# Flight Mechanics I: Labs

Bioengineering and Aerospace Department

February 5, 2019

## Contents

**1 Introduction** **2**
1.1 Methodology of the labs . . . . . 2
1.2 Students keeping their old marks . . . . . 2

**2 Lab 1: Equations of motion** **3**
2.1 Aim . . . . . 3
2.2 Equation of motion . . . . . 3
2.3 Tasks . . . . . 3
2.3.1 Preliminary analytical calculations . . . . . 3
2.3.2 Matlab Model . . . . . 4
2.3.3 Simulink Model . . . . . 5
2.4 Content of the report . . . . . 5

**3 Lab 2: Aircraft Performance** **6**
3.1 Aim . . . . . 6
3.2 Cruise . . . . . 7
3.2.1 Tasks . . . . . 7
3.3 Motion in a vertical plane with constant velocity elevation angle . . . . . 7
3.3.1 Tasks . . . . . 8
3.4 Turning manoeuvre . . . . . 8
3.4.1 Tasks . . . . . 8
3.5 Content of the report . . . . . 9

# 1 Introduction

## 1.1 Methodology of the labs

The continuous evaluation of the course involves four labs and the following related activities:

1. Attending the lab sessions.
2. Delivering the lab reports.
3. Oral Exam about the labs.

This documents includes the description of each of the labs and the assignment. At the beginning of the course, the coordinator will publish the lists with the compositions of three lab groups (named 45, 46, and 456) and the schedule of each of them. Students will make teams of two people that, after attending the lab sessions, will prepare reports with the results of each lab. The rules are:

1. The reports will be uploaded to Aula Global before one week after the lab sessions.
2. Each report should include a cover page (title of the lab and names of the students) plus a maximum of 5 pages with the results of the lab. It will be a two-column document with font size equal to 11pt.
3. The reports should focus on the results and the questions of the assignments. Students can include short theoretical explanations (if useful to justify the results). They should avoid long Introductions repeating the theoretical information already delivered in sessions.
4. For each lab, each group will submit a .zip file with the name *Group_X_Lab*$_Y$, where $X$ is the group number and $Y$ is the lab session number. Inside the .zip they will include the pdf of the report and, if generated during the lab session, Matlab and XFLR5 files.

The reports will be corrected by the professors and the marks of the labs will be published according to UC3M rules.

| | Title | Professor | Schedule |
|---|---|---|---|
| Lab 1 | Equations of motion | J. Zhou | 456: 11/3/2019, 13:00-15:00, 7.0.J02<br>45: 13/3/2019, 11:00-13:00, 7.0.J02<br>46: 15/3/2019, 13:00-15:00, 7.0.J02 |
| Lab 2 | Aircraft Performance | J. Zhou | 456: 8/4/2019, 13:00-15:00, 7.0.J02<br>45: 10/4/2019, 13:00-15:00, 7.0.J02<br>46: 12/4/2019, 13:00-15:00, 7.0.J02 |
| | Oral Exam<br>Oral Exam | All<br>All | 45: TBD<br>46: TBD |

Table 1: Schedule of the labs and the oral exam.

## 1.2 Students keeping their old marks

1. Beatriz Cabrero Saiz
2. Rafael Gasso Climent
3. Alejandro Martinez Cervero
4. Mikel Cortazar del Prado
5. Feliz Bautista Paz
6. Oscar del Olmo Pea
7. Alejandro Gomez Martnez
8. Alejandro Martnez Cervero

# 2 Lab 1: Equations of motion

## 2.1 Aim

The students will develop skills for the numerical simulation of dynamic systems and will practice with three concepts in flight mechanics: equilibrium state, stability (static and dynamic), and numerical integration. They will also see the different effects of changing the physical parameters and the initial conditions. Trajectories will be visualized in variable-versus-time diagrams and in phase space diagrams. The analysis of the lab combines analytical and numerical calculations.

