Warsaw University of Technology

FACULTY OF
POWER AND AERONAUTICAL ENGINEERING

Institute of Aeronautics and Applied Mechanics

Bachelor's diploma thesis

Aerospace Engineering
Mechanics of flight

MISSILES GUIDANCE LAWS: TRAJECTORY CALCULATIONS

Carlos Alcolea Pérez
Student record book number K-4972

Thesis supervisor
Prof. dr hab. inż. Krzysztof Sławomir Sibilski

Warsaw, 2019

Contents

Images

Tables

1. ABSTRACT.

This project corresponds to a final student project in the degree of Aeroespace Engineering, and it is focused on calculating with Matlab R2015a trajectories of a missile heading to its target following different guidance laws. The main requirement for the algorithm is to be able to calculate the trajectory of the missile no matter the movement of the target, representing the real path of the missile. In addition, with this project it is also expected to perform a comparison among classical guidance laws (Pure Pursuit, Proportional Navigation and Line of Sight) according to the load factor experienced by the missile along its flying path and to study the changes in the trajectory and load factor with missile's velocity and wind drift. In order to develop the practical work, it is necessary to understand the basic concepts of target acquisition parameters (kind of guidance, systems and used sensors for navigation) as well as the systems that missiles use for obtaining its own position.

Key words: missile; target; guidance; load factor; acquisition parameters; navigation; trajectory; Pure Pursuit Guidance; Proportional Navigation Guidance; Line of Sight Guidance.

2. GUIDANCE SYSTEMS.

Guidance is defined as the way a missile heads or is headed towards a target, and it is divided into three phases: launch, midcourse and terminal. Launching is the phase where the missile is boosted. The missile may be guided from the beginning of the phase or can be launched without any guidance. The midcourse phase is the longest in terms of time and distance, and guidance is normally required unless the missile is a ballistic one. In the last phase, the terminal one, the missile is guided towards its target, therefore the accuracy of the guidance system is extremely important.

2.1. Auto guided systems.

In auto guided systems, the missile itself includes the necessary elements for target acquisition parameters, guidance and navigation and control. There are three auto guided systems: active homing, semi-active homing and passive homing.

2.1.1. Active homing.

In active homing, the missile contains a transmitter, which emits energy towards the target, and a receiver, responsible for detecting the reflected energy. In order to properly detect the target, an error calculator system is required, whose aim is to remove from the received signal unwanted information coming from other objects or surfaces closed to the missile, obtaining a clear signal from the target. From that signal, a calculator predicts the future position of the target and sends information to the missile's auto pilot, which is responsible for deflecting the corresponding aerodynamic surfaces, heading the missile towards its target. These kind of missiles, have the main advantage of being completely autonomous, since they do not require external systems for guidance.

In this kind of missiles, the transmitter consists of a microwave radar, with typical bands of C(4 Ghz), X(6 Ghz) and K(15 Ghz) or a millimetre-wave (MMW) radar with 94 GHz frequency, placed inside the missile, therefore, the main problem it would be its size and weight. From the radar range equation:

$$R_{max} = \left(\frac{P_a \cdot {G_s}^2 \cdot \sigma \cdot \lambda^2}{64 \cdot \pi^3 \cdot P_{min}} \right)^{\frac{1}{4}}$$

Where:

- R_{max}: maximum radar range
- P_a: transmitted power
- σ: effective surface of the target to the radar
- G_s: antenna gain.
- λ: wavelength
- P_{min}: minimum power required for detection

Considering the antenna gain $G = k \cdot \frac{A}{\lambda^2}$, where k is a constant and A is the antenna area, for a given transmitted power and wavelength and a certain target size, $R_{max}=f(A, P_{min})$. Since area of the antenna is limited by the missile's diameter, the range of active homing systems is limited as well.

2.1.2. Semi-active homing.

In semi-active homing systems, from the launch base the target is pointed with a radiation source, which is reflected by the target and detected by the missile's receptor. Then, the systems works exactly the same as in active homing: with the received signal, a calculator predicts the following position of the target, and the autopilot will deflect the corresponding external surfaces to head the missile towards its target. The main difference with the previous kind of homing is that there is no transmitter placed inside the missile, which allows to reduce the weight and simplify the system components. Nevertheless, the missile will have a dependence from the launch base radar for the whole trajectory until impact, and if the reflected radiation by the target is lost, the guidance could not continue.

In the case of a semi-active guidance, the radar range equation states:

$$R_{max} = \left(\frac{P_s \cdot G_t \cdot G_s \cdot \sigma \cdot \lambda^2}{64 \cdot \pi^3 \cdot P_{min}} \right)^{\frac{1}{4}}$$

Where:

- P_s: transmitted power by launch base radar
- G_t: transmitter antenna gain.
- G_s: receiver antenna gain.

It is seen that, if P_s and G_t are greater than P_a and G_s, respectively, the range of the semi-active homing will overcome the one of the active homing.

In those systems the radiations used by the launch base transmitter depend on its application: For air to air, and surface to air missiles, microwave radiation is normally band C(4 Ghz), X(6 Ghz) and K(15 Ghz) and for surface to surface applications, IR laser with wavelength 1.1 µm is used.

2.1.3. Passive homing.

Passive homing missiles are characterized by the lack of transmitter and total independence from launch base. The missile is equipped with a receiver which detects the radiation of the target as well as a guiding system to head the missile towards its target. However, its use is constrained to the amount of energy reflected by the target. That is why its use is limited and has a great dependence from atmospheric conditions and can only be applied for low ranges.

The seeker of the system may work with IR radiation, TV location or MMW radiation. Regarding, the location by IR radiation, the sensor detects heated flow, which comes from the engines of the target. The actual temperature of the target is not important, but the difference in temperature between the target and its surroundings. In these cases, the sensor is cooled to remove the inner temperature of the missile itself and allow greater sensibility for the detection of external flow temperature variations, its use, however, is limited to non-rainy days as the water steam easily absorbs heat. They receive wavelengths from 1.2 µm to 4.3 µm. With respect to TV location, a camera is placed on the head of the missile which by applying optical filters is capable to detect the position of the target, with the limitation that during low light hours such as night, its efficiency is very low and the guidance would be really poor. Finally, it can be used the natural radiation of the bodies (MMW radiation from 34GHz to 94 GHz band) to detect the target, however, it can be only applied for low ranges.

2.2. External guidance.

In this method, all the guidance instructions are sent to the missile by an external source. The missile contains a receiver which obtains the information regarding the trajectory it must follow. Then, the receiver sends a signal to the autopilot which performs the calculated trajectory manoeuvres. Regarding the command guidance, it is possible to differentiate two methods: Command and Beam-rider Method.

2.2.1. Command guidance.

Command guided systems can be split in radar guidance and optical guidance. Regarding the first category, the missile and target position are detected by means of one or two radars, whose microwave radiation is normally band S (2 GHz) C, X or K. For both bodies it is obtained its distance with respect to the radar as well as angles, and with that data, it is calculated the deviation from the missile to the pre-established trajectory and the error signal is transmitted to the missile so that it redirects its path. With respect to the second category, the tracking of the bodies is performed optically with an automatic TV whose working principle matches the one explained in passive homing.

Independently from the tracking method used, command systems contained the following functions: tracking, error calculator, transmission and missile reaction.

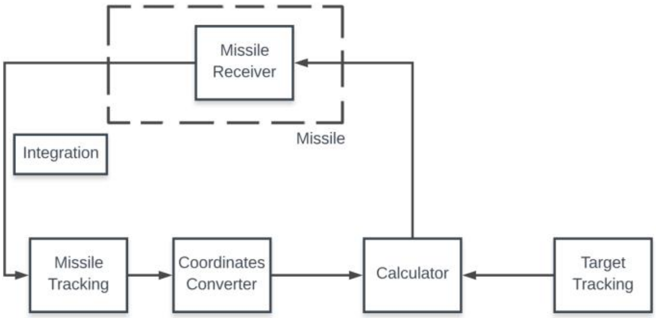

Image 1: Command guidance working scheme.

- Tracking: The radar provides information about the bodies distance, angles and its derivatives.
- Coordinates converter: Prior to its input to the calculator, the information by the tracking sensor must be compatible, therefore a common system of reference for target and missile is needed.
- Calculator: With the receiving information from the previous systems (missile and target rates, elevations and bearings), it calculates the manoeuvre movements that the missile must follow and generates a steering signal, with high frequency bands to avoid losses due to the atmosphere: HF(3-30 Mc/s), VHF (30-300 Mc/s) and UF(300-3000 Mc/s).
- Missile receptor: processes the receiving signal and sends information to autopilot for performing trajectory changes.

2.2.2. Beam-rider Method.

In this case, the target is pointed with a radar and the missile receives the emission and uses it for maintaining the flying trajectory under the beam axis, which means that the missile's flight path control unit detects when the missile is at the beam axis of the guidance radar as well as any deviation from it and is able to correct the path. This method has two main advantages with respect to the previous one: firstly, just one radar (which must have conical scan for providing at the same time both target tracking and correction of the beam axis for the missile) is needed, and secondly, several missiles can be launched simultaneously as the missile formulates its own directional commands.

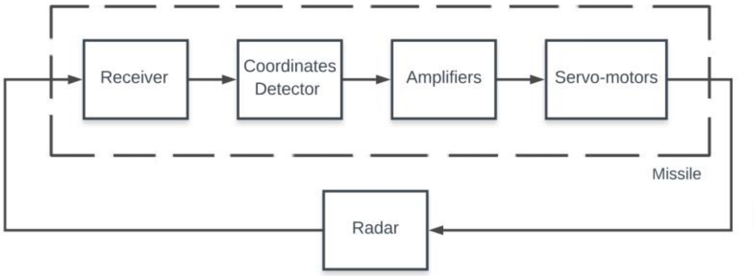

Image 2: Beam-rider method working scheme.

The radar is sending constant error signals to the missile. They consist on deviations of the missile position with respect to the beam axis: amplitude, direction and sense. The missile receiver, which consists of a fix antenna, detects the radar's signal and transmits the information to the Coordinates Detector, which separates the coordinates of pitch and yaw so that the servo-motors change its attitude towards the centre of the beam, reducing its position error.

A similar type of guidance is the Line of Sight (LOS) guidance, where the target is followed with a radar or laser. The position variation of the target allows to calculate its angular rate, and therefore a steering command is sent to the missile in such a way that it is forced to have the same angular rate as the target, being at all time in the line between the target and the point where the tracker radar is placed. If the missile velocity is great enough, the missile will hit its target.

3. NAVIGATION SYSTEMS.

Navigation systems are the equipment needed by the missile in order to determine its state vector: position, velocity and attitude. The main navigation systems used are: Inertial Navigation, Terrain Reference, and Satellite navigation.

3.1. Inertial Navigation.

Inertial navigation consists of obtaining the position of the missile by a double integration of the acceleration vector measured by accelerometers, and the attitude by gyroscopes placed inside the missile. The main advantage of this kind of systems is that they are autonomous, since the position of the missile is determined by the on-board equipment without transmitting nor receiving any kind of radiation, However, this systems is characterized by the error propagation, since every time, the current position of the missile is estimated from the previous calculated result. In consequence, an error correction system is required.

3.1.1. Sensor’s principle of operation.

Accelerometers are sensors which measure the applied acceleration in a given direction by measuring the displacement of a mass suspended by a spring.

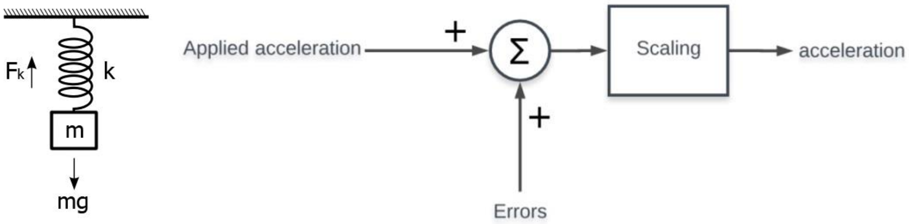

Image 3: Accelerometer's principle of operation

By Hooke’s Law $F_k = k \cdot x$, it is possible to obtain the force of the spring due to the displacement of a mass. Then, by applying second Newton’s law $F = m \cdot a$, where the only force acting on the mass (apart from the gravitational one) is the elastic force, both terms are equal: $k \cdot x = m \cdot a$. In consequence, the sensor is measuring the acceleration on the longitudinal axis of the spring. Therefore, on a missile, three accelerometers must be placed orthogonally in order to measure all accelerations. However, gravity acceleration is not measured since the gravity is acting constantly and therefore the displacement of the mass does not change over time due to gravity, that is why, a gravity calculator s needed. In addition, the sensor’s case is placed rigidly to the vehicle so that missile and sensor have the same acceleration.

With respect to gyroscopes, they are used in order to obtain the attitude of the missile so that the acceleration vector can be double integrated obtaining velocity and position respectively. As well as in accelerometers, three gyroscopes are needed in order to measure yaw, pitch and roll.

In order to measure yaw angle, a gyroscope is placed spinning on Y axis. When the missile rotates about the yaw (Z axis), the gyroscope inertia causes the gyroscope to resist the change in its rotational axis about the Y axis. This resisting force is acting against a spring, so that, the greater the yaw the greater the deflection of the gyroscope will be. It is important to highlight, that rotations about the Y axis will have no effect since the axis of rotation would coincide with the spin axis and rotations around X axis are not feasible and will only produce reaction forces on the bearings.

