(z_NS55_interp))-z_NS55_interp;
%
% rel_d = sqrt((rel_x).^2+(rel_y).^2+(rel_z).^2);
```