FLIGHT MECHANICS: HOMEWORK -III REPORT

Emircan Kılıçkaya

Istanbul Technical University

110150045

Author Note

This paper is dedicated to UCK322E CRN:21218

Correspondence concerning this article should be addressed to Emircan Kılıçkaya,

Faculty of Aeronautics and Astronautics, Istanbul Technical University, Maslak 34469.

E-mail: kilickaya15@itu.edu.tr

School No:110150045

Date: 24.05.2019

FLIGHT MECHANICS HW3 REPORT

Problem 1:

Euler method and Runge-Kutta methods are specific algorithms in order to solve differential equations. They both come handy in different situations. Euler is the simplest and oldest one, so let's start with that.

$$\frac{dy}{dx} = f(x, y)$$

When one wants to estimate the next y value, something like below is used:

New value = old value + slope × step size

Or in mathematical terms:

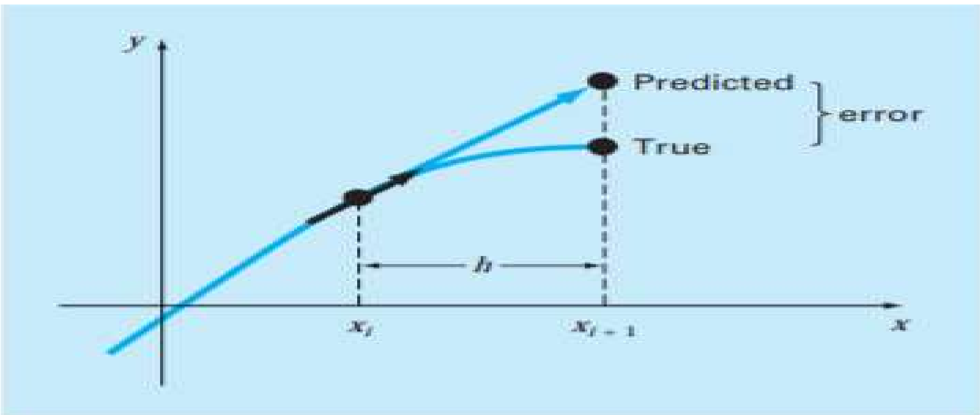
$$y_{i+1} = y_i + \phi h$$

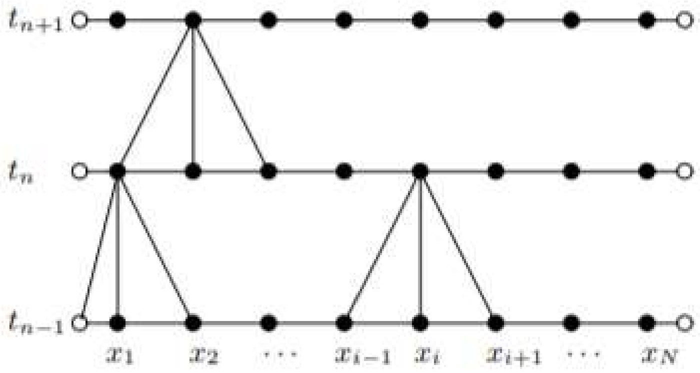
The first derivative provides a direct estimate of the slope at x:

$$\phi = f(x_i, y_i)$$

where $f(x_i\ ,y_i)$ is the differential equation evaluated at x_i and y_i. This estimate can be substituted into Eq. (2):

$$y_{i+1} = y_i + f(x_i, y_i)h$$





This formula is referred to as Euler's (or the Euler-Cauchy or the point-slope) method. A new value of y is predicted using the slope (equal to the first derivative at the original value of x) to extrapolate linearly over the step size h.