Regarding the pitch and roll measurements, the gyroscopes are mounted on a powered rotor spinning on the vertical axis (Z axis) placed on a platform. Therefore, the accuracy of the indicators depends on the accuracy of the vertical. Pitch and roll angles are measured as rotation angles of the rotor with respect to the platform it is mounted. If the spin axis is deviated from the vertical, torque motors are switched on and move the rotor back to its previous position.

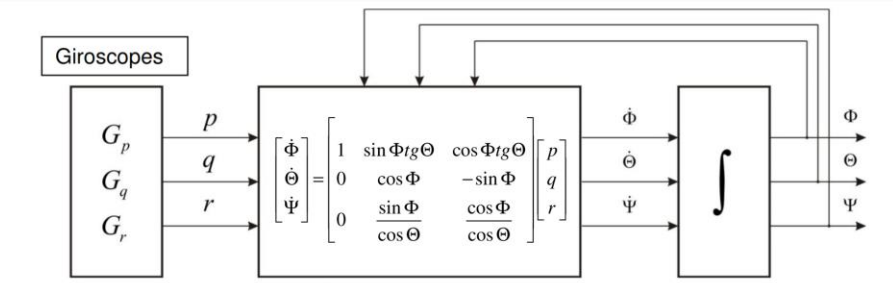

Image 4: Gyroscope's working scheme

In gyroscopes sensors, it is needed an integration process in order to obtain final attitude angles. The sensors may be mounted on a gimbal or strapdown platform, being this last one the most used nowadays.

The final scheme of the working of the system can be displayed as follows:

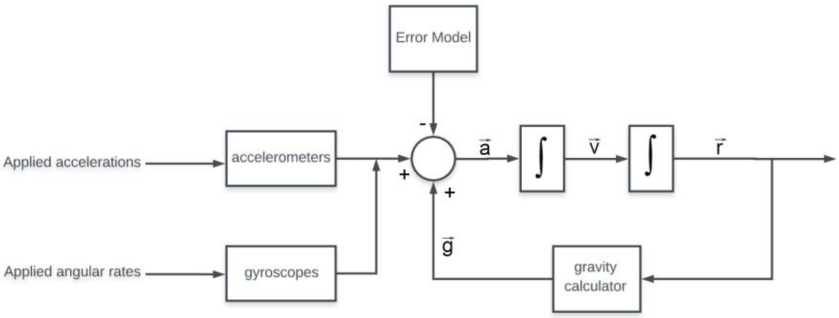

Image 5: INS signal processing

3.2. Terrain Referenced Navigation.

As it was mentioned before, the main disadvantage of inertial navigation systems is the error propagation, which makes that an error correction system is required. Terrain Referenced Navigation (TRN) can be used as a technique to update the position of a missile by comparing the characteristics of the terrain that the missile is overflying with the data stored in an on-board computer.

In order to determine position, first, the on-board computer needs to have the mean altitude of the terrain the missile will overfly, in order to do so, it is needed to divide the terrain in small regions and compute its mean altitude. Furthermore, it is needed a wide extension of terrain data in order to make sure that the missile will not get lost. Once the missile is flying over this terrain, the missile's altimeter radar will determine the mean altitude of the terrain. Comparing both measures, the missile would be able to determine its real trajectory and correct its path.

In the terminal phase of the missile, TRN is not suitable anymore since this system has an error around 30 m, which is not small enough in order to impact its target. Then, Terrain Feature Navigation (TFN) is used. This method is very similar to TRN but with higher precision. In this case, the on-board computer has images of forest areas, buildings, roads,… and the missile's sensors (infrared, MMW, radar,..) compares the surrounding environment with the storage images, allowing to impact with high precision.

3.3. Satellite Navigation.

Satellite Navigation is based on the use of GPS system in order to provide the current position (latitude, longitude an altitude) and timing with high precision. The GPS is made up to three segments: Space, composed by 24 satellites distributed in the Medium Earth Orbit, Control, composed by ground antennas and monitor stations whose aim is to update the satellite orbital parameters (ephemeris) and clock parameters, and User, which is composed by the civil and military users with the software and hardware necessary to be detected by the satellites.

Satellites are constantly sending information about its ephemeris and the time the message was sent. The on-board equipment needed by the User therefore, is a receiver which calculates its position by timing the signals sent by the satellites and compute the distance to each satellite. These distances and transit time are used in order to determine the position of the receiver.

$$(x_{sat\ i} - x_{user})^2 + (y_{sat\ i} - y_{user})^2 + (z_{sat\ i} - z_{user})^2 = c \cdot \Delta t_{sat\ i}$$

Where sub index “i” stands for the known coordinates of each one of the satellites.

In order to solve the previous system of equations at least four satellites are needed, three of those are needed to compute the x,y,z coordinates of the user and the fourth one to correct the clock error.

The main problem about GPS is the constant need of satellite signal, therefore this system is not generally used as a single one, but it is rather used as a combination with Inertial Systems (INS) in order to correct its error. GPS can also be used combined with Dead Reckoning technique in order to correct its error or be the main source of navigation in case of lack of satellite signal.

Dead Reckoning is a navigation technique which consists of estimating the future position every little time interval by knowing the current position and the velocity vector. Therefore, if the position is known and the velocity and heading of the body too, assuming constant velocity in a small time interval, the next position can be assumed, following this process iteratively the position in every moment is known. However, the main problem is the propagation error and the lack of accuracy after a long time of operation. That is why Dead Reckoning must be used accompanied by GPS or INS.

4. MISSILE GUIDANCE LAWS.

The main objective of a missile guidance law is to generate a steering guidance command based on a specific strategy used in order to hit the target. This guidance command is generated at very short time periods in such a way that can be assumed to be continuously generated. Guidance laws can be classified into classical guidance and modern but, for the scope of the project only classical laws, which are those which have been traditionally used in missiles are analysed, since they are the base for the majority of modern guidance laws. Those are: Pure Pursuit Guidance, Proportional Navigation Guidance and Line of Sight Guidance.

4.1. Matlab code implementation.

For implementing the Matlab code, it is intended to calculate the trajectory of a missile towards its target by applying a classical guidance law and obtain the coordinates of a trajectory that would coincide with the one that the missile would follow in real life. However, since the technology used in a real missile was not available, different assumptions and simplifications have been introduced in order to represent the behaviour of the missile. Those assumptions are the following ones:

- The trajectory of the target is fixed, which means, that the target path is programmed so that in all cases follows the same one. The main goal of this assumption is to be able to make comparisons among guidance laws, since in the project we are not focused on target's path but on the missile's one. Nevertheless, the trajectory of the target is randomly chosen only taking into account that it could be a trajectory that a target would be able to follow, since the objective is to obtain an algorithm capable of calculating any trajectory no matter which the path of the target is. There have been considered different cases: linear trajectory of the target with a parabolic avoidance manoeuvre and sinusoidal trajectory of the target.
- Missile and target velocities are constant, since for impact the condition that must be satisfied is that missile's velocity is greater than target's. Therefore, from the design point of a missile, the objective is to obtain a high speed flying vehicle, and although the target varies its velocity, as long as the missile flies at greater speed, impact will take place. In the code, the velocity of the missile is always greater than the velocity of the target in order to satisfy the impact condition, and therefore no accelerations need to be considered in the missile since the impact will take place anyway.
- No Earth rotation effects are taking into account since in the trajectory calculation, a low range guidance is implemented, the impact takes little time, and therefore, Earth rotation effects can be neglected as they influence very little the trajectory.
- Guidance commands are implemented every 0.01/0.1 seconds, in order to satisfy the principle of continuity. In addition, it is assumed in all cases that the real target acquisition system of the missile and autopilot would be ideal, and so on the missile knows the position of the target in every instant and is able to redirect its path. For navigation systems, Dead Reckoning technique is implemented for Pure Pursuit and Proportional Navigation Guidance, and LOS guidance for the Line of Sight law.

- For calculating the trajectory, we are considering the position of the missile and target by its centre of gravity position, In addition, the missile is considered to be guided since the launch phase.

4.1.2. Load factor calculation.

In every flying object, structural strength aspects are determinant in order that the vehicle is able to fly. During its operational life, a missile is subjected to multiple loads due to manoeuvre movements, wind gusts, induced vibrations, etc. In a lot of cases, those loads depend on the atmospheric conditions during flight, so it is very important to be able to calculate a parameter that indicates whether the missile would be able to perform the trajectory or it would collapse and will not be able to impact.

In a missile, its required manoeuvre is defined by the load factor, n, which can be defined as is the normal acceleration experienced divided by the gravity.

$$n = \frac{a_n}{g}$$

In order to calculate the load factor, it is needed to estimate the normal acceleration, which has been done in the following way:

Guidance commands are implemented every 0.01 seconds for Pure Pursuit Guidance and 0.1 seconds for Proportional Navigation and Line of Sight Guidance in order to reduce computational power. This means that every 0.01/0.1 seconds the real missile would receive the guidance signal, and its autopilot would deflect the corresponding surfaces in order to change the current path of the missile, which would lead to a change in its velocity vector. In the code, as it was mentioned previously, it is needed to assume that target acquisition system of the missile and autopilot would be ideal, and therefore every 0.01/0.1 seconds the velocity vector is being updated according to the guidance law used. In consequence, for the Pure Pursuit Guidance Law, and Proportional Navigation Guidance Law, in the code, the Dead Reckoning technique is implemented. This means that the missile, by knowing its current position and its velocity vector is able to calculate its future position at instant t+0.01/0.1 seconds. Iteratively, velocity vector of the missile is being updated and therefore its position over time too.

Since $a = \frac{dV}{dt}$, and taking into account that the results are being updated every 0.01/0.1 seconds it can be approximated that $a \approx \frac{\Delta V}{\Delta t}$. Since the missile is flying at constant velocity ($v_m = const$) the whole acceleration corresponds to the normal one as the tangential one is null. Then the calculation of the load factor consist of applying a direct formula.

$a = \sqrt{{a_t}^2 + {a_n}^2}$, where $a_t = 0,\ a = a_n = \frac{dV}{dt} \approx \frac{\Delta V}{\Delta t}$,

Therefore, $a_n = \sqrt{{a_x}^2 + {a_y}^2 + {a_z}^2}$, every component of the acceleration is calculated as follows:

$$a_x \approx \frac{\Delta V_x}{\Delta t} \quad ; \quad a_y \approx \frac{\Delta V_y}{\Delta t} \quad ; \quad a_z \approx \frac{\Delta V_z}{\Delta t}$$

If the missile had been flying varying its velocity, the Matlab function *polyfit* would have been needed in order to obtain a formula of the acceleration so that it is possible to calculate the contribution of the tangential one. However, as it was mentioned before, if the missile is flying at enough speed, it will impact no matter the target has accelerations. So as long as the velocity of the target is lower that the missile's velocity, this last one can remain constant.

4.2. Pure Pursuit Guidance Law.

The main idea of Pure Pursuit Guidance is that the missile's velocity vector is maintained pointing in all moment towards its target. Therefore, when implementing the code, it is needed to update every 0.01 seconds, the velocity vector of the missile in such a way that it is coincident with the line of sight missile-target:

If $\overrightarrow{r_m} = \begin{pmatrix} x_m \\ y_m \\ z_m \end{pmatrix}$ denotes the vector position of the missile, and $\overrightarrow{r_t} = \begin{pmatrix} x_t \\ y_t \\ z_t \end{pmatrix}$ the vector position of the target, with respect to the origin of reference which is assumed to be on the launching platform of the missile, the line of sight vector can be computed as: $\overrightarrow{r_{mt}} = \begin{pmatrix} x_t - x_m \\ y_t - y_m \\ z_t - z_m \end{pmatrix}$, and therefore $\overrightarrow{V_m} = V_m \cdot \frac{\overrightarrow{r_{mt}}}{|\overrightarrow{r_{mt}}|}$.

Updating periodically the calculated parameters: missile velocity vector, missile and target position, we obtain the corresponding trajectories.

4.2.1. Case 1: linear trajectory of the target with parabolic avoidance manoeuvre

For the linear trajectory, the velocity vector is defined by $\overrightarrow{V_t} = vt \cdot \frac{\begin{pmatrix}1\\0\\1\end{pmatrix}}{\sqrt{2}}$. Then, the parabolic manoeuvre is performed by the target when the missile is at 5000 m distance. In order to be a real trajectory, the vertex of the parabola is located at the point in which the target is in that instant therefore: the velocity vector of the curve is defined by $\overrightarrow{V_t} = vt \cdot \frac{\begin{pmatrix}1\\0\\m\end{pmatrix}}{\sqrt{1+m^2}}$, where $m = 2 \cdot a \cdot xt - 2 \cdot a \cdot xt_0$, being xt_0 the x coordinate of the target when the missile is at 5000 m distance, xt is the current position of the target, which will be updated over time, and $a = \frac{1}{10000}$.

For this case, it is considered that the initial coordinates of the target are $\overrightarrow{r_t} = \begin{pmatrix}1000\\2000\\10000\end{pmatrix}$ m and $\overrightarrow{r_m} = \begin{pmatrix}0\\0\\0\end{pmatrix}$ m, considering that the origin of the system of reference coincides with the launching platform of the missile.

