RMIT University

School of Engineering

EEET2248 – Electrical Engineering Analysis

Lab Experiment #4

Calculus for Rocket Trajectory Analysis

Lecturer: Dr. Katrina Neville

Student Name: Nathan Paul Williams

Student Number: s3707244

Group: Tuesday 10:30-12:30

Submission Due Date: 25th May 2018

Problem Statement

Given a data set containing rocket altitude relative to time at intervals:

- Extrapolate velocity and acceleration
- Determine maximum values and corresponding time
- Determine time spent above target g-force
- Determine times where speed exceeded mach 1,2,3,4,etc..

Input

- Spreadsheet containing time vs altitude
- Speed of sound
- Value for G (acceleration due to gravity)

Output

- Max speed
- Max acceleration
- Times at which mach speeds are initially reached
- Time period spent above 9 g's
- Altitude, velocity and acceleration subplots
- Mach speed step plot

Design

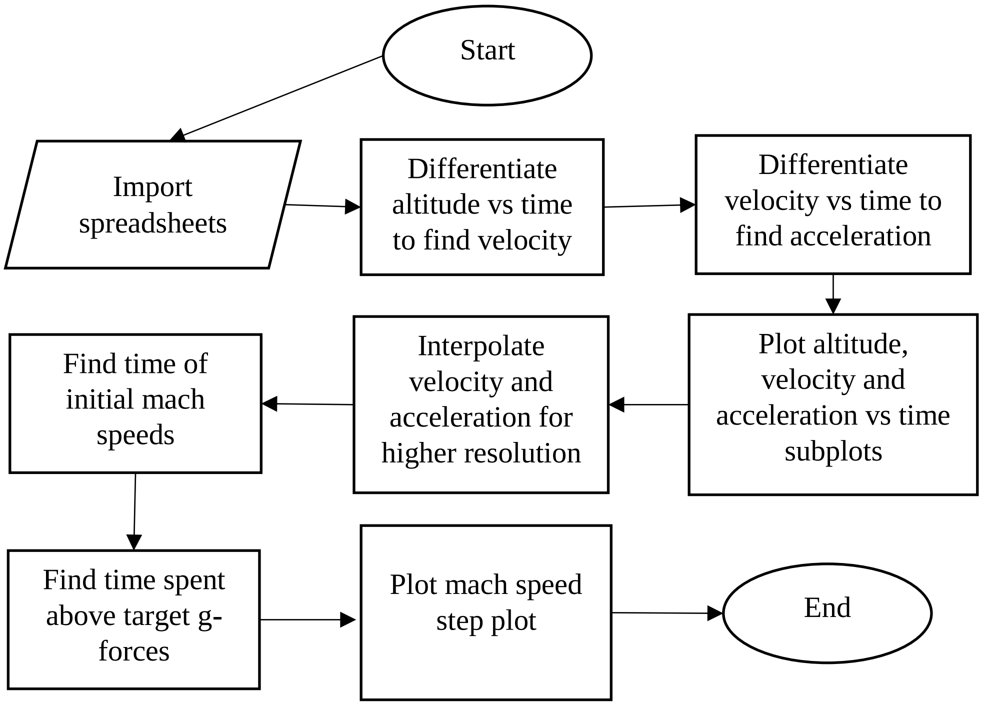
Start
Import spreadsheets
Differentiate altitude vs time to find velocity
Differentiate velocity vs time to find acceleration
Plot altitude, velocity and acceleration vs time subplots
Interpolate velocity and acceleration for higher resolution
Find time of initial mach speeds
Find time spent above target g-forces
Plot mach speed step plot
End

Figure 1: Design flowchart

The design of my program was focused on providing accurate data output that can be interpreted and contrasted by the user with minimal effort. My program begins by importing the data spreadsheet and assigning variables to altitude and time. The altitude is then differentiated with respect to time to find velocity using the diff() function. This is done by dividing the diff(altitude) by diff(time). This is repeated with velocity to find acceleration. Each iteration the time range must be adjusted as each differentiation returns an array of size n-1 when n is the original size as we are calculating for points in between the original data points. E.g. velocity is calculated using the difference in between each altitude recorded. These 3 data sets are then plotted in 3 subplots to allow the user to contrast and compare them. Max values are found using the max() function and their corresponding times are found using the find() function. Velocity and acceleration are then interpolated using interp1() to increase the resolution of the data to provide a more accurate result in the following analysis.