## 2.2 Equation of motion

A disk of mass $m_d$ and radius $R$ is placed horizontally on the top of a linear torsional spring of constant $k(Nm/rad)$. It can rotate without friction about a vertical axis that passes through its center of mass $O_E$. Let $\psi(t)$ be the angle between the $O_E X_E$ axis of an Earth frame and the $O_E X_B$ axis of a *body* frame linked to the disk, (see Figure 1). For $\psi = 0$ the spring is relaxed. At point $P$, $\overline{O_E P} = x_p \boldsymbol{i_B}$, we attach a symmetric airfoil of chord $c$ ($c << R$) linked to the disk. Its axis of symmetry coincides with $O_E X_B$. The system is immersed in a flow of density $\rho$ and velocity $\boldsymbol{U}_\infty = -U_\infty \boldsymbol{i}_E$. The airfoil has a lift coefficient slope equal to $a_w$, surface $S$ and both its mass and its drag are negligible as compared with the mass of the disk and its lift, respectively. The interaction of the fluid with the airfoil produces a lift force and zero torque about point $P$. Hereafter, we consider the dynamics of the system made of the disk and the airfoil, which behave as a rigid body.

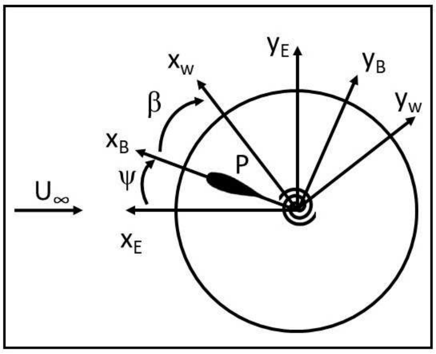


Figure 1: Frame of references and mechanical system

| Parameter | Value |
|---|---|
| $m_d$ | $1.0kg$ |
| $R$ | $0.5m$ |
| $k$ | $0.1Nm$ |
| $U$ | $12m/s$ |
| $\rho$ | $1.0kg/m^3$ |
| $a_w$ | 4 |
| $S$ | $0.005m^2$ |
| $x_p$ | *see below* |

Table 2: Parameter Values

For $U_\infty >> x_p\dot{\psi}$, the total torque about $O_E$ is

$$\boldsymbol{M}_{O_E} = \boldsymbol{M}_{O_E}^{airfoil} + \boldsymbol{M}_{O_E}^{spring} = \left(\frac{1}{2}\rho S U_\infty^2 a_w x_p \cos\psi - k\right)\psi \boldsymbol{k}_B \equiv M_{O_E}\boldsymbol{k}_E \tag{1}$$

The component along $\boldsymbol{k}_E$ of the equation describing the evolution of the angular momentum, $d\boldsymbol{H}_{O_E}/dt = \boldsymbol{M}_{O_E}$, reads[1]

$$\ddot{\psi} = \Omega^2 \left[\delta \cos\psi - 1\right]\psi, \tag{2}$$

where we took into account that $\boldsymbol{H}_{O_E} = \bar{\boldsymbol{I}}_{O_E} \cdot \boldsymbol{\omega} = \frac{1}{2} m_d R^2 \dot{\psi} \boldsymbol{k}_E$, and introduced the parameters

$$\Omega^2 \equiv \frac{2k}{m_d R^2}, \quad \delta \equiv \frac{\rho S U_\infty^2 x_p a_w}{2k} \tag{3}$$

## 2.3 Tasks

### 2.3.1 Preliminary analytical calculations

1. Write Eq. 2 as a first order dynamical system

$$\frac{d\boldsymbol{u}}{dt} = \boldsymbol{f}\left(\boldsymbol{u}; \boldsymbol{p}\right) \tag{4}$$

   with the state vector $\boldsymbol{u} = \begin{bmatrix}\psi & \dot{\psi}\end{bmatrix}$ and the parameter vector $\boldsymbol{p} = [\Omega \quad \delta]$,

2. Find the three equilibrium states that satisfy $\boldsymbol{f}\left(\boldsymbol{u}; \boldsymbol{p}\right) = 0$. If necessary, determine the parameter range where each equilibrium state exists.

[1]Students should demonstrate this results at home.

3. Compute the stability derivative $\partial M_{O_E}/\partial\psi$ evaluated at the equilibrium state $\boldsymbol{u}_0 = [0 \;\; 0]$.