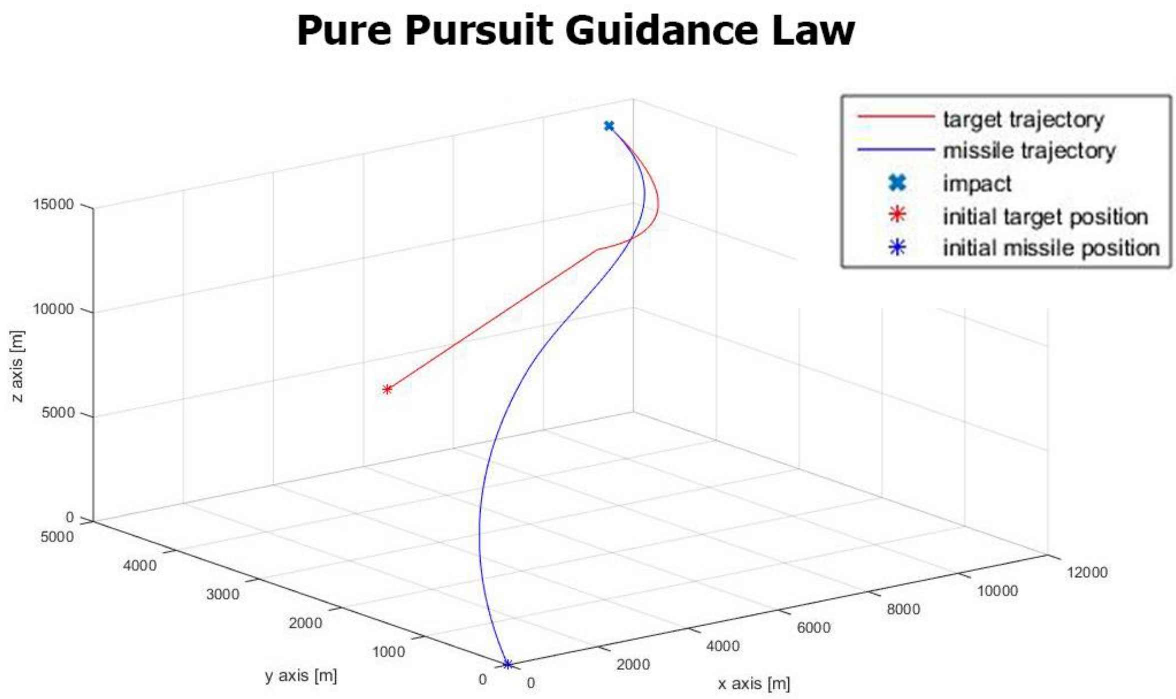

Image 6: Pure Pursuit, Case 1, trajectory plot

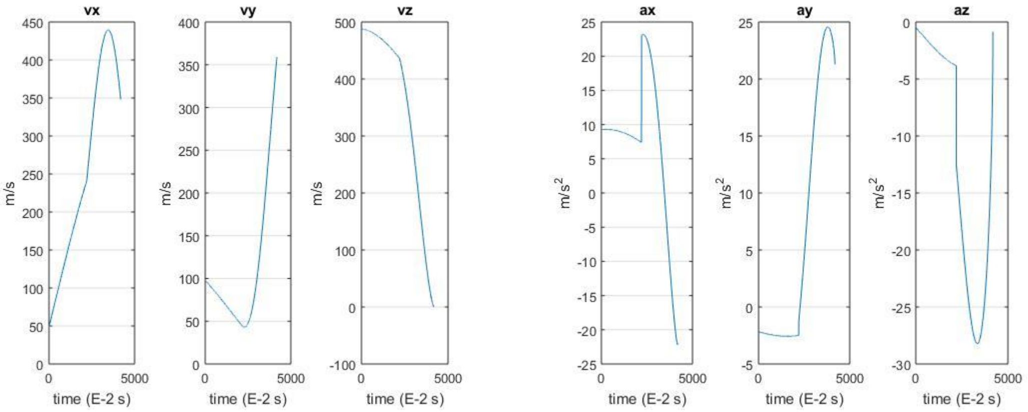

Image 7: Pure Pursuit, Case 1, velocity and acceleration plots

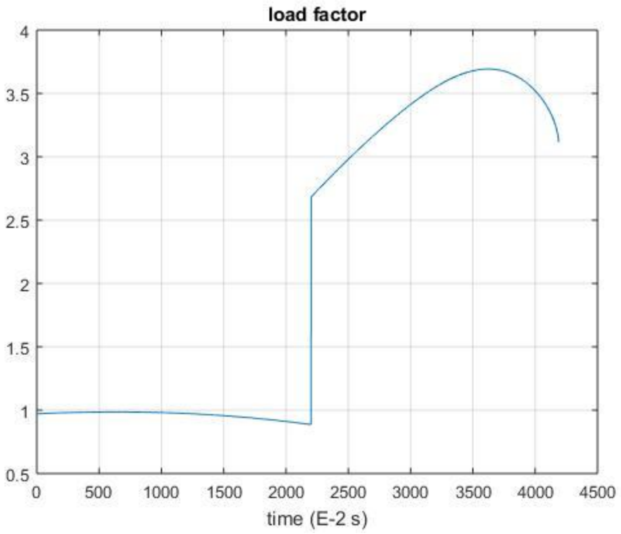

Image 8: Pure Pursuit, Case 1, load factor plot

In the first part, which corresponds to the linear trajectory, the missile's load factor is quite low. However, when the target manoeuvres (second part), as a consequence of the sudden change in the flight direction, there is a peak in each of the components of the acceleration which is translated into a sudden increase in the load factor.

4.2.2. Case 2: sinusoidal trajectory of the target.

In this case, the curve is defined by $\overrightarrow{V_t} = vt \cdot \frac{\begin{pmatrix} 1 \\ m \\ 0.5 \end{pmatrix}}{\sqrt{1+0.5^2+m^2}}$, where

$m = \cos\left(\frac{xt}{2000}\right)$, being xt the current position of the target which will be updated over time. For this case, it is considered that the initial coordinates of the target are $\overrightarrow{r_t} = \begin{pmatrix} 7000 \\ 5000 \\ 10000 \end{pmatrix}$ m and $\overrightarrow{r_m} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$ m.

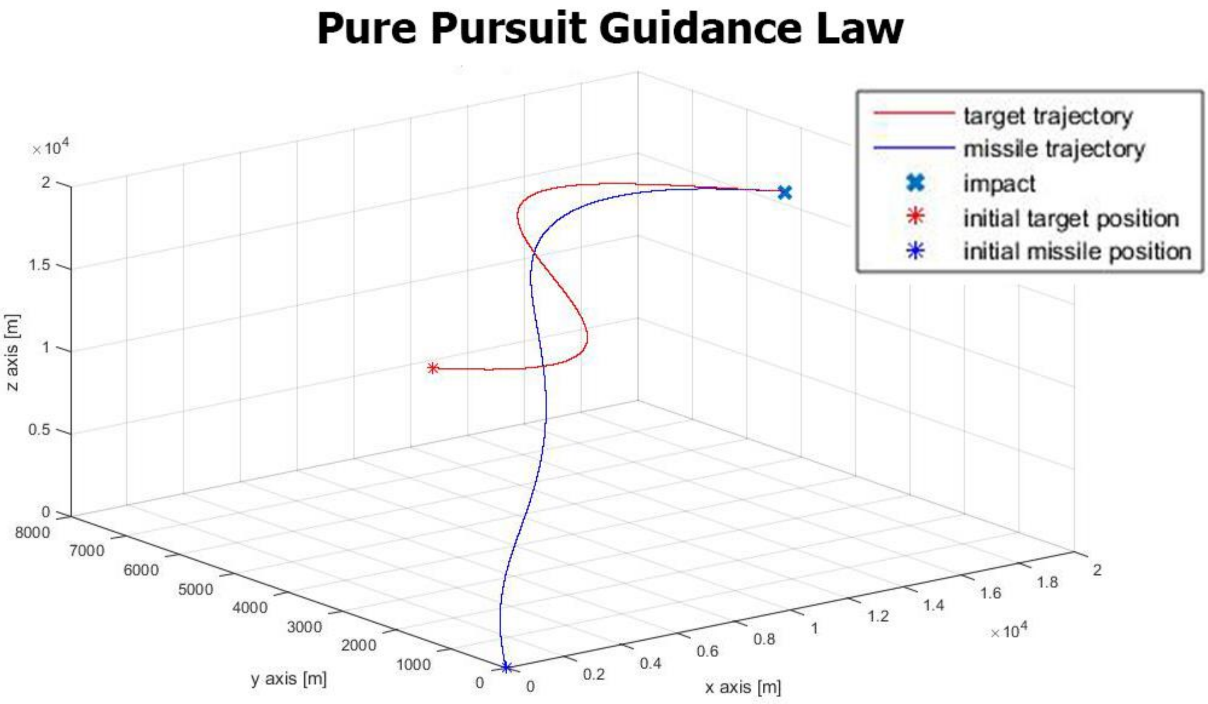

Image 9: Pure Pursuit, Case 2, trajectory plot

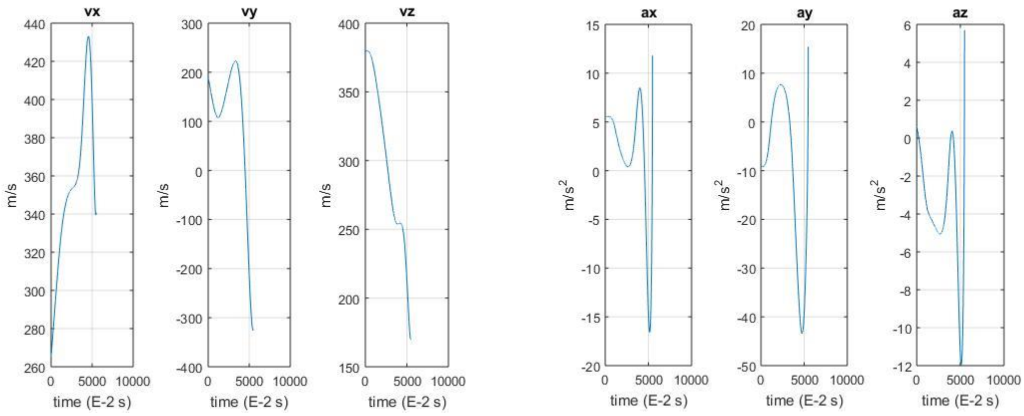

Image 10: Pure Pursuit, Case 2, velocity and acceleration plots

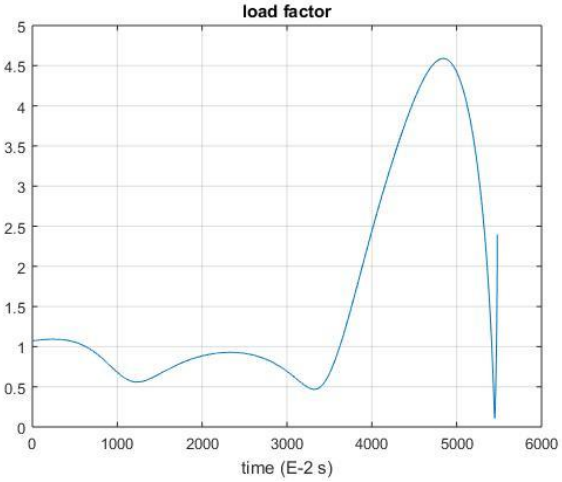

Image 11: Pure Pursuit, Case 2, load factor plot

The sinusoidal trajectory of the target can be considered as a path where the target is manoeuvring several times. Therefore, the missile needs to correct its path too, that is why, in the load factor graph three peaks are seen, each one corresponding to one manoeuvre made by missile. Up to this point, it is seen the first idea previously mentioned: manoeuvering is translated into a load factor increase due to the structural loads acting on the body.

4.3. Proportional Navigation Guidance Law.

In this guidance law, the objective is to approximate the real trajectory to the instantaneous ideal collision path, which is a linear impact trajectory so that the missile has no normal acceleration.

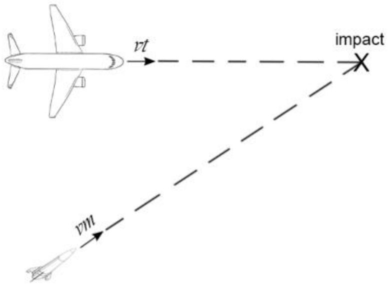

Image 12: Ideal collision path

Therefore, the missile's guidance system computes the instantaneous impact point considering that the target is following a straight trajectory characterized by its instantaneous velocity vector, and the missile follows a linear trajectory towards this point. This assumptions is being updated every little time, so that the missile's autopilot is able to correct the trajectory of the missile, and keep heading towards its target.

In the Matlab code, for implementing that condition we calculate the time it takes the missile to reach the instantaneous impact point

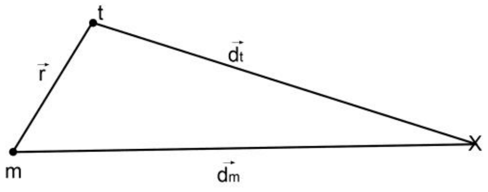

Image 13: Instantaneous impact path geometry

$\overrightarrow{dm}$: distance vector of the missile to impact.

$\overrightarrow{dt}$: distance vector of the target to impact.