Another class of explicit one-step methods are the Runge-Kutta methods, which are frequently used in practice particularly for aerodynamical flow simulations. Runge-Kutta (RK) methods achieve the accuracy of a Taylor series approach without requiring the calculation of higher derivatives. Many variations exist but all can be cast in the generalized form of this:

$$y_{i+1} = y_i + \phi(x_i, y_i, h)h$$

where $\phi(x_i, y_i, h)$ is called an increment function, which can be interpreted as a representative slope over the interval. The increment function can be written in general form as:

$$\phi = a_1k_1 + a_2k_2 + \cdots + a_nk_n$$

where the a's are constants and the k's are:

$$k_1 = f(x_i, y_i)$$

$$k_2 = f(x_i + p_1h, y_i + q_{11}k_1h)$$

.

.

.

where the p's and q's are constants. Notice that the k's are recurrence relationships. That is, k_1 appears in the equation for k_2, which appears in the equation for k_3, and so forth. Because each k is a functional evaluation, this recurrence makes RK methods efficient for computer calculations.

The most popular RK methods are fourth order. The following is the most commonly used form, and we therefore call it the classical fourth-order RK method:

$$y_{i+1} = y_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)h$$

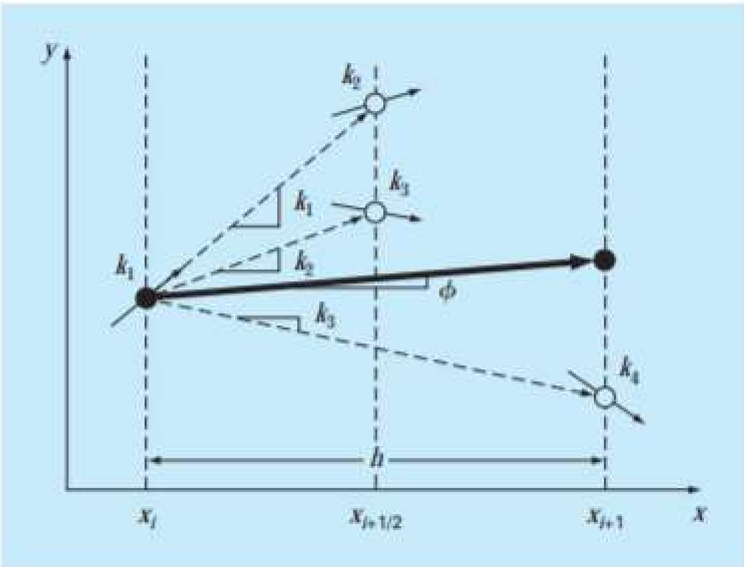
where

$$k_1 = f(x_i, y_i)$$

$$k_2 = f(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_1 h)$$

$$k_3 = f(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_2 h)$$

$$k_4 = f(x_i + h, y_i + k_3 h)$$



In conclusion, Runge-Kutta 4th order method gives more accurate results than Euler’s method and it should be selected when accuracy is at stake. Local truncation error for Euler's method is h^2 while for RK 4th order it is h^5. However, Euler’s method is simpler and it is easier to implement in code. Also it takes less time to compile, which makes it much more efficient. So, at situations where the error margin is enough Euler’s method could be used.

Problem 2:

In order to produce the graphics below, one needs to run the MATLAB program called “HOMEWORK3_MAIN.m” inside the zip-file. All other functions inside the zip-file are necessary implementations of older homeworks and they were used in this homework too. That’s the reason of their existence. Just running the main program is sufficient. The program to run is:

“HOMEWORK3_MAIN.m”

During the calculations of a complete 360° turn, banking angle is chosen as 30°. BADA manual suggests 35° but 30° is chosen just to be on the safe side and also for the comfort of any possible passengers on the plane.

For the initial condition of V_{tas}, piece of code in program is:

*climb_speed_schedule(h(1))*0.514444444;*

Since the plane is first climbing, climb speed schedule was used to determine the first velocity. The constant 0.5144 is necessary for converting knots to m/s. $V_{cl,1}$ and $V_{cl,2}$ values were assumed as 360 Knots due to non-existance of the GPF file.