4. Discuss the static stability of $\boldsymbol{u}_0$ according to the sign of $\partial M_{O_E}/\partial\psi\,|_{\boldsymbol{u}_0}$

5. Compute the analytical formula for the maximum value of the length $x_p$, named $x_p^*$, that makes the equilibrium state $\boldsymbol{u}_0$ statically stable.

6. Compute $x_p^*$ for the parameter values of the Table above.

7. Summarize all your analytical results in the table below

| Question | Answer |
|---|---|
| 1 | $\boldsymbol{f} =$ |
| 2 | $\boldsymbol{u}_0 = [0 \quad 0]$<br>$\boldsymbol{u}_+ =$ $\qquad$ $\boldsymbol{u}_+$ exists if $x_p > \cdots$<br>$\boldsymbol{u}_- =$ $\qquad$ $\boldsymbol{u}_-$ exists if $x_p > \cdots$ |
| 3 | $\partial M_{O_E}/\partial\psi\,|_{\boldsymbol{u}_0}=$ |
| 4 | $\partial M_{O_E}/\partial\psi\,|_{\boldsymbol{u}_0}< 0 \to \boldsymbol{u}_0$ is $\cdots$<br>$\partial M_{O_E}/\partial\psi\,|_{\boldsymbol{u}_0}> 0 \to \boldsymbol{u}_0$ is $\cdots$ |
| 5 | $x_p^* =$ |
| 6 | $x_p^* =$ |

### 2.3.2 Matlab Model

The student will create the following Matlab functions:

1. <u>*Parameters*</u>. Goal: compute the dimensionless parameters. Function Inputs: none. Function Output: components of vector $\boldsymbol{p}$.

2. <u>*ODE*</u>. Goal: compute the right hand side of Eq. 4. Function Inputs: time $t$ and state vector $\boldsymbol{u}$. Function Output: $d\boldsymbol{u}/dt$. It uses vector $\boldsymbol{p}$ as a global variables

Create a Matlab script, named *Main*, that performs the following tasks

1. Define $\boldsymbol{p}$ as a global variable

2. Initialize vector $\boldsymbol{p}$ by using the function *Parameters*.

3. Compute <u>all</u> the equilibrium states and plot them in a graph $\psi - \dot{\psi}$. Use these calculations as inputs in function *ODE* and check that the equilibria were calculated correctly.

4. Compute 15 trajectories with random initial conditions and plot them in three diagrams: $\psi - t$, $\dot{\psi} - t$, and $\psi - \dot{\psi}$. Students can use the Matlab integrator *ode*45.

Run the Main for $x_p = 0.1$ and $x_p = 0.06$. Save the figures and think about their physical meaning (relation between the motion of the disk and the position of the airfoil).

### 2.3.3 Simulink Model

The student will construct the Simulink model of Fig. 2. It has the following *User-Defined Functions* (Matlab Function)[2]:

1. *Sim_Parameters*. Goal: compute the dimensionless parameters. Block Inputs: none. Block Output: components of vector $\boldsymbol{p}$

2. *Sim_IC*. Goal: compute the initial conditions for the numerical integration. Block Inputs: none. Block Output: components of vector $\boldsymbol{u}$ at $t = 0$.

3. *Sim_ODE*. Goal: compute the right hand side of Eq. 4. Block Inputs: parameter vector $\boldsymbol{p}$ and state vector $\boldsymbol{u}$. Block Output: $d\boldsymbol{u}/dt$.