To find the times at which initial mach speeds are reached a few tools are implemented. Firstly, an anonymous function is defined to calculate mach speed, taking velocity as an input. An array of zeros is created that is of length equal to the mach speed at max velocity. A counter variable for mach speed is also intitialised. A for loop then iterates through velocity, checking if the return of the mach function, given that velocity, is greater than the current mach speed counter. If so, it will record the corresponding time in the array at the index of the mach speed counter and then increment the mach speed counter. Another for loop then prints out all values in the array using fprintf in a neat string displaying the mach number and the corresponding time.

To calculate time spent above the target g-force, a for loop iterates through acceleration and records times in an array if the acceleration exceeds the required value. Fprintf() then prints the first and last time values in a neat string. Finally a step plot of mach speeds vs time is created using the floor() of the value returned by the mach() function for each velocity value. This process is summarised in the flowchart in fig 1.

Discussion

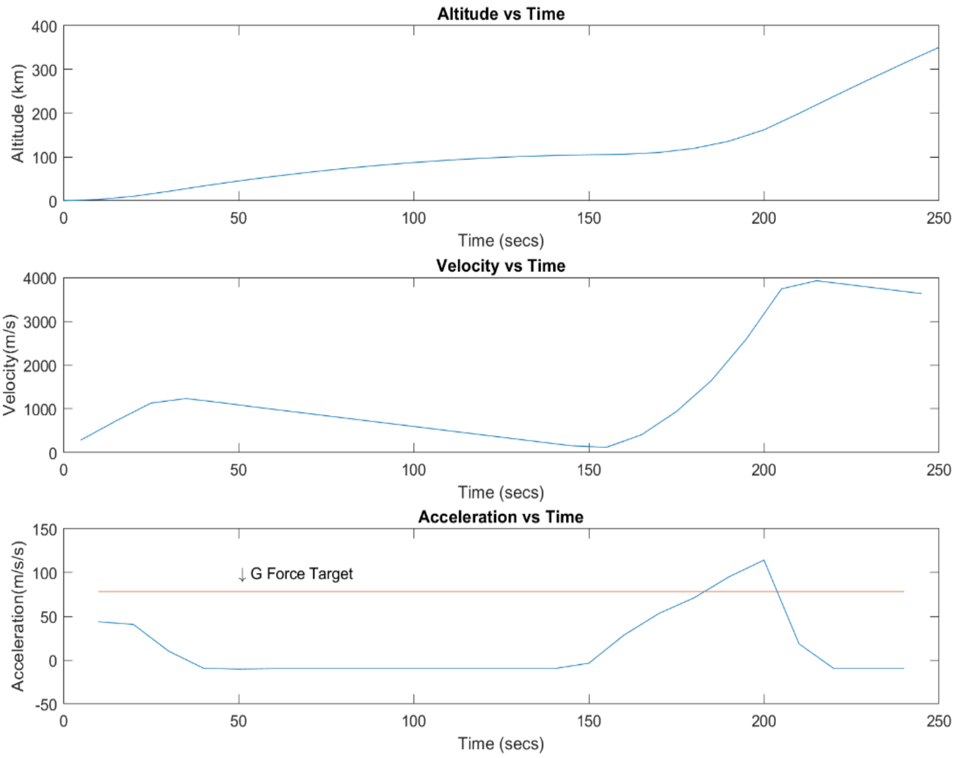

Figure 2: Subplots for alt, acc, vel vs time

The output subplots of altitude, velocity and acceleration vs time are shown in fig 2. Each plot appears to correspond to the gradient of the graph above. On the basic principles of calculus this is evidence that the output of the data is correct. The acceleration graph also has a horizontal line to visually demonstrate the times where the force experienced is greater than 9 g's. Notice the x limits are reduced by 10 seconds (5 seconds each side) in each plot due to the change in range upon differentiation as explained in the design.

```
Max speed 3926.9 m/s is reached at 220 seconds

Max acceleration 114.22 m/s/s is reached at 200 seconds


Mach 1 is reached at 6.1 seconds

Mach 2 is reached at 13.6 seconds

Mach 3 is reached at 21.7 seconds

Mach 4 is reached at 180.5 seconds

Mach 5 is reached at 185.1 seconds

Mach 6 is reached at 188.6 seconds

Mach 7 is reached at 192.1 seconds

Mach 8 is reached at 195.5 seconds

Mach 9 is reached at 198.4 seconds

Mach 10 is reached at 201.3 seconds

Mach 11 is reached at 204.2 seconds


G-force is greater than 9 gs between 183.1 and 203.7 seconds
```