$\vec{r}$: line of sight missile-target vector

Since we are working at constant target and missile velocity:

$$\vec{r} = \begin{pmatrix} x_t - x_m \\ y_t - y_m \\ z_t - z_m \end{pmatrix} \qquad \overrightarrow{dt} = \begin{pmatrix} v_{tx} \\ v_{ty} \\ v_{tz} \end{pmatrix} \cdot t \qquad \overrightarrow{dm} = \vec{r} + \overrightarrow{dt} = \begin{pmatrix} (x_t - x_m) + v_{tx} \cdot t \\ (y_t - y_m) + v_{ty} \cdot t \\ (z_t - z_m) + v_{tz} \cdot t \end{pmatrix}$$

$$\left|\overrightarrow{dm}\right| = vm \cdot t$$

Therefore:

$$vm \cdot t = \sqrt{((x_t - x_m) + v_{tx} \cdot t)^2 + ((y_t - y_m) + v_{ty} \cdot t)^2 + ((z_t - z_m) + v_{tz} \cdot t)^2}$$

With *solve* Matlab function, it is possible to solve the previous equation and obtain the time. This equation will be calculated iteratively since it is necessary to update continuously the position of the missile and target in order to find the instantaneous time for impact.

The velocity vector of the missile in each guidance command (t=0.1 s) would be defined by the distance vector of the missile to impact ($\overrightarrow{dm}$) over the calculated time (t):

$$\overrightarrow{Vm} = \frac{\overrightarrow{dm}}{t}$$

4.3.1. Case 1: linear trajectory of the target with parabolic avoidance manoeuvre.

In order to the later comparison with the Purse Pursuit Guidance Law and Line of Sight Guidance in case 1, we define the same trajectory as in the previous guidance law, and we obtain the following results:

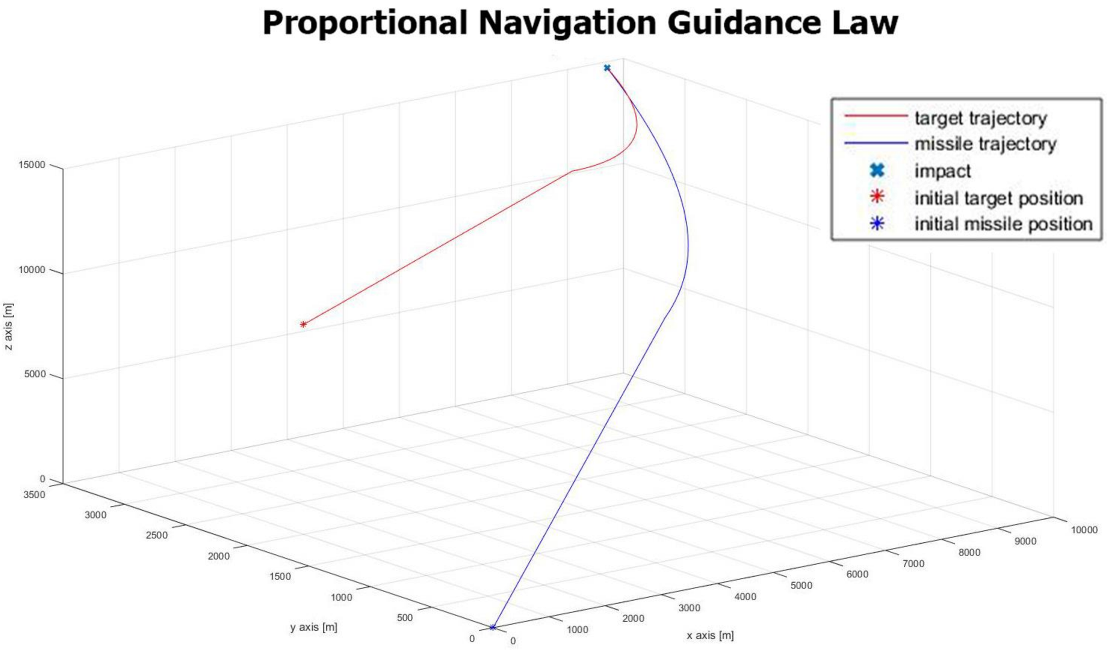

Image 14: Proportional Navigation, Case 1, trajectory plot

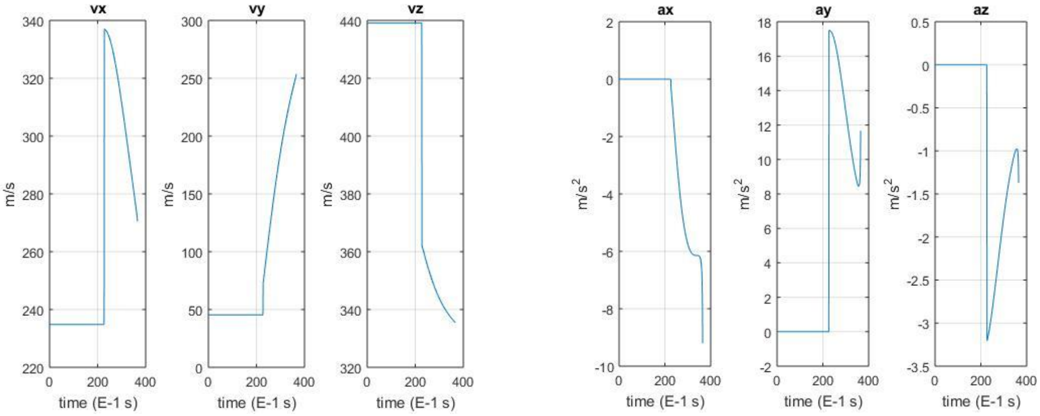

Image 15: Proportional Navigation, Case 1, velocity and acceleration plots

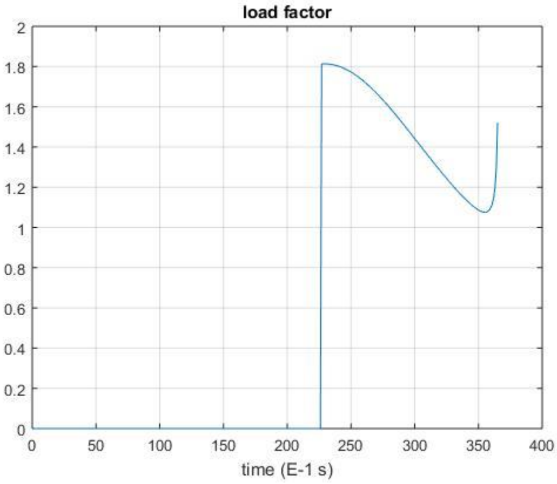

Image 16: Proportional Navigation, Case 1, load factor plot

It is seen that following the Proportional Navigation Guidance Law, when the target is following a linear trajectory, the missile bears no load factor, as it does no manoeuvre. In addition, it is noticed, that the velocity of the missile is constant and the acceleration null along the first part of the trajectory, since in every guidance command (t=0.1 s), the instantaneous collision point is the same as in the previous one, so the missile is following an ideal collision trajectory. However, when the target suddenly changes its trajectory, there's a load factor jump, confirming the aforementioned idea of the impact of manoeuvres on load factor. Nevertheless, it is seen that the load factor jump is lower than in Pure Pursuit Navigation Law. Later on, we will analyse if this fact is just a coincidence or Proportional Navigation presents generally lower load factor than Pure Pursuit.

4.3.2. Case 2: sinusoidal trajectory of the target.

Again, in case 2, we define the same trajectory as in the previous guidance law, to being able to make a result comparison. The following data is obtained:

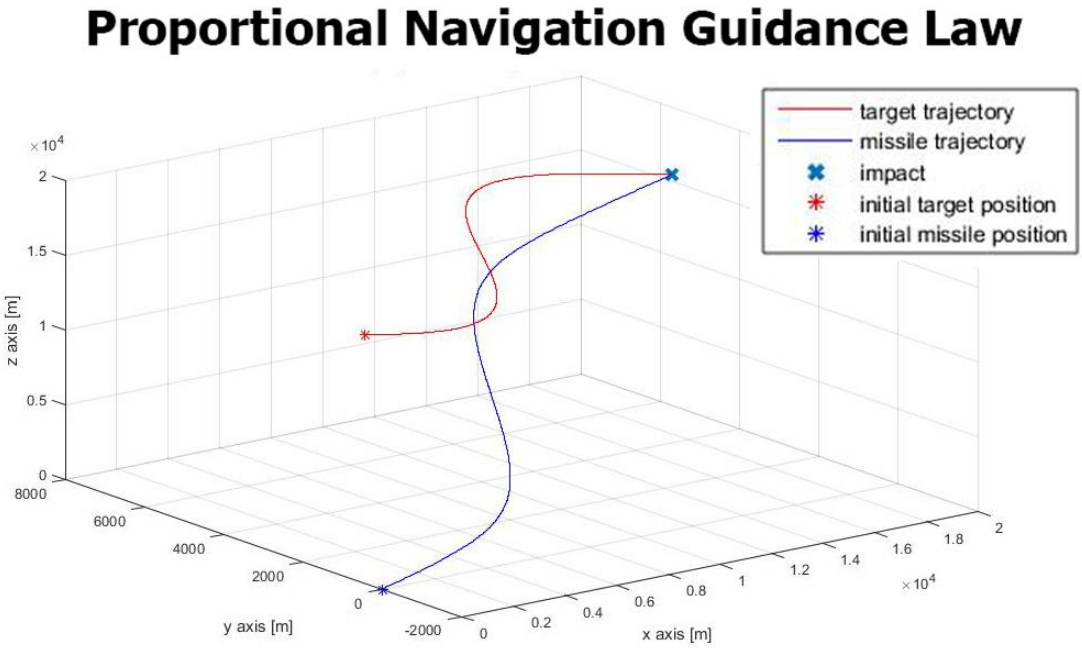

Image 17: Proportional Navigation, Case 2, trajectory plot

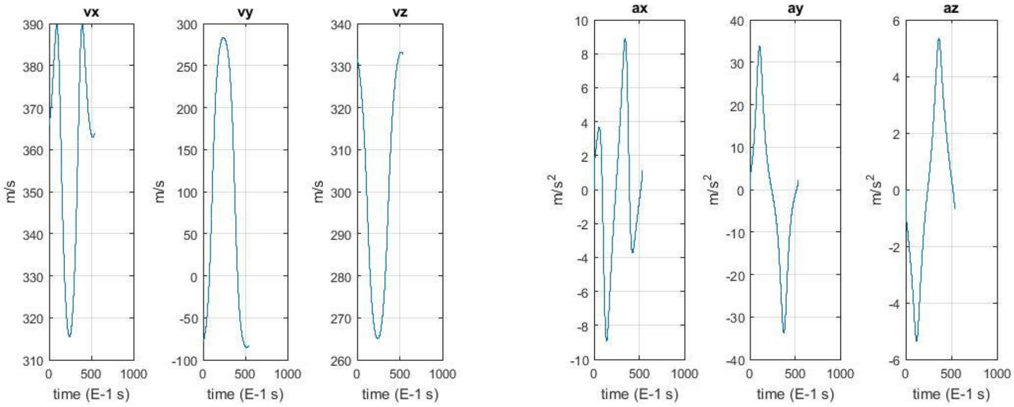

Image 18: Proportional Navigation, Case 2, velocity and acceleration plots

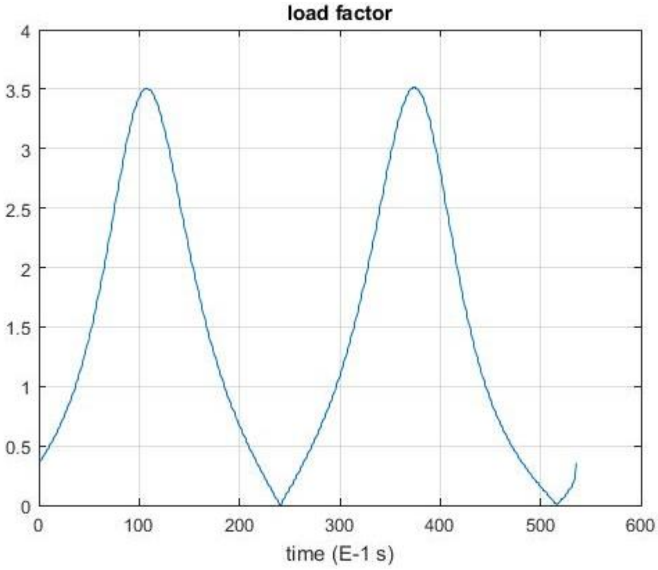

Image 19: Proportional Navigation, Case 2, load factor plot

Considering the trajectory plot as well as the velocity graphs, acceleration and load factor distribution, it is sees that missile is following also a repetitive path, in consequence, the load factor varies following a repetitive curve too. Moreover, the maximum load factor in Proportional Navigation is only 3.5129 versus the 4.5887 in Pure Pursuit. Therefore, comparing this two guidance laws, we confirm the second conclusion: Proportional Navigation is not as sensitive as Pure Pursuit with respect to target manoeuvre, which means that Proportional Navigation is more convenient when the target moves at high speed and changes its path.

In order to study the behaviour of the missile when applying Proportional Navigation guidance, it is needed to study the projections of the trajectories on the XY plane.