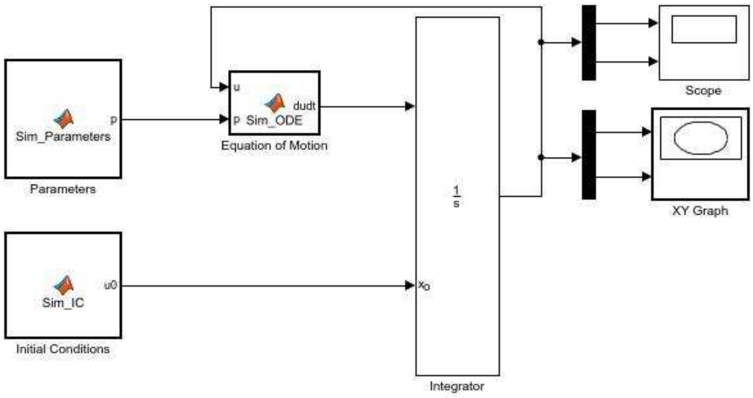


Figure 2: Frame of references and mechanical system

The Model has also the following blocks

1. *Integrator*. This *Continuous* block is used to integrate the ordinary differential equation. After create the block, click it with the right button of the mouse, open "Block Parameters", and set "Initial Condition Source " as "external".

2. Scope. This *Sink* block is used t o visualize the variables of the state vector versus time.

3. XY Graph. This *Sink* block is used to visualize the plot $\psi - \dot{\psi}$ (phase space).

In *Model Configuration Parameters* set the *maximum step size* to 0.01, *relative tolerance* to $10^{-6}$ and stop time to 100.

Run the Simulink model for $x_p = 0.1$ and $x_p = 0.06$ and try with several initial conditions. Compare your results with the diagrams constructed in Sec. 2.3.2.

## 2.4 Content of the report

1. Results of the analytical calculations.

2. Figures with $\psi$ versus $t$ and $\psi - \dot{\psi}$ for $x_p$ larger and smaller than $x_p^*$ and several representative initial conditions.

3. Discussion of the results while relating the dynamics in the $\psi$ versus $t$ and $\psi - \dot{\psi}$ diagrams. What type of motion does the disk perform as a function of the parameter $x_p$ and the initial conditions ?

4. Relate your results with normal aircraft configurations (relative position of the vertical stabilizer with respect to the center of mass of the aircraft). According to the Lab, why is the vertical stabilizer located at the tail ?

5. Using the Matlab model, and for $x_p > x_p^*$ and $x_p < x_p^*$, construct again the $\psi - \dot{\psi}$ diagram but adding a dissipation to the spring
$$\ddot{\psi} = \Omega^2 \left[\delta \cos\psi - 1\right] \psi - \dot{\psi}, \tag{5}$$

   Discuss the results.

[2]Note that if the output is a vector, it should be initialized before assigning the values of its components. For instance, write $\boldsymbol{p} = zeros(2, 1)$ before writing $p(1) = \Omega$ and $p(2) = \delta$ in function *Parameters*

# 3 Lab 2: Aircraft Performance

## 3.1 Aim

In this lab the student will practice with the performance equations found in the theoretical sessions. They will also run simulations of more complex models that remove some of the hypothesis used in the classroom. Cruise, descent, and turning manoeuvres will be considered. Students will formulate the equations of motion with both physical and dimensionless variables and will acquire knowledge on Matlab and Simulink. The calculations will be carried out for the aircraft of parameters given in Table 3 and the following hypotheses:

1. Drag polar.
2. Thrust model.
$$T(\Pi, h) = \Pi \times T_{max}\mid_{SL} \left(\frac{\rho(h)}{\rho\mid_{SL}}\right)^{0.7} \tag{6}$$
3. International Standard Atmosphere.
4. Initial weight of the aircraft for <u>each</u> manoeuvre is equal to MTOW.

| Parameter | Symbol | Value |
|---|---|---|
| Wingspan | b | 47.574m |
| Surface | S | 286.15$m^2$ |
| Aspect Ratio | Æ | 7.9 |
| Oswald Efficiency | $e_0$ | 0.8 |
| Zero-lift drag coefficient | $C_{D0}$ | 0.018 |
| Maximum Thrust at sea level | $T_{max}\mid_{SL}$ | 444822N |
| Specific fuel consumption | $SFC_T$ | 0.000167$s^{-1}$ |
| Maximum Takeoff Weight | $MTOW$ | $1.333 \times 10^6 N$ |

Table 3: Aircraft parameters

Students have access to the following subroutines in Aula Global:

1. <u>*Parameters*</u>. Goal: compute a vector $\boldsymbol{p}$ with the parameters. Function input: none. Function outputs: aircraft parameters.
2. <u>*ISA*</u>. Goal: compute atmospheric values. Function inputs: altitude (m). Function outputs: air density, pressure, temperature and speed of sound.
3. <u>From_D_2_ND</u>. Goal: transform variables with dimensions to dimensionless variables. Function inputs: thrust, velocity, aircraft parameters, altitude, and weight. Function outputs: dimensionless thrust and velocity, and characteristic thrust and velocity.
4. <u>From_ND_2_D</u>. Goal: transform dimensionless variables to variables with dimensions. Function inputs: dimensionless thrust and velocity, aircraft parameters, altitude, and weight. Function outputs: thrust, velocity, and characteristic thrust and velocity.
5. <u>Per_Cruise</u>. Goal: given the dimensionless thrust (or velocity), compute the dimensionless velocity (or thrust) in cruise conditions. Function inputs: dimensionless thrust or velocity and flag to select them. Function outputs: dimensionless velocity and thrust in cruise conditions.
6. <u>ODE_Cruise</u>. Goal: compute the equation of motion at cruise. Function inputs: times and cruise state vector (see below). Function outputs: time derivative of the cruise state vector.
7. <u>Fun_Control</u>. Goal: compute the control vector. Function inputs: Flag to select the type of manoeuvre, state vector, environment vector, and parameters of the aircraft. Function outputs: control vector.

Hereafter we will call *performance equations* to the models developed in the theoretical sessions, which typically assume stationary conditions and constant mass. In this document, the *performance equations* are given in dimensionless form. We will call *simulation equations* to the models that remove these assumptions. This latter set of equations will be written with dimensions. Tasks with the symbol ★ should be included in the report.

## 3.2 Cruise

Students will study the aircraft motion in cruise conditions. The *performance equations* are

$$\hat{T}(\hat{v}) = \frac{1}{2}\left(\hat{v}^2 + \frac{1}{\hat{v}^2}\right), \quad or \quad \hat{v}(\hat{T}) = \sqrt{\hat{T} \pm \sqrt{\hat{T}^2 - 1}} \tag{7}$$

and the *simulation equations*

$$\frac{dv}{dt} = (T - D)\frac{g}{W}, \qquad \frac{dx}{dt} = v, \qquad \frac{dh}{dt} = 0, \qquad \frac{dW}{dt} = -SFC_T T \tag{8}$$

The cruise state vector is $\boldsymbol{u}_s = [v,\ x,\ h,\ W]$ and the cruise control vector $\boldsymbol{x}_c = [\Pi]$. Note that the pilot needs to adjust the $C_L$ (angle of attack) to the value $C_L = 2W/\rho S v^2$, which comes from $L = W$, in order to keep constant altitude.

### 3.2.1 Tasks

1. Write a Matlab programme that makes the following actions:
   (a) Define vector $\boldsymbol{p}$ as a global variable.
   (b) Make a call to *Parameters*
   (c) Generate a vector $\boldsymbol{u}_s(0)$ with the following initial conditions: $x_0 = 0$, $H_0 = 10000m$, $W_0 = MTOW$, and $v_0$ equal to 1.2 times the base velocity for that initial conditions.
   (d) Using a Matlab numerical integrator, for instance *ode45*, integrate the equations of motion with initial condition $\boldsymbol{u}_s(0) = [v_0,\ x_0,\ H_0,\ W_0]$ and final time equal to one hour. Students can use the function *ODE_Cruise*.
   (e) Make a figure with four plots: velocity, distance, height, and weight versus time. ★
   (f) From the numerical integration, compute the variables $\hat{T}$ and $\hat{v}$. Make a plot with the trajectory $\hat{v} - \hat{T}$ and the analytical relation given by Eq. 7. ★
   (g) Repeat the calculations with $v_0$ equal to 0.6 and 0.4 times the base velocity. Discuss the results. ★
2. Construct the Simulink model of Fig. 3.2.1
   (a) Repeat the simulations carried out with Matlab and compare the results.
   (b) Discuss the advantages and disadvantages of working with Matlab and Simulink. ★

