[redacted] - Fall 2019 Project
Simulations of Mars Landing

1 Introduction

In this project, you are to simulate the trajectory of several capsule landing on Mars. At initial time, the capsule trajectories are produced by the gravitational force from Mars. A thruster will be turned on for a short period of time to produce a thrust needed to initiate landing. As the capsule enters Mars atmosphere, it slows down due to the effect of air drag and the deployment of a parachute. You will simulate and analyze the trajectories of the capsule before and during the landing.

From Newton's second law, the motion of the capsule can be described by the following differential equations:

$$
\begin{aligned}
\frac{\partial U}{\partial t} &= \frac{Th^x}{m} - GM_e\frac{X}{(X^2+Y^2+Z^2)^{3/2}} - C_d\frac{\rho A}{2m}U\sqrt{U^2+V^2+W^2}, \\
\frac{\partial V}{\partial t} &= \frac{Th^y}{m} - GM_e\frac{Y}{(X^2+Y^2+Z^2)^{3/2}} - C_d\frac{\rho A}{2m}V\sqrt{U^2+V^2+W^2}, \\
\frac{\partial W}{\partial t} &= \frac{Th^z}{m} - GM_e\frac{Z}{(X^2+Y^2+Z^2)^{3/2}} - C_d\frac{\rho A}{2m}W\sqrt{U^2+V^2+W^2}, \\
\frac{\partial X}{\partial t} &= U, \\
\frac{\partial Y}{\partial t} &= V, \\
\frac{\partial Z}{\partial t} &= W, \qquad (1)
\end{aligned}
$$

where t is time (in seconds); X, Y, and Z are position (in meters) of the capsule relative to the center of Mars in rectilinear coordinate. U, V, and W are the velocity components (in meters per second) in the X, Y and Z directions, respectively. Th^x, Th^y and Th^z are three components of thrust (in N) produced by the function **thruster**. Mars is assumed to be stationary in this project.

The parameters in the equation 1 above are given as follows:

- Radius of Mars: $R = 3.3895 \times 10^6\,(m)$
- Mass of the Mars: $M = 6.39 \times 10^{23}\,(kg)$
- Gravitational constant : $G = 6.67408 \times 10^{-11}\,(m^3\,kg^{-1}\,s^{-2})$
- Mass of the capsule: $m = 800\,(kg)$
- Drag coefficient, C_d, of the capsule varies in time as produced by function **drag_parameters**
- Frontal area, A (m^2), of the capsule varies in time due to the deployment of the parachute as produced by function **drag_parameters**

- Air density,ρ (kg/m^{-3}), varies with altitude as produced by **air_density**

Furthermore, the following quantities will be of interest while analyzing the capsule trajectories:

- Altitude of the capsule (m): $h = \sqrt{X^2 + Y^2 + Z^2} - R$
- Speed of the capsule ($m\,s^{-1}$): $V_{mag} = \sqrt{U^2 + V^2 + W^2}$
- Acceleration of the capsule ($m\,s^{-2}$): $Acc = d(V_{mag})/dt$

2 Approach

Six trajectories will be simulated. The difference among the trajectories are due to the initial position and the initial velocity of the satellites. The initial position and velocity components are stored in text file: **simulation_data.txt**. When the thruster is turned on, the instantaneous values of the three components of thrust (Th^x, Th^y, and Th^z) are produced by function **thruster**. The capsule slows down during decent due to the frictional drag. The frictional drag is computed using the drag coefficient, the capsule's frontal area, and air density. Download **simulation_data.txt**, **thruster.m**, **air_density.m** and **drag_parameters.m** from [redacted].

Using Euler-Cromer method, equations 1 can be transformed into the following algebraic form:

$$\begin{aligned}
U_{n+1} &= U_n + \left[\frac{Th_n^x}{m} - GM\frac{X_n}{(X_n^2+Y_n^2+Z_n^2)^{3/2}} - C_{d,n}\frac{\rho_n A_n}{2m}U_n\sqrt{U_n^2+V_n^2+W_n^2}\right]\Delta t, \\
V_{n+1} &= V_n + \left[\frac{Th_n^y}{m} - GM\frac{Y_n}{(X_n^2+Y_n^2+Z_n^2)^{3/2}} - C_{d,n}\frac{\rho_n A_n}{2m}V_n\sqrt{U_n^2+V_n^2+W_n^2}\right]\Delta t, \\
W_{n+1} &= W_n + \left[\frac{Th_n^z}{m} - GM\frac{Z_n}{(X_n^2+Y_n^2+Z_n^2)^{3/2}} - C_{d,n}\frac{\rho_n A_n}{2m}W_n\sqrt{U_n^2+V_n^2+W_n^2}\right]\Delta t, \\
X_{n+1} &= X_n + U_{n+1}\Delta t, \\
Y_{n+1} &= Y_n + V_{n+1}\Delta t, \\
Z_{n+1} &= Z_n + W_{n+1}\Delta t, \qquad (2)
\end{aligned}$$

where subscript n denotes variables at current time, subscript n+1 denotes variables at time that is Δt ahead.