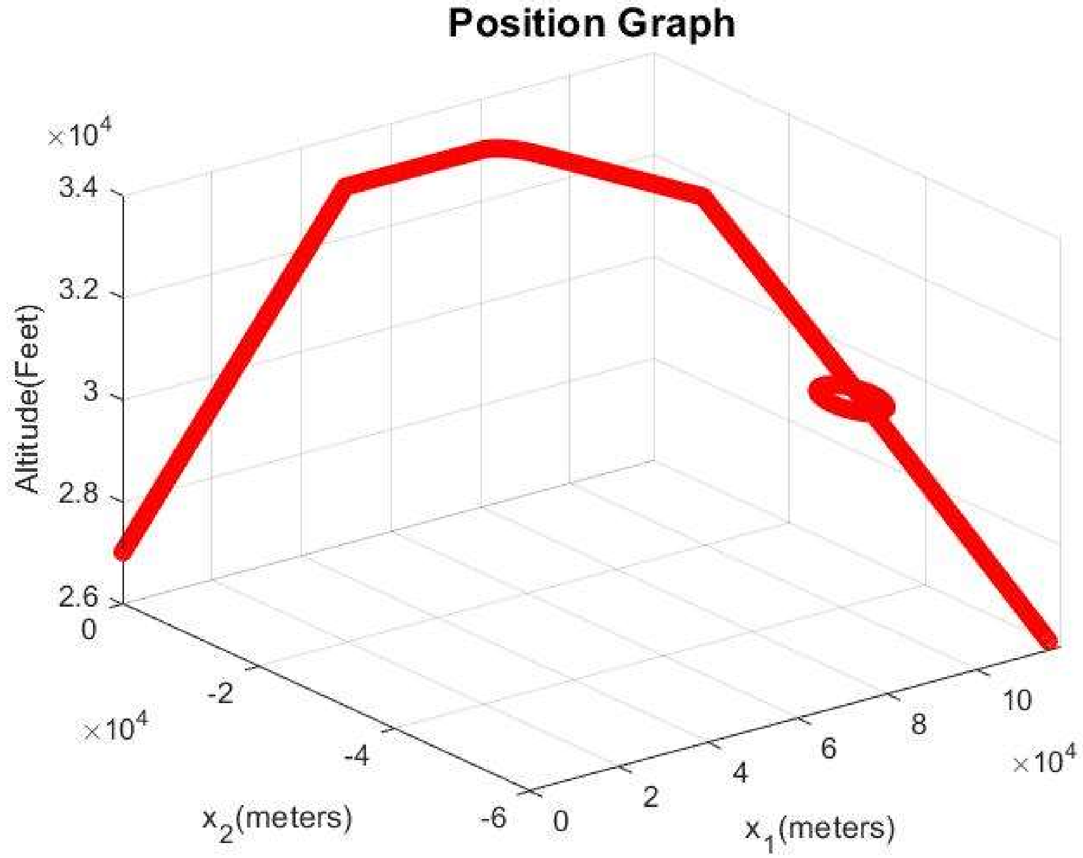
Another important note is that the time is converted to minutes at the end of the program in order to produce better graphs.The code responsible for this:

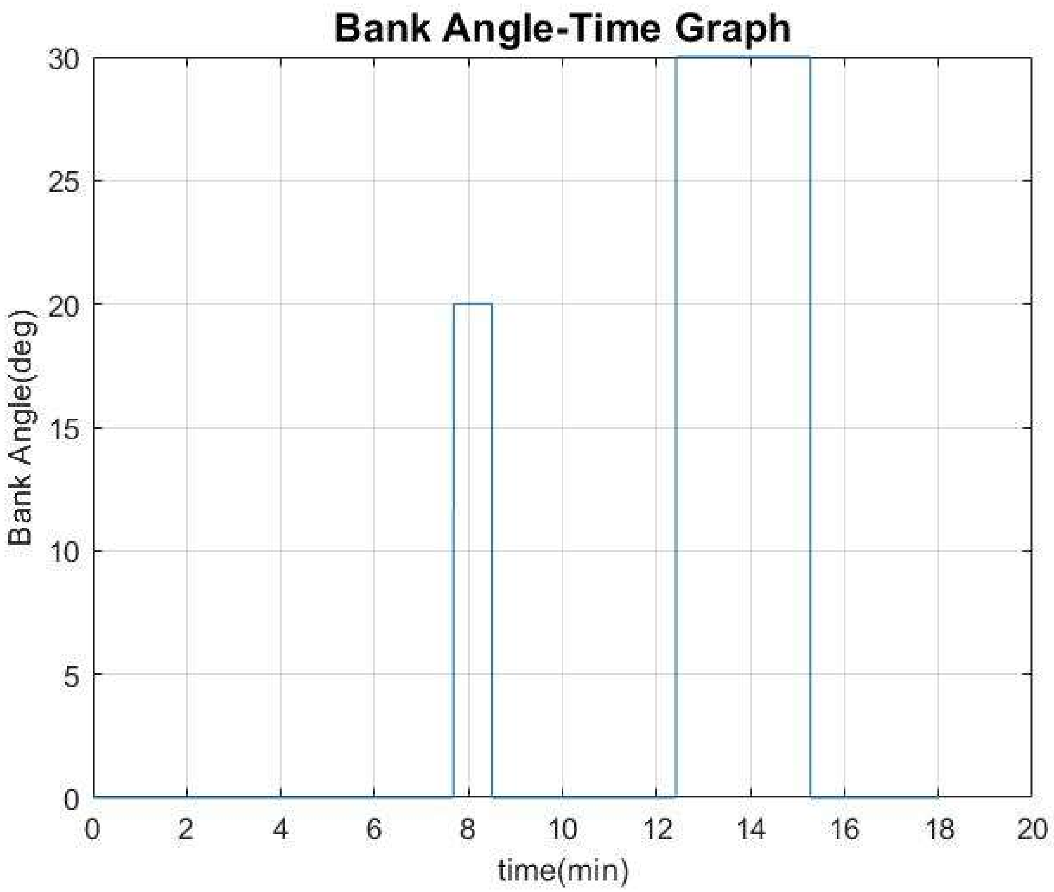
t=t./60;

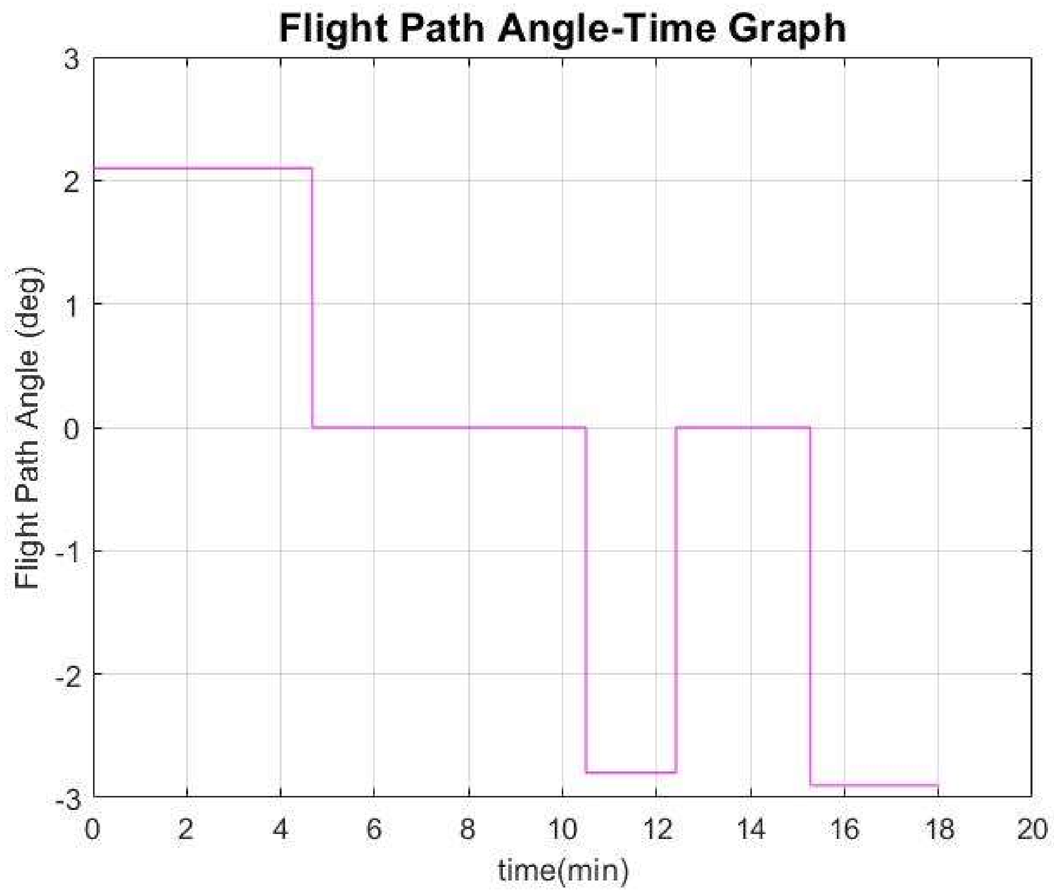
There are some commented lines of code in the program. They were used for test purposes and left deliberately.