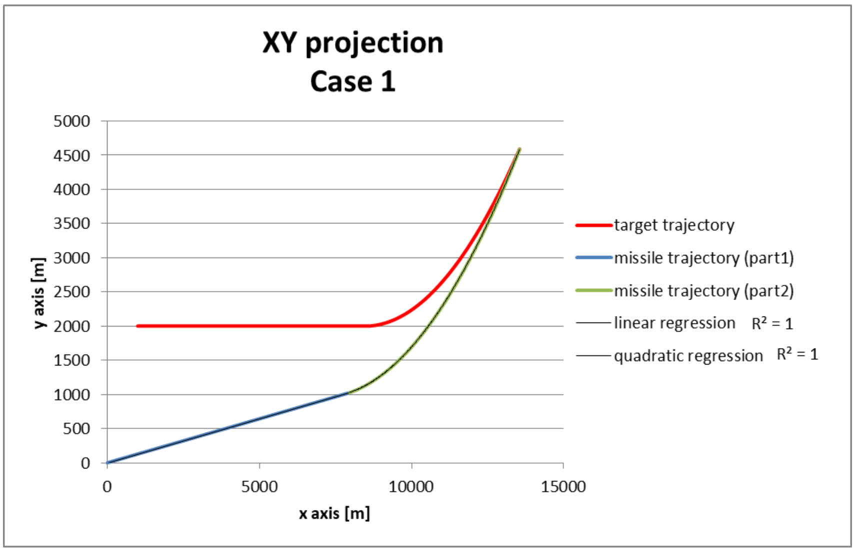

Image 20: Proportional Navigation, XY projection, Case 1

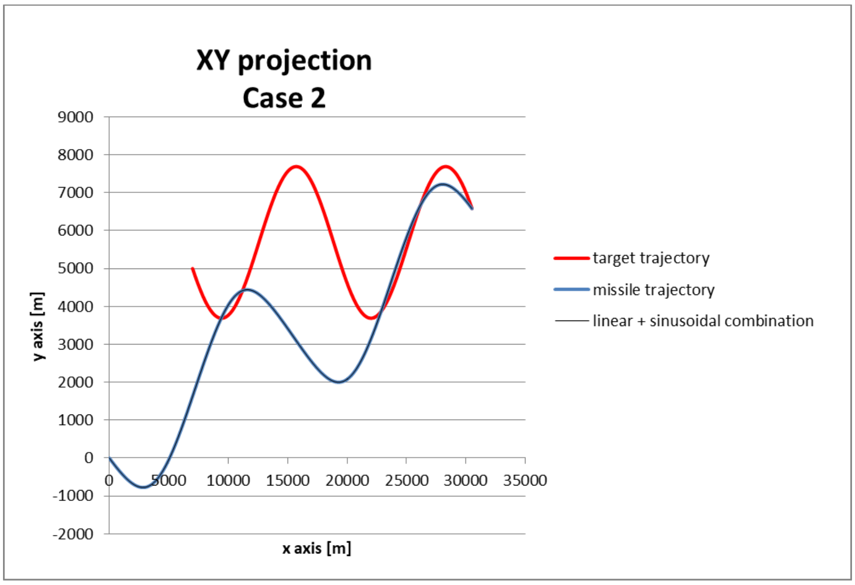

Image 21: Proportional Navigation, XY projection, Case 2

As it can be seen, in Case 1, when the target follows a straight path, the missile follows also a linear one, and when the target manoeuvres following a parabolic movement, the projection of the missile path coincides also with a quadratic function. With respect to case 2, as it was mentioned, the velocity vector of the target is $\overrightarrow{V_t}= vt \cdot \frac{\begin{pmatrix}1\\ m\\ 0.5\end{pmatrix}}{\sqrt{1+0.5^2+m^2}}$, where $m = \cos\left(\frac{xt}{2000}\right)$. Therefore, the movement coincides with the combination of a sinus function plus a linear trajectory. Analysing the projection of the missile trajectory on Case 2, it is shown that it is also coincident with a combination of a linear and sinusoidal movement. Therefore, we reach into a very important conclusion: The behaviour of the missile movement when applying Proportional Navigation Guidance, matches the one of the target, and this fact will have a very important impact on time for impact.

4.4. Line of Sight Guidance Law.

As it was exposed in section 2.2.2., in LOS guidance, a steering signal is sent to the missile by an external radar in such a way that the missile is forced to have the same angular rate as the target, being at all time in the line between the target and the point where the tracker radar is placed.

In order to implement this condition in the Matlab code we operate in the following way:

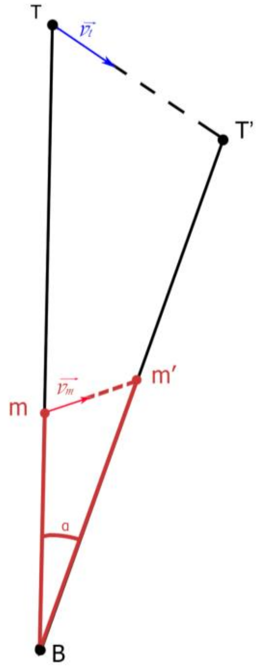

Image 22: LOS geometry scheme

m : missile position in time t.
m': missile position in time t+Δt s
T : target position in time t.
T': target position in time t+ Δt s.
B: base of the radar

We can define the following vectors:

$\overrightarrow{BT}$: base-target line of sight at time t.
$\overrightarrow{BT'} = \overrightarrow{vt} \cdot \Delta t + \overrightarrow{BT}$: base-target line of sight at time t+ Δt s
$\overrightarrow{Bm}$: base-missile line of sight at time t.
$\overrightarrow{Bm'}$: base-missile line of sight at time t+ Δt s.

$\overrightarrow{BT} \cdot \overrightarrow{BT'} = \left|\overrightarrow{BT}\right| \cdot \left|\overrightarrow{BT'}\right| \cdot \cos(\alpha)$, it is possible to free α from the equation

Then the condition of this guidance law, is that α, is the same for the missile and the target. Then in order to obtain $\left|\overrightarrow{Bm'}\right|$ it is necessary to apply the cosine theorem to the red triangle:

$$(\left|\overrightarrow{Vm}\right| \cdot \Delta t)^2 = \left|\overrightarrow{Bm}\right|^2 + \left|\overrightarrow{Bm'}\right|^2 - 2 \cdot \left|\overrightarrow{Bm}\right| \cdot \left|\overrightarrow{Bm'}\right| \cdot \cos(a)$$

This process must be applied iteratively, which means that for every Δt=0.1s the defined target vectors are updated, and so on α varies over time, In addition, it is needed, to update also the missile vectors, so after every computation of $\left|\overrightarrow{Bm'}\right|$, it is mandatory to upadate $\left|\overrightarrow{\mathrm{Bm}}\right|$ for the following iteration, which means that, in t+ Δt s, $\left|\overrightarrow{Bm}\right|_{t+\Delta t} = \left|\overrightarrow{Bm'}\right|_{t}$.

By applying this calculations iteratively over time, we obtain the calculations for this guidance law. As in previous guidance laws, we established the same trajectories of the target for each case.

4.4.1. Case 1: linear trajectory of the target with parabolic avoidance manoeuvre.

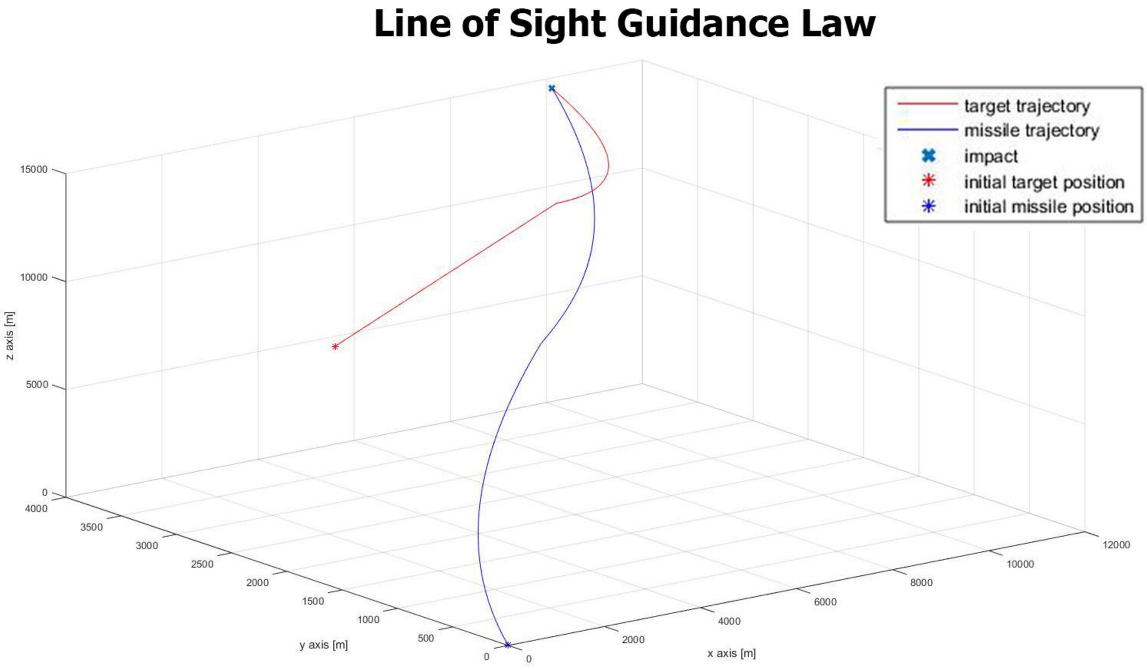

Image 23: Line of Sight, Case 1, trajectory plot

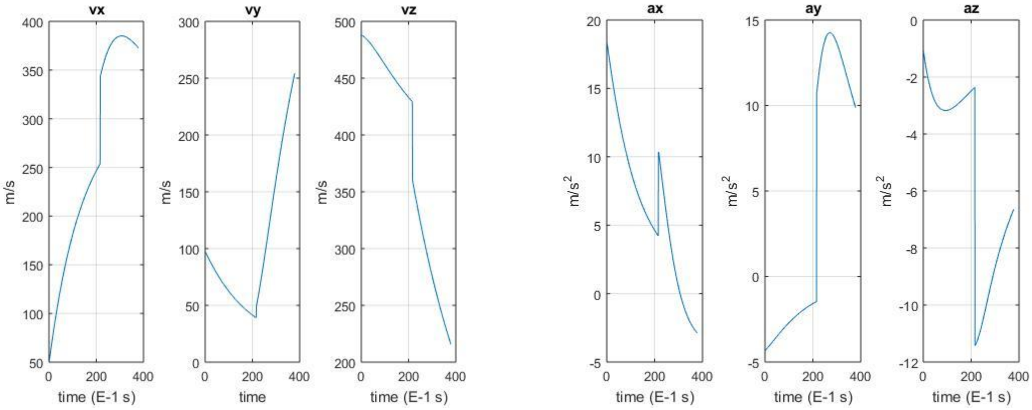

Image 24: Line of Sight, Case 1, velocity and acceleration plots

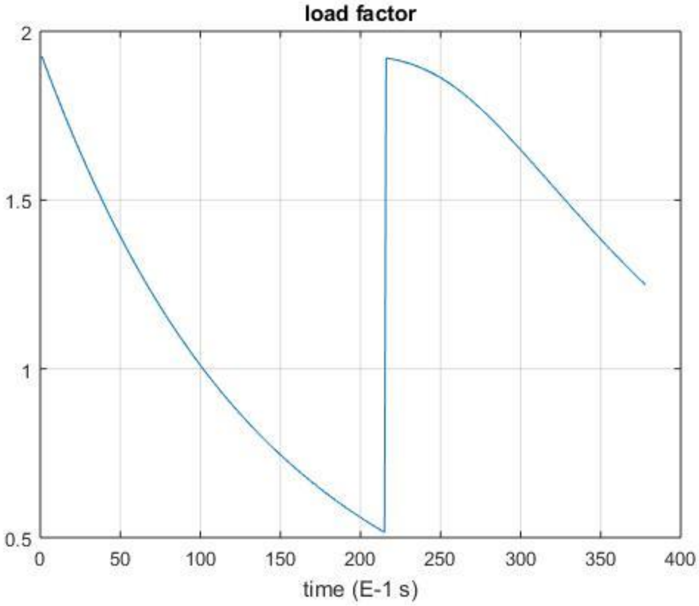

Image 25: Line of Sight, Case 1, load factor plot

In this case, it is seen the expected sudden change in load factor when the target manoeuvres since the missile needs to suddenly correct its path. The maximum load factor in this instant is 1.9257, similar to the maximum value reached when using Proportional Navigation (1.8133) rather than Pure Pursuit (3.6926), which allows to reach the third conclusion: Pure Pursuit is the guidance law that present worse results when manoeuvering. In addition, we reach to another point: although Line of Sight Guidance shows similar maximum load factor as Proportional Navigation, when the target follows a straight path, Proportional Navigation presents optimum results (better than Line of Sight results) since the missile follows the ideal collision path, with no normal accelerations.