3 Tasks to perform

To simulate and analyze the trajectories, you are to write three MATLAB files: **trajectory.m**, **read_input.m** and **project.m**. The descriptions of these files are given below.

1, File trajectory.m:

This is the function that solves the equations 2 for a given set of initial conditions. Use $\Delta t = 0.2\,s$. The function should have the following header: **function [T, X, Y, Z, U, V, W] = trajectory(Xo, Yo, Zo, Uo, Vo, Wo)** where the inputs are components of the initial position (Xo, Yo, Zo) and initial velocity (Uo, Vo, Wo). The outputs are quantities discussed above. All inputs are scalars while all outputs are vectors. You need to simulate the trajectories until the capsule lands on the ground.

2, File read_input.m:

This function reads the parameters stored in the file **satellite_data.txt** into MATLAB. The function should have the following declaration: **function [Xo, Yo, Zo, Uo, Vo, Wo, tstart, tend, maxthrust] = read_input(inputfile, traj_id)** where **inputfile** is a string denoting the name of the file to be read and **traj_id** is an integer indicating the trajectory ID number. The outputs are the initial position (**Xo**, **Yo**, **Zo**) and components of initial velocity (**Uo**, **Vo**, **Wo**). When the input **traj_id** is not available in the file, the function should set all outputs to **NaN** and display an error warning to screen.

3, File project.m:

This is the main script to perform the tasks (and subtasks) described below.

Task 1: Here, you simulate landing trajectories. There are six trajectories with different initial values of position and velocity. Call function **trajectory** to simulate the landing. For each trajectory, the outputs should include time (T), three components of position (X, Y, Z) and three components of velocities (U, V, W), and all of these variables should be vectors with the same length.

Task 2: Here, you will create 3 figures based on the results from Task 1. All figures should have title, axis labels with correct units and legends. Use different colors to indicate the different trajectories. Please make sure that all 3 figures are plotted when your **project.m** is executed.

- Create figure 1 to plot the six trajectories in six different panels. Also include the landing positions of the capsule and Mars' surface in each panel. Use different colors for different trajectories. Use function **plot3** and **subplot**. Sample script to include the Mars' surface is given on [redacted], see the file **plot_Mars.m**.

- Create figure 2 which includes 3 panels. In the top panel, plot altitude versus time for all trajectories. Plot speed versus time in the middle panel and acceleration versus time in the bottom panel. Use function **subplot** to create different panels of the figure.

- Create figure 3 to plot acceleration versus altitude for all satellites. Plot the acceleration on the horizontal axis and altitude on the vertical axis. Put all plots together in a single panel. Set the vertical axis in log scale (**set(gca,'Yscale','log')**).

Task 3: Create a 6-element data structure named **stat** with the following fields:

- **trajectory_id**: to include a number ranging from 1 to 6 to indicate the trajectory ID.

- **final_time**: to include the landing time.

- **final_position**: to include a 3-element vector indicating the landing location.

- **final_velocity**: to include a 3-element vector indicating the velocity component (U, V, W) at landing.

- **final_speed**: to include the speed at landing.

- **time_hmax_altitude**: to include the times at which the capsule is farthest from Earth (e.g. times at which the altitude h shows a local maximum).
- **orbital_period**: to include the time that the capsule take to revolve around Mars for one time *before* the descent. *Hint: Subtract the times of the first two local maxima of altitude h.*

Task 4: Create a file named **report.txt** to report the following:

- Your name on the first line
- Your PID on the second line
- A string **traj_id, landing_time (s), landing_speed (m/s), orbital_period (s)** on the third line
- Corresponding values of trajectory ID, landing time, landing speed and orbital period before the descent from line fourth to line nineth. Use single digit for the trajectory ID and format %15.9e for others.

Task 5: At the end of your **project.m** script, set the following:

```
p1a =evalc('help read_input ';
p1b =evalc('help trajectory ';
p1c ='See figure 1';
p1d ='See figure 2';
p1e ='See figure 3';
p2a = stat(1);
p2b = stat(2);
p2c = stat(3);
p2d = stat(4);
p2e = stat(5);
p2f = stat(6);
p3 = evalc('type report.txt');
```

and answer the following questions using figure 3:

Q1. At what altitude does the capsule does the capsule feel the effect of air drag? Put the answer in string **p4a = '...'**.

Q2. At what altitude is the parachute deployed? Put the answer in string **p4b = '...'**.

[redacted]