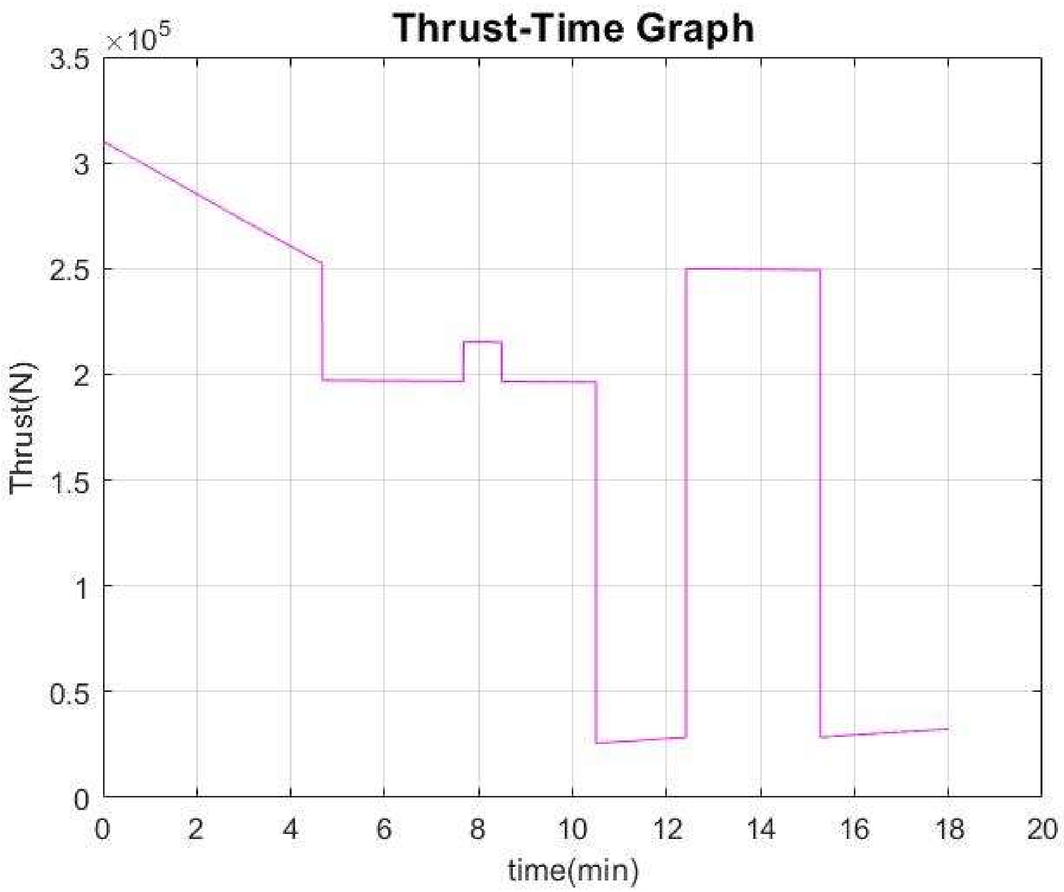
Also, during this process some bugs were found in the functions of older homeworks and they were corrected immediately. These are the updated versions.