4.4.2. Case 2: sinusoidal trajectory of the target.

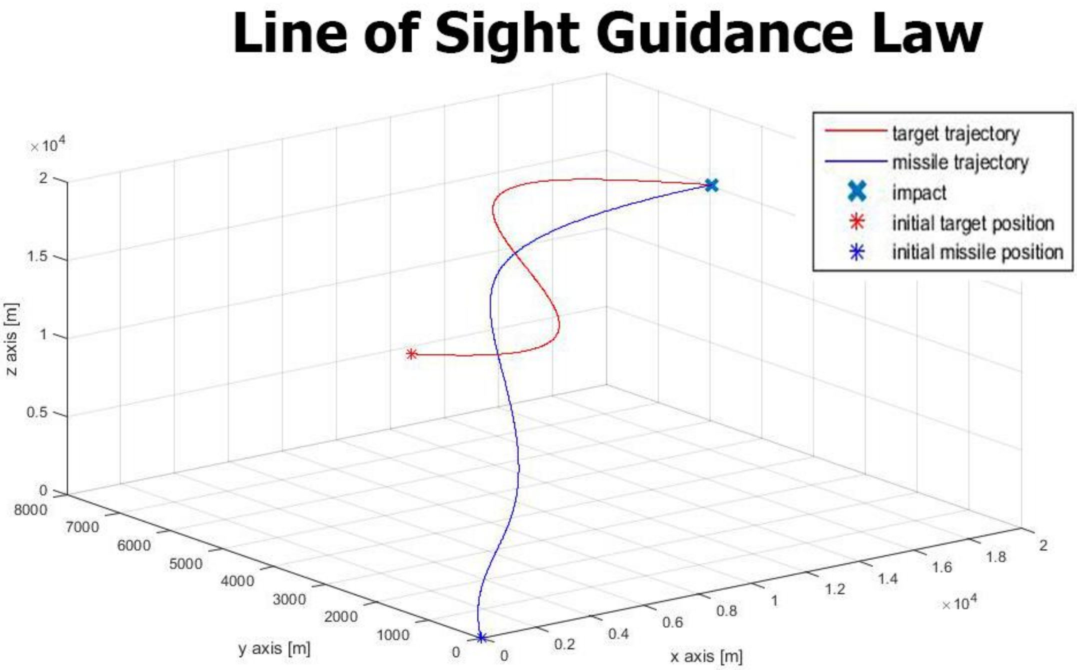

Image 26: Line of Sight, Case 2, trajectory plot

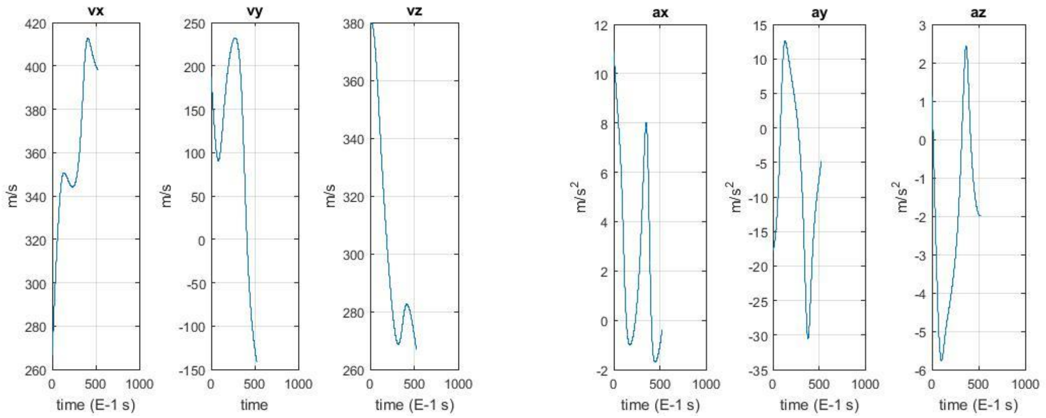

Image 27: Line of Sight, Case 2, velocity and acceleration plots

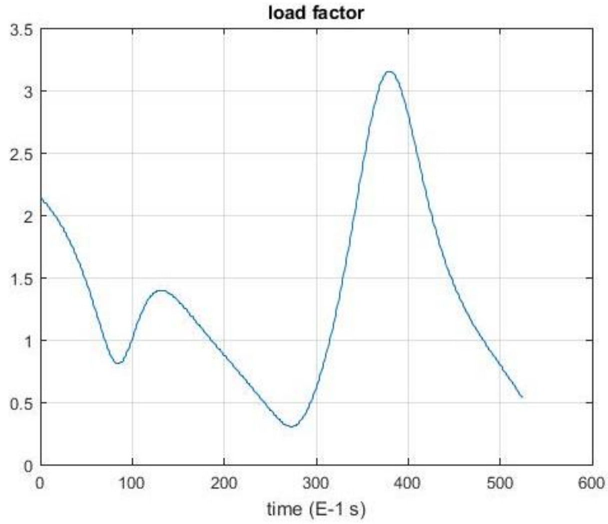

Image 28: Line of Sight, Case 2, load factor plot

Regarding the maximum load factor reached (3.1531), it is similar to the result in Case 2 of the Proportional Navigation Guidance (3.5129). So it is possible to conclude, that although the load factor distribution is completely different (as it was also seen in Case 1), the maximum load factors are similar between those two guidance laws. This evidence that for a guidance strategy, heading the missile towards other direction rather than the target itself, helps to reduce the load factor.

4.5. Effect of missile speed.

In this section, the aim is to study the impact of the missile speed on the trajectory performed by the missile in order to obtain a trend of how this parameter impacts the trajectory of the missile, as well as the load factor which the missile will cope with along the curve.

If we compare trajectories, where the target remains flying at vt=300 m/s, but the missile velocity is different, we see that obviously the impact takes longer when missile velocity is lower. However, in order to study properly each situation we need to know, how velocity would affect the load factor that the missile will face, in order to properly establish the design conditions.

4.5.1. Pure Pursuit Guidance Law

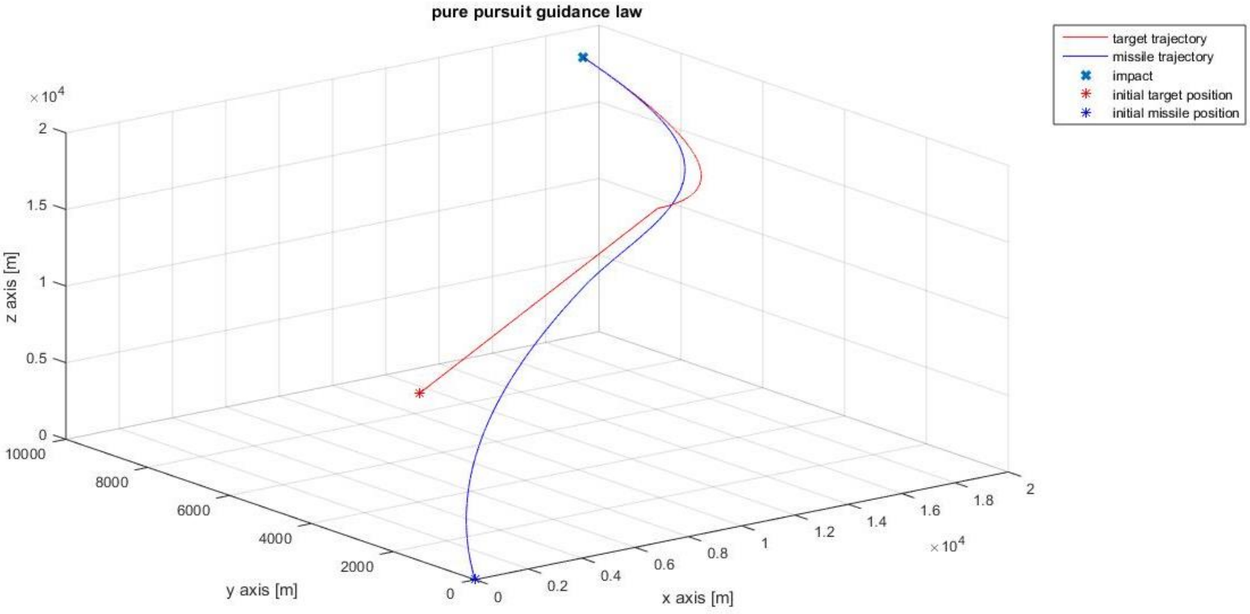

Image 29: Pure Pursuit, Case 1, trajectory plot, vm= 400 m/s

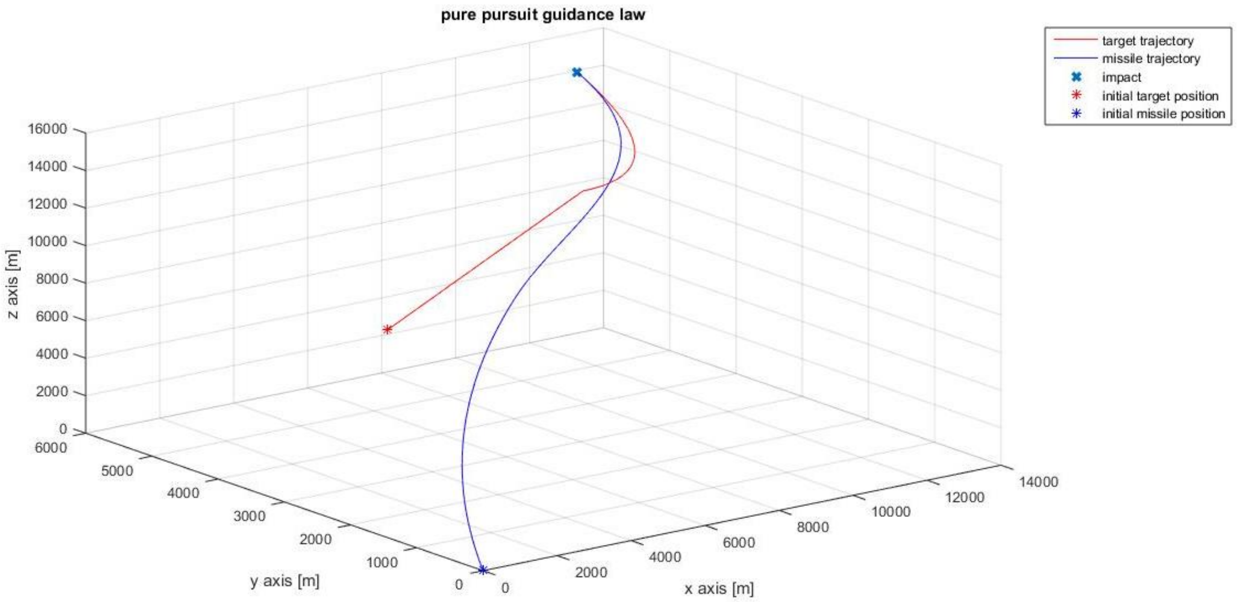

Image 30: Pure Pursuit, Case 1, trajectory plot, vm= 475 m/s

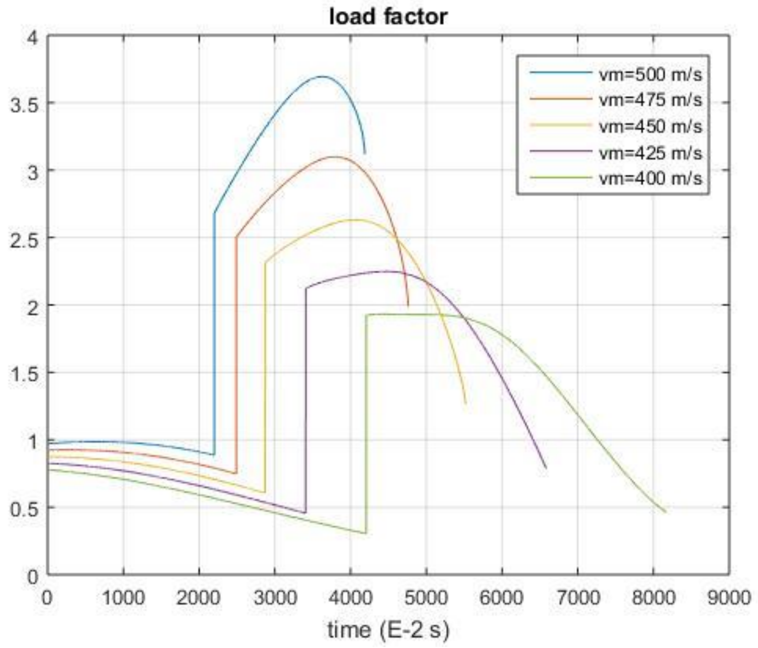

Image 31: Pure Pursuit, Case 1, load factor plot for different velocities

It is seen that if the only parameter changing is the missile's velocity, greater missile's speeds, will provide higher load factors in both linear and parabolic trajectories. In addition, when flying at lower velocities the peak in the quadratic path is flattened. It must be highlighted also that this guidance law is very sensitive to target's manoeuvre, since the difference between maximum load factor when the missile flies at 500 m/s and 400 m/s is 1.765. In order to properly understand the explanation of why missile velocity has such an impact on load factor, it is needed to see the projection of the trajectories prior change.