Figure 3: CLI output

The CLI output is shown in fig 3. The max speed and acceleration values output appear to correctly correspond to the plots in fig 2 providing more evidence our program is accurate. The mach number is the multiple of the speed of sound [1] (e.g. mach 6 is 6 times the speed of sound). Determining a speed of sound value to use was quite complex as it varies quite dramatically with temperature which changes with altitude according to a mathematical model [2]. For the purpose of this task I decided to use the speed of sound at 0 degrees celcius which is 331.2 m/s [3]. However; if this program was to be implemented for practical purposes the model referenced would have to implemented in the program. This is to allow for the changing temperature and hence speed of sound with altitude.

For the g-force experienced by the rocket to be 9 g's the acceleration must be enough to cause 8g's of force, as there is 1 g of force already on the rocket due to gravity (again this changes slightly with altitude however it was simplified to a constant 1g for this task). G-force is the force experienced under acceleration due to gravity which is approximately equal to 9.8 m/s/s [4]. Therefore the acceleration of the rocket must be greater than 78.4 m/s/s to experience over 9 g's of force. Our output shows this is between 183.1 and 203.7 seconds which appears to correspond to the visual guide on the plot in fig 2.

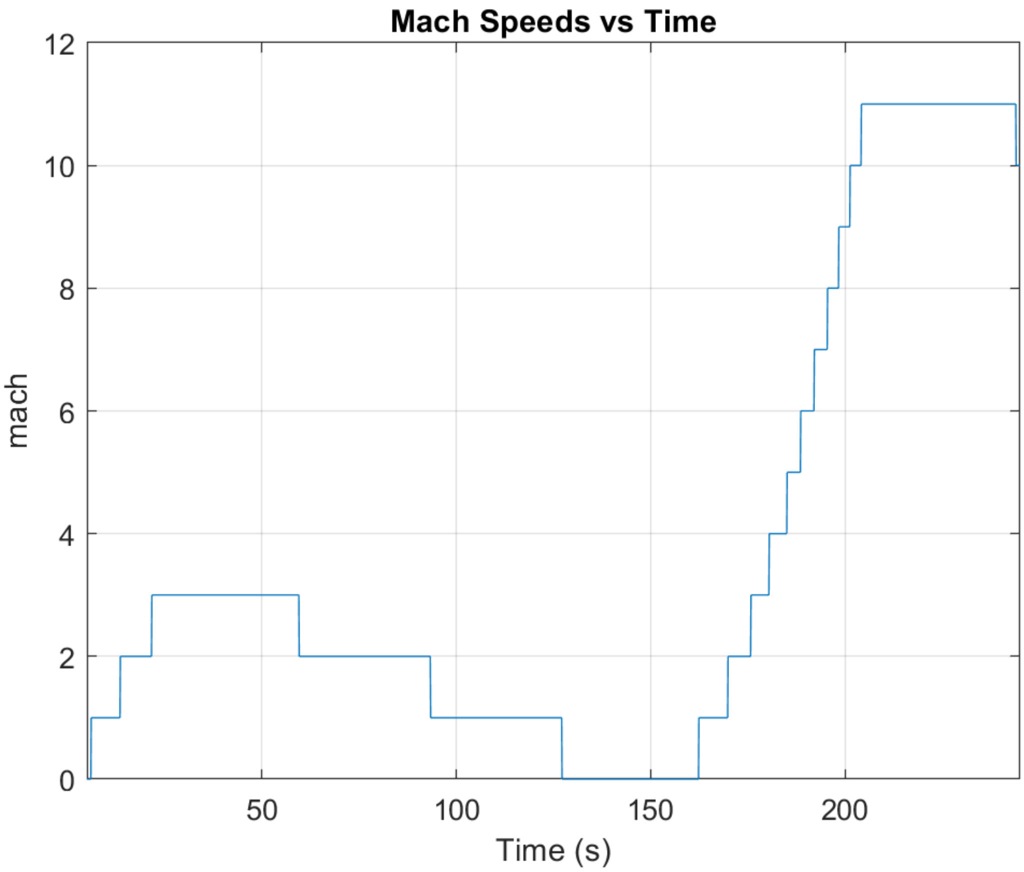

Figure 4: Step plot mach speeds vs time

Fig 4 shows the output step plot of mach number vs time. The plot appears to correspond to the intial mach speed times output in the CLI interface providing another piece of evidence our program is accurate. Note this plot displays the mach number exceeded i.e. the floor of (velocity/speed of sound) rather than a decimal ratio.