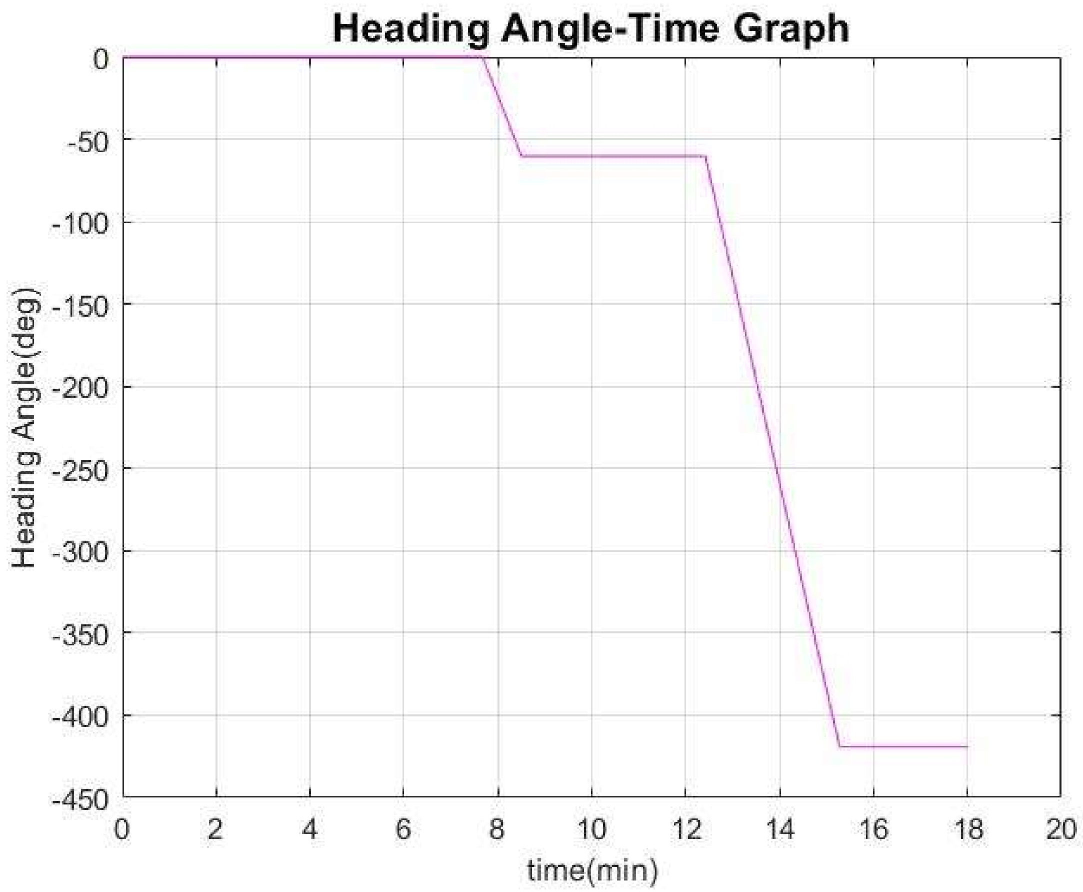
Graphs are below and can be inspected cautiously.