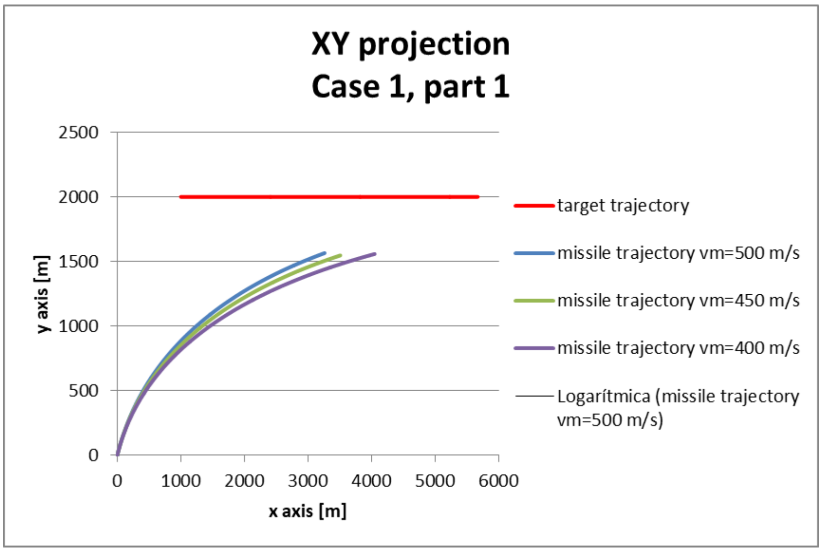

Image 32: Pure Pursuit, XY projection, Case 1, Part 1

As it can be seen in the trajectory plots, the missile tends to approach very fast the target, in such a way, that very soon follows a path very similar to the one of the missile and very closed to it. If the target manoeuvres the aerodynamic surfaces of the missile need to be deflected in such a way that allow to change the thrust direction so that the missile is capable to correct its path. Being closer to the target supposes that for the same time a bigger turn rate of the missile is needed, which supposes greater normal accelerations. The effect of the velocity would be that the highest the speed, the fastest the missile approaches target path, and therefore, a sudden manoeuvre supposes big load factors.

This phenomenon that has been seen experimentally, can be explained from a physical point of view: When moving at constant speed, lateral accelerations are the result of changes in the direction of the velocity vector. When the missile is far away from the target, a movement of the target requires a slight change in the direction of the missile velocity vector in order to correct the path, that would be a small angle of change in the direction. However, when the missile is closer to the target, a change of trajectory supposes that the missile, according to the guidance law, performs a greater correction in the flight direction, which means that the angle between those two consecutive vectors is greater than in the first case, and therefore greater lateral accelerations.

Nevertheless, Case 1 consist of only one manoeuvre, so that in order to properly study the effect of speed on load factor it is mandatory to study Case 2, since the sinusoidal path of the target makes the missile to continuously manoeuvre in order to impact.

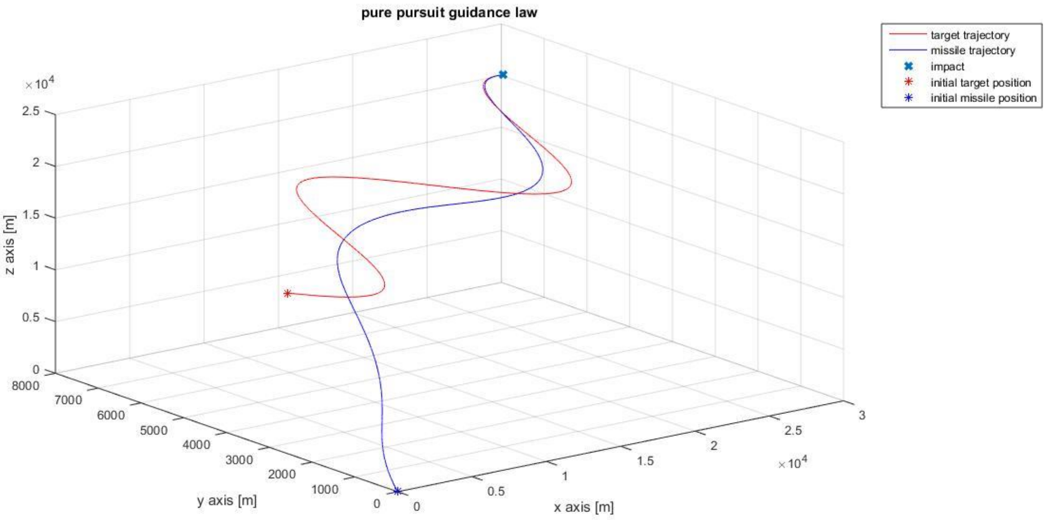

Image 33: Pure Pursuit, Case 2, trajectory plot, vm= 400 m/s

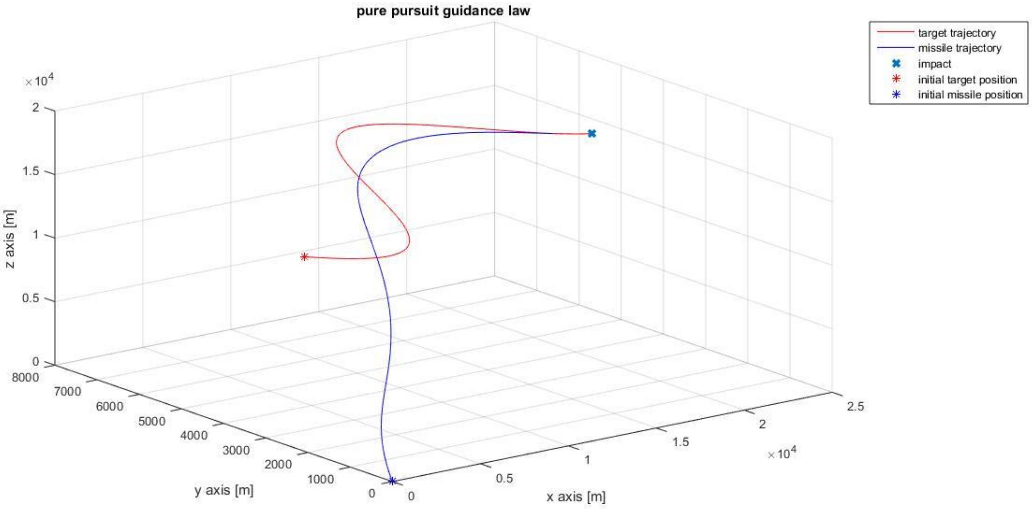

Image 34: Pure Pursuit, Case 2, trajectory plot, vm= 475 m/s

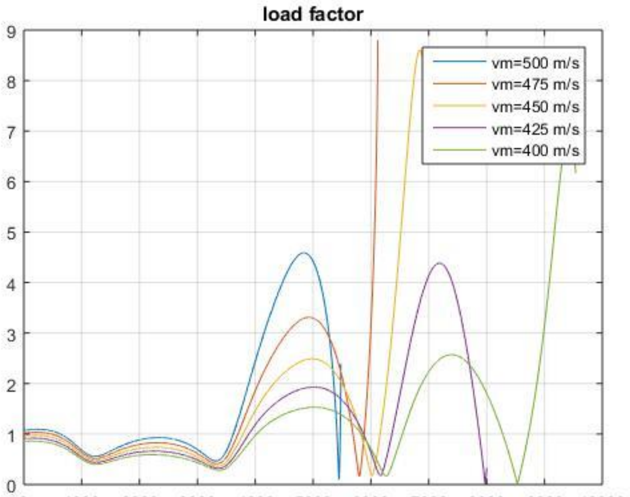

Image 35: Pure Pursuit, Case 2, load factor plot for different velocities

In order to understand the case, it is needed to analyse it from two parts. The first part is before the missile flying at 500 m/s impacts. In this part, what it is seen is the same trend as in Case 1: greater velocities give greater load factors, when manoeuvring. However, in Case 2, the target is performing a sinusoidal path, which could be interpreted as a multiple manoeuvre path, and so on, the missile has to manoeuvre several times too. Every manoeuvre supposes an increase in load factor, as we have already confirmed along the project. The missile flying at 500 m/s, only needs to manoeuvre three times, but when the missile flies at lower velocities more manoeuvres are needed. Furthermore, the amplitude of the peaks are raised, because every time the missile is closer to the target and therefore manoeuvering supposes a big change in the missile velocity vector direction, which means that the load factor tends to be increased. As it can be seen, Pure Pursuit Guidance Law is very sensitive to target's manoeuvre, and their effect on the load factor is difficult to predict, that is why lower velocities may present worse results as consequence of the multiple path changes required.

To sum up, for Pure Pursuit Guidance, lower velocities are more suitable when there are few manoeuvres of the target. However, as seen in case 2, when there are multiple manoeuvres greater velocities will require less number of path changes as the missile will impact before, and so on, maximum load factor will be smaller.

4.5.2. Proportional Navigation Guidance Law

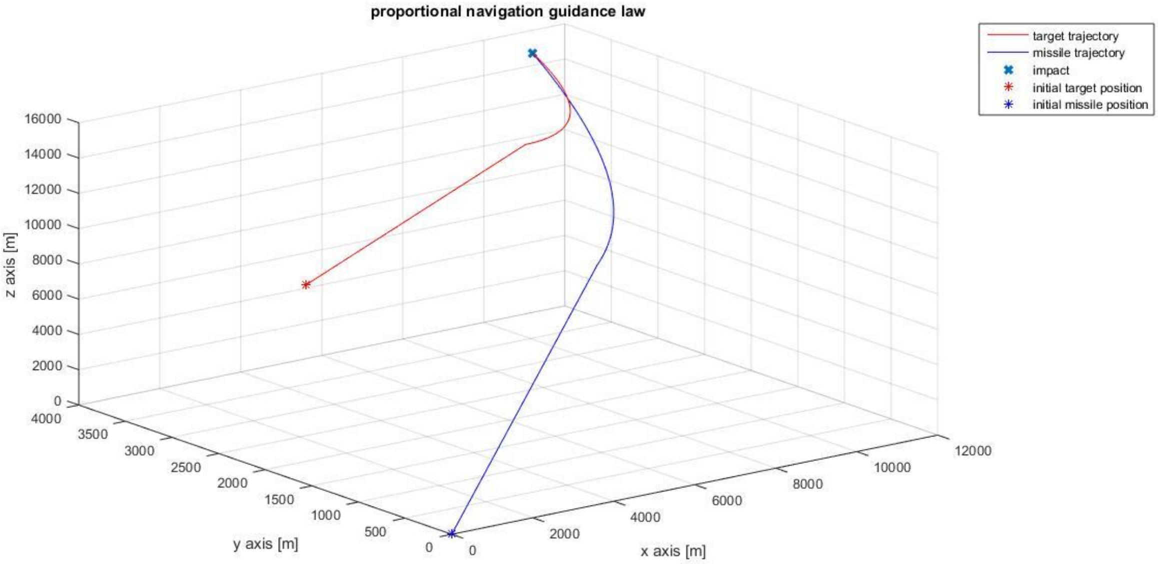

Image 36: Proportional Navigation, Case 1, trajectory plot, vm= 400 m/s

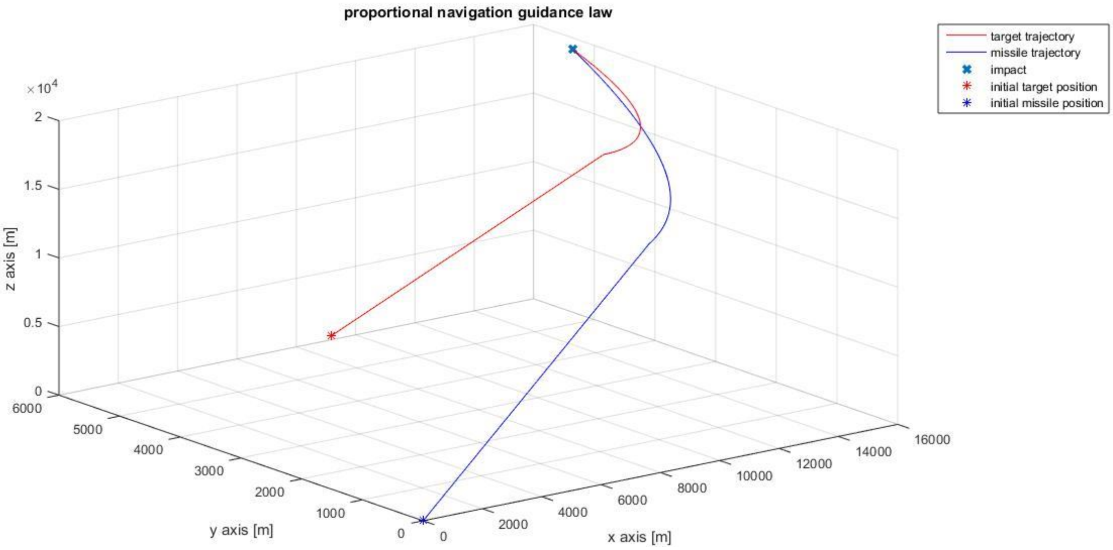

Image 37: Proportional Navigation, Case 1, trajectory plot, vm= 475 m/

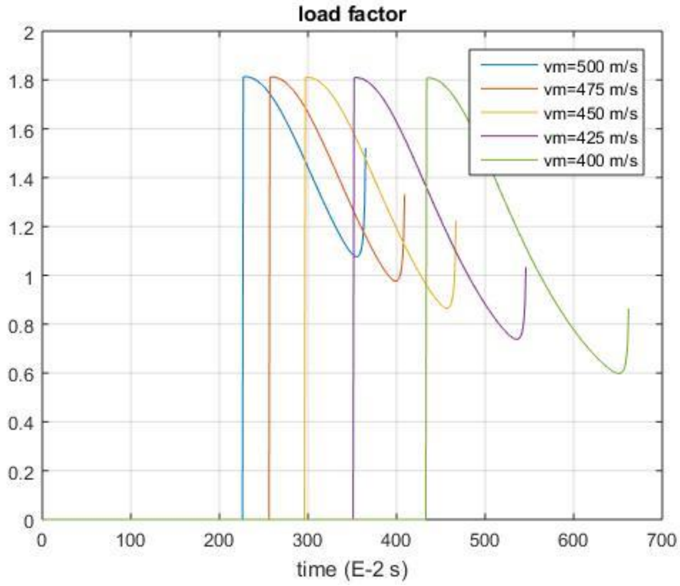

Image 38: Proportional Navigation, Case 1, load factor plot for different velocities

In this case, changes of missile will practically not change the maximum load factor reached. It is seen, that on impact there is a sudden increase of it, however we are not focused on this as the normal accelerations reached are not critical and in this instant the impact takes place, and therefore load factor in is not determinant. This property is very important from a design point of view, as it is possible to predict the maximum load factor independently of the missile fight velocity.