Conclusion

A solution has been presented that addresses the problem statement and from the limited testing appears to output accurate data and correctly perform all intended operations. Values were output for max speed, max acceleration, initial mach times and time period spent above 9 g's along with the relevant plots. For future improvement the program could implement the model for atmospheric temperature referenced to provide a more accurate value for the speed of sound and/or implement a changing value for g with altitude to more accurately calculate the g-forces experienced by the rocket. Overall the program provides output that can be easily interpreted and analysed by the user however further testing could be required with more input data to verify the accuracy.

References

[1] "NASA". (2018, 16/05/2018). *Mach Number*. Available: https://www.grc.nasa.gov/www/k-12/airplane/mach.html

[2] "NASA". (2015, 16/05/2018). *Earth Atmosphere Model*. Available: https://www.grc.nasa.gov/www/k-12/airplane/atmosmet.html

[3] H. Physics". (2018, 16/05/2018). *Speed of Sound in Various Bulk Media*. Available: http://hyperphysics.phy-astr.gsu.edu/hbase/Tables/Soundv.html

[4] NIST. (2018, 16/05/2018). *Fundamental Physics Constants*. Available: https://physics.nist.gov/cgi-bin/cuu/Value?gn

Appendix

```
%% Lab 4
%initialise
clc;
clear;

%% variables
step = 10;
soundS = 331.2;
gForce = 9.8;
gTarget = 9*gForce - gForce;
%% functions
mach = @(speed) (speed/soundS);
%% Import
%import data sheet
data = xlsread("data.xlsx");
time = data(:,1);
alt = data(:,2);
altP = alt/1000;

figure('position', [250, 70, 1000, 700])
subplot(3,1,1)
plot(time,altP)
title('Altitude vs Time')
xlabel('Time (secs)')
ylabel('Altitude (km)')

y = diff(alt)/diff(time);
vel = y(:,1);
t2 = 5 + time(1):step:time(end);
t2 = reshape(t2,[25,1]);

subplot(3,1,2)
plot(t2, vel)
title('Velocity vs Time')
xlabel('Time (secs)')
ylabel('Velocity(m/s)')


y2 = diff(vel)/diff(t2);
acc = y2(:,1);
t3 = 5 + t2(1):step:t2(end);
t3 = reshape(t3,[24,1]);

subplot(3,1,3)
plot(t3,acc)
hold on
plot(t3,gTarget*ones(size(t3)))
text(50,100,'\downarrow G Force Target')
title('Acceleration vs Time')
xlabel('Time (secs)')
ylabel('Acceleration(m/s/s)')

maxS = max(vel);
maxST = find(vel == maxS)*10;
```

```
maxA = max(acc);
maxAT = find(acc == maxA)*10;

%interp arrays
velX = 5:10:245;
velXI = 5:0.1:245;
t2 = 5:0.1:245;
vel = interp1(velX, vel, velXI, 'linear');

accX = 10:10:240;
accXI = 10:0.1:240;
t3 = 10:0.1:240;
acc = interp1(accX, acc, accXI, 'linear');



machArray = zeros(1,floor(mach(maxS)));
machC = 1;
for i = 1:1:length(vel)
    if (mach(vel(i)) >= machC)
        machArray(machC) =  t2(i);
        machC = machC + 1;
    end
end

fprintf('Max speed %g m/s is reached at %g seconds\n',maxS,maxST);
fprintf('Max acceleration %g m/s/s is reached at %g seconds\n\n',maxA,maxAT);

for i = 1:1:floor(mach(maxS))
    fprintf('Mach %g is reached at %g seconds\n',i,machArray(i));
end


gArray = [];

for i = 1:1:length(acc)
    if acc(i) > gTarget
        gArray = [gArray,t3(i)];
    end
end

startG = gArray(1);
endG = gArray(end);

fprintf("\nG-force is greater than 9 gs between %g and %g seconds\n",startG,endG);

figure(4)
plot(t2,floor(mach(vel)))
grid on
xlim([5 245])
title('Mach Speeds vs Time')
xlabel('Time (s)')
ylabel('mach')
```