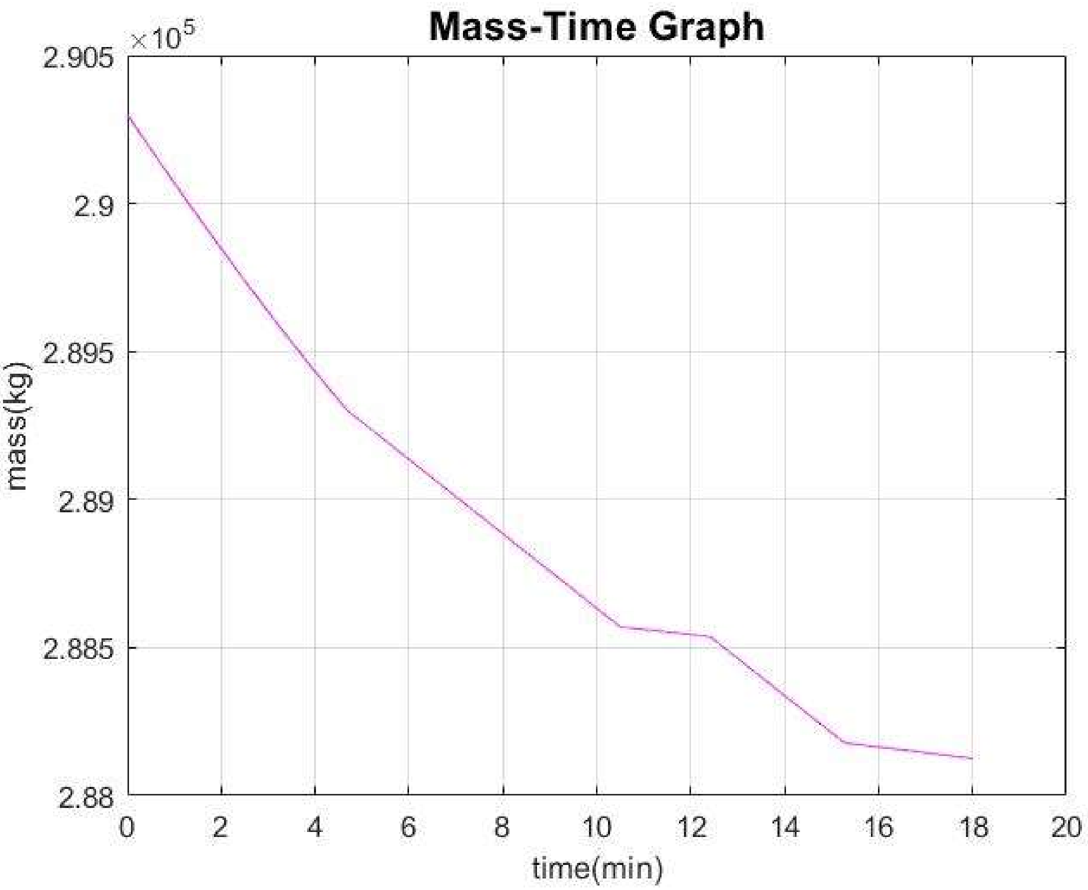

Position Graph
×10^4
3.4
3.2
3
2.8
2.6
Altitude(Feet)
0
-2
×10^4
-4
-6
x_2(meters)
0
2
4
6
8
10
×10^4
x_1(meters)