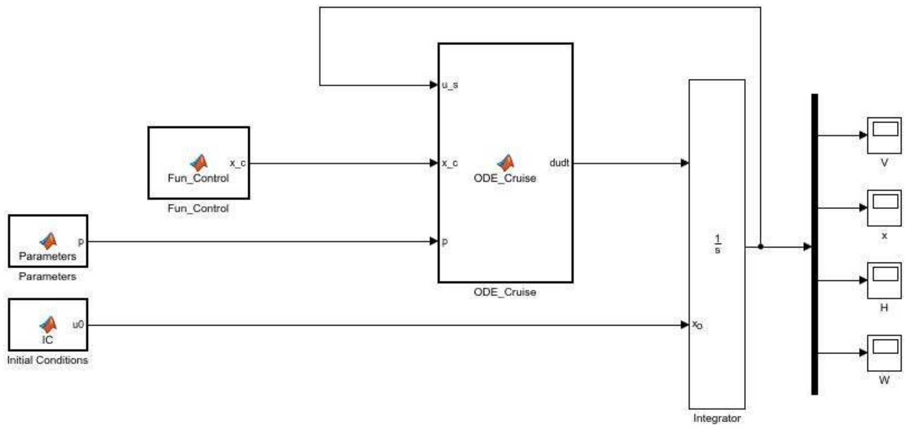


Figure 3: Simulink model for cruise performance computation.

## 3.3 Motion in a vertical plane with constant velocity elevation angle

The motion of the aircraft in a vertical plane with $\dot{\gamma} = 0$ will be studied. The *performance equations* are

$$\hat{T}(\hat{v},\gamma) = \frac{1}{2}\left(\hat{v}^2 + \frac{\cos^2\gamma}{\hat{v}^2}\right) + E_m \sin\gamma, \quad or \quad \gamma\left(\hat{v},\hat{T}\right) = \arcsin\left[E_m \hat{v}^2\left(1 \pm \sqrt{1 - \frac{2\hat{T}\hat{v}^2 - 1 - \hat{v}^4}{E_m^2 \hat{v}^4}}\right)\right] \tag{9}$$

and the *simulation equations*

$$\frac{dv}{dt} = (T - D - W\sin\gamma)\frac{g}{W}, \qquad \frac{dx}{dt} = v\cos\gamma, \qquad \frac{dh}{dt} = V\sin\gamma, \qquad \frac{dW}{dt} = -SFC_T T \tag{10}$$

The descent state vector is $\boldsymbol{u}_s = [v,\ x,\ h,\ W]$ and the descent control vector $\boldsymbol{x}_c = [\Pi, \quad \gamma]$. Note that the pilot needs to adjust the $C_L$ (angle of attack) to the value $C_L = 2W\cos\gamma/\rho S v^2$ to keep $\dot{\gamma} = 0$.

### 3.3.1 Tasks

1. Following the same methodology of *Per_Cruise*, *ODE_Cruise*, write the corresponding functions *Per_Vertical*[3], *ODE_Vertical*. Update the function *Fun_Control* with the control law for descent manoeuvre with constant $\gamma$.
2. Write a Matlab programme that makes the following actions:
   (a) Make a $\gamma$-isolines plot versus $\hat{v} - \hat{T}$. Use Matlab function named *contour*, the intervals $0.1 < \hat{v} < 3$ and $0.1 < \hat{T} < 3$, and plot the isolines with $\gamma = -25°, -10°, -2°, 0°$, and $2°$. ★
   (b) Compute the thrust level $\Pi$ that is necessary for the following manoeuvre $v = 1.2v\mid_{E_m}$, $\gamma = -2°$, $H = 15km$, $W = MTOW$. ★
   (c) Generate a vector $\boldsymbol{u}_s(0)$ with the following initial conditions: $x_0 = 0$, $H_0 = 15km$, $W_0 = MTOW$, and $v_0$ equal to 1.2 times the base velocity for that initial conditions.
   (d) Integrate the equations of motion numerically with the control inputs $\Pi = 0.5$ and $\gamma = -2°$, initial condition $\boldsymbol{u}_s(0)$, and final time equal to 15 minutes.
   (e) Plot the trajectory in the $\hat{v} - \hat{T}$ plane together with the $\gamma$-isolines. ★
   (f) Discuss the results. The control law imposed $\gamma = -2°$, Does the trajectory follow the $\gamma = -2$ isoline ?
3. Construct a Simulink model similar to the one in Fig. 3.2.1 for climbing and descent manoeuvres.
   (a) Repeat the simulation carried out with Matlab, and make another one with exactly the same inputs except the initial velocity that is equal to 0.5 times the initial base velocity.