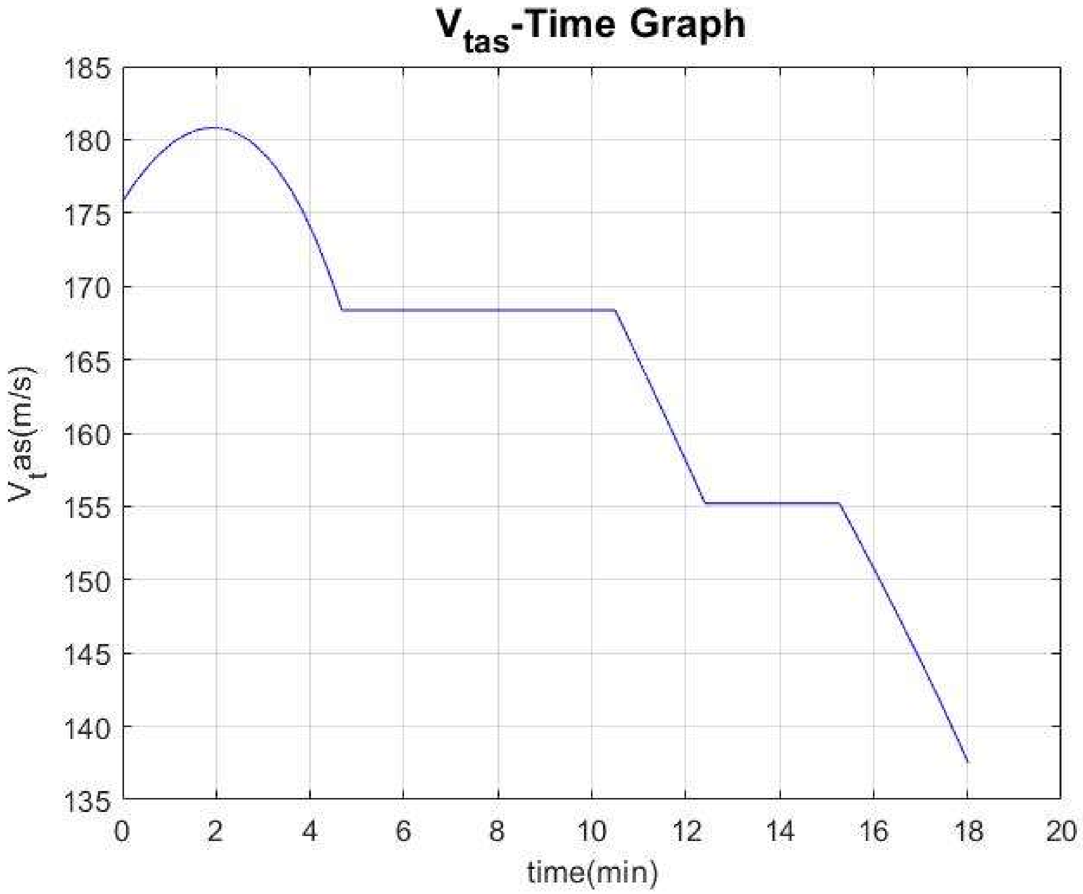

Bank Angle-Time Graph
30
25
20
15
10
5
0
Bank Angle(deg)
0
2
4
6
8
10
12
14
16
18
20
time(min)


Flight Path Angle-Time Graph
Flight Path Angle (deg)
time(min)
3
2
1
0
-1
-2
-3
0
2
4
6
8
10
12
14
16
18
20


Thrust-Time Graph
×10^5
Thrust(N)
time(min)
3.5
3
2.5
2
1.5
1
0.5
0
0
2
4
6
8
10
12
14
16
18
20


Heading Angle-Time Graph
Heading Angle(deg)
time(min)
0
-50
-100
-150
-200
-250
-300
-350
-400
-450
0
2
4
6
8
10
12
14
16
18
20


Mass-Time Graph
$\times 10^5$
mass(kg)
time(min)
2.905
2.9
2.895
2.89
2.885
2.88
0
2
4
6
8
10
12
14
16
18
20



Vtas-Time Graph
185
180
175
170
165
160
155
150
145
140
135
Vtas(m/s)
0
2
4
6
8
10
12
14
16
18
20
time(min)

REFERENCES

Chapra, S., & Canale, R. (2010). *Numerical Methods for Engineers* (6th edition). New York, NY: McGraw-Hill.

Schäfer, M. (2006). *Computational Engineering – Introduction to Numerical Methods.* Berlin, Germany: Springer.

EEC Technical/Scientific Report No.13/04/16-01(BADA3.11)

Ghosh, A.(2015). NPTEL Flight Mechanics Lecture Notes.