## 3.4 Turning manoeuvre

Students will study coordinated turning manoeuvres with $\gamma = 0$. The *performance equations* are

$$n\left(\hat{v},\hat{T}\right) = \frac{1}{\cos\mu} = \hat{v}\sqrt{2\hat{T} - \hat{v}^2}, \qquad \frac{v\mid_{E_m}\dot{\chi}}{g}\left(\hat{v},\hat{T}\right) = \sqrt{2\hat{T} - \hat{v}^2 - \frac{1}{\hat{v}^2}}, \qquad \frac{Rg}{v\mid_{E_m}^2} = \frac{1}{\sqrt{\frac{2\hat{T}}{\hat{v}^2} - 1 - \frac{1}{\hat{v}^4}}} \tag{11}$$

and the *simulation equations*

$$\frac{dv}{dt} = (T - D)\frac{g}{W}, \qquad \frac{dx}{dt} = v\cos\chi, \qquad \frac{dy}{dt} = V\sin\chi, \tag{12}$$

$$\frac{dH}{dt} = 0, \qquad \frac{d\chi}{dt} = \frac{L}{W}\frac{g}{v}\sin\mu \qquad \frac{dW}{dt} = -SFC_T T \tag{13}$$

Note that the pilot needs to adjust the $C_L$ (angle of attack) according to $W = L\cos\mu$ for keeping $\dot{\gamma} = 0$.

### 3.4.1 Tasks

1. Identify the state vector and the control vectorin the *simulation equations*. ★
2. Following the same methodology of *Per_Cruise*, and *ODE_Cruise*, write the corresponding functions *Per_Turning* and *ODE_Turning*. Update the function *Fun_Control* with the control law for turning manoeuvres with constant $\mu$.
3. Make a a figure with four panels showing the isolines of $\mu$, $n$, $Rg/v\mid_{E_m}^2$, and $v\mid_{E_m}\dot{\chi}/g$ versus $\hat{v} - \hat{T}$. ★
4. For $H_0 = 10km$, compute the thrust level $\pi$ and bank angle $\mu$ corresponding to $\hat{v} = 1.1$ and $\hat{T} = 1.2$. Find also the corresponding velocity $V_0$

[3]Note that you will need to add vector $\boldsymbol{p}$ as an input and $\gamma$ as output (take the one with the minus sign in Eq. 11)

5. Integrate the equations numerically with initial conditions equal to $V_0$, $X_0 = 0$, $Y_0 = 0$, $H_0$, and $\chi = 0$, integration time equal to 5 hours, and with the constant thrust level and bank angle computed in the previous question. Make a figure with four panels: (i) velocity versus time, (ii) $\chi$ versus time, (iii) $X$ versus $Y$ and, (iv) $W$ versus time. Discuss the results. ★

6. Plot $\hat{v}(t)$ versus $\hat{T}(t)$ provided by the numerical integration. Plot in the same figure the isolines of $\mu$ given by the *performance equations*. Discuss the results. ★

### 3.5 Content of the report

Prepare a report with the results of all the tasks that have been marked with ★

## References


[1] A. Deperrois *About stability analysis using XFLR5 - Revision 2.1.* November 2010
`http://www.xflr5.com/docs/XFLR5_and_Stability_analysis.pdf`

[2] *xflr5. Analysis of foils and wings operating at low Reynolds numbers.* XFLR5 v6.02, 28/02/2013. `https://engineering.purdue.edu/~aerodyn/AAE333/FALL10/HOMEWORKS/HW13/XFLR5_v6.01_Beta_Win32(2)/Release/Guidelines.pdf`

[3] `http://airfoiltools.com/search/index`