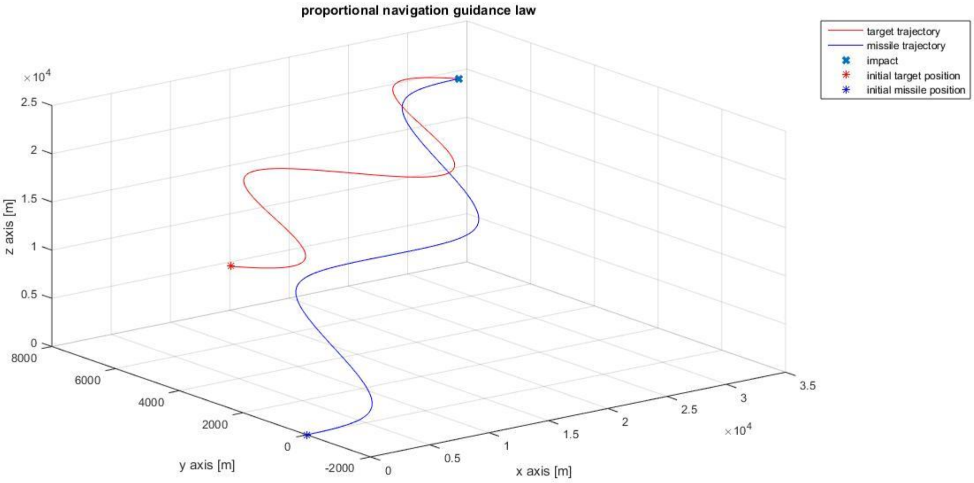

Image 39: Proportional Navigation, Case 2, trajectory plot, vm= 400 m/s

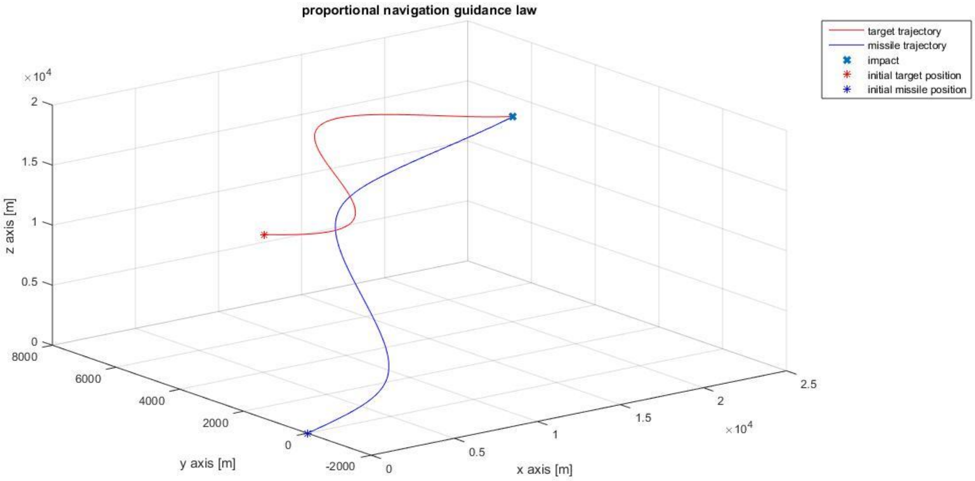

Image 40: Proportional Navigation, Case 2, trajectory plot, vm= 475 m/s

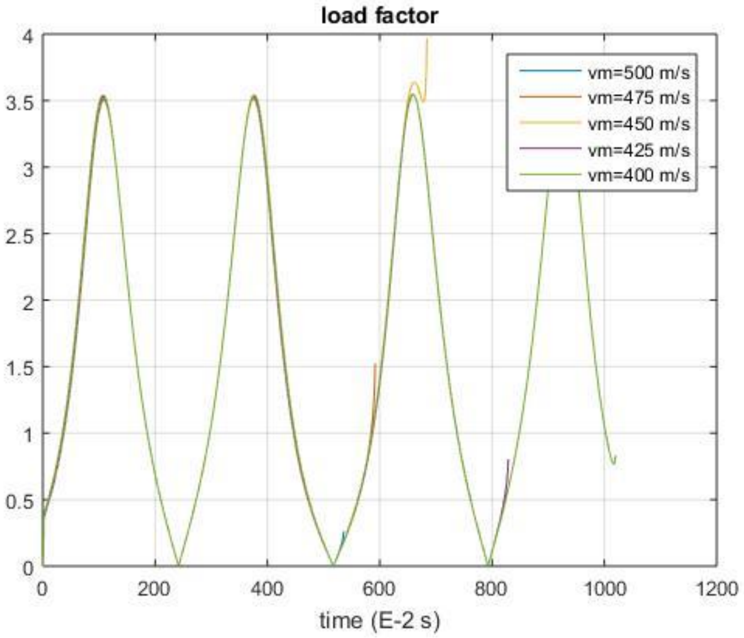

Image 41: Proportional Navigation, Case 2, load factor plot for different velocities

It is seen, that in Case 2, the missile is following an approximate sinusoidal trajectory towards its target, and since this guidance law is not as sensitive as Pure Pursuit regarding target manoeuvre, peak values are nearly not increased when manoeuvering. Therefore, the load factor distribution is practically coincident with the exception of the sudden change when impact.

To conclude, as the maximum load factor experienced in Proportional Navigation is practically equal no matter missile's velocity, which means that velocity of the missile has very little impact on it, it would be more suitable to fly at high speeds to collide with the target as soon as possible.

4.5.3. Line of Sight Guidance Law.

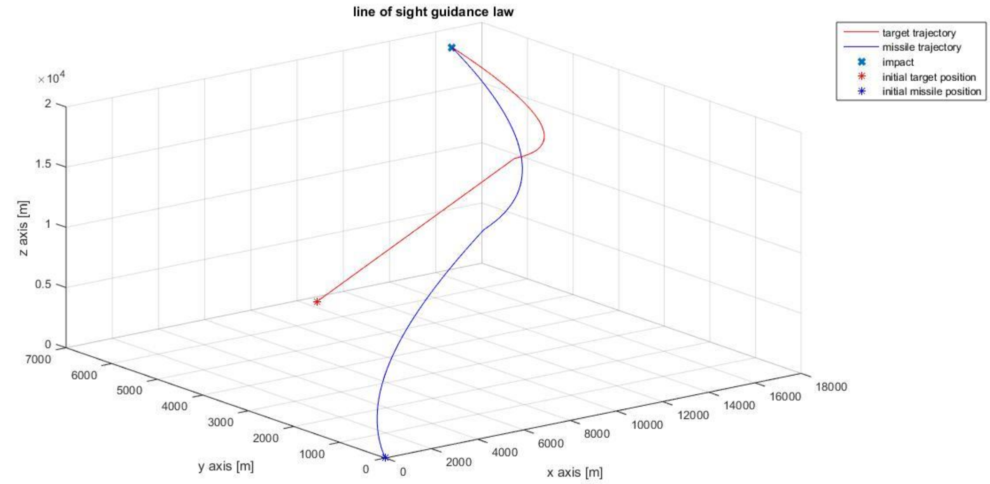

Image 42: Line of Sight, Case 1, trajectory plot, vm= 400 m/s

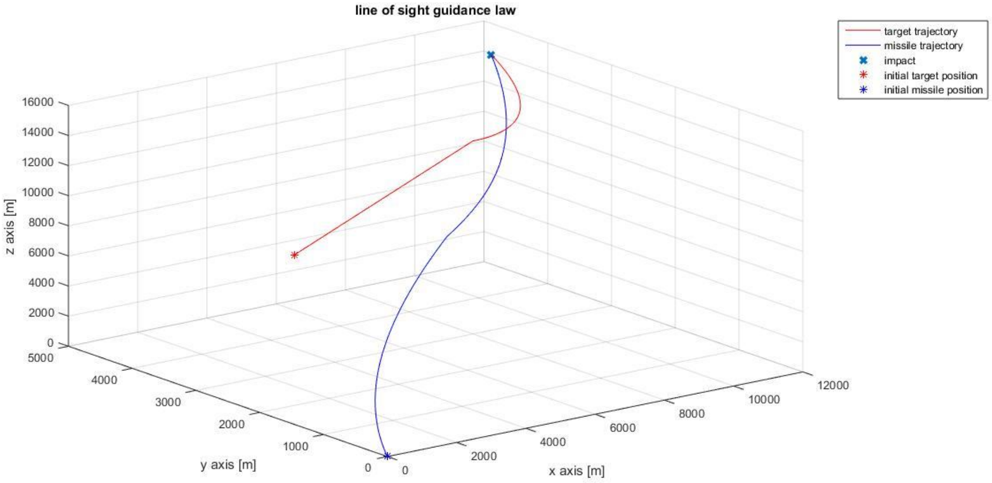

Image 43: Line of Sight, Case 1, trajectory plot, vm= 475 m/s

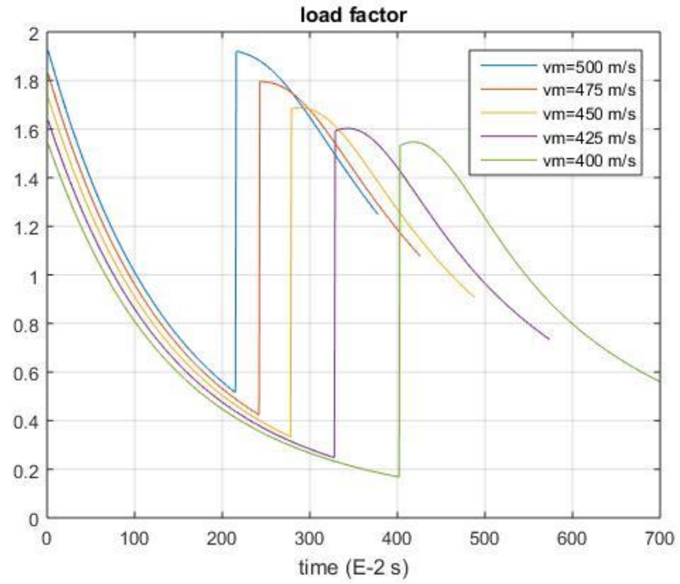

Image 44: Line of Sight, Case 1, load factor plot for different velocities

It is seen that greater velocities, provide higher load factors and when flying at lower velocities the peak in the quadratic path is flattened: same trend as in Pure Pursuit. Nevertheless, it is noticed that the main difference with this aforementioned law is that Line of Sight Guidance is not as sensitive regarding missile velocity as the difference in maximum load factor between the missile flying at 500 m/s and 400 m/s is only around 0.4, while in Pure Pursuit this difference was around 1.70.

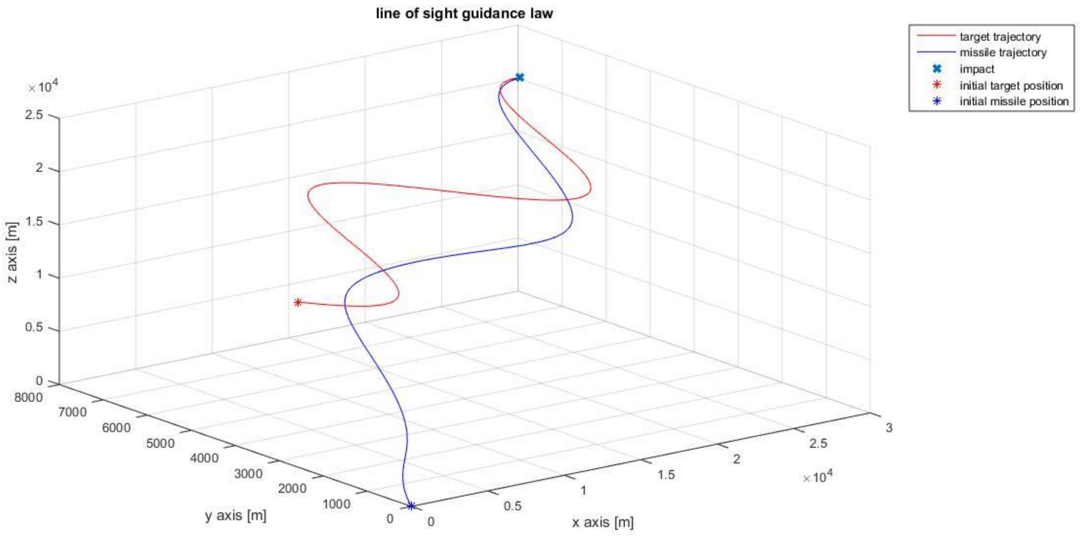

Image 45: Line of Sight, Case 2, trajectory plot, vm= 400 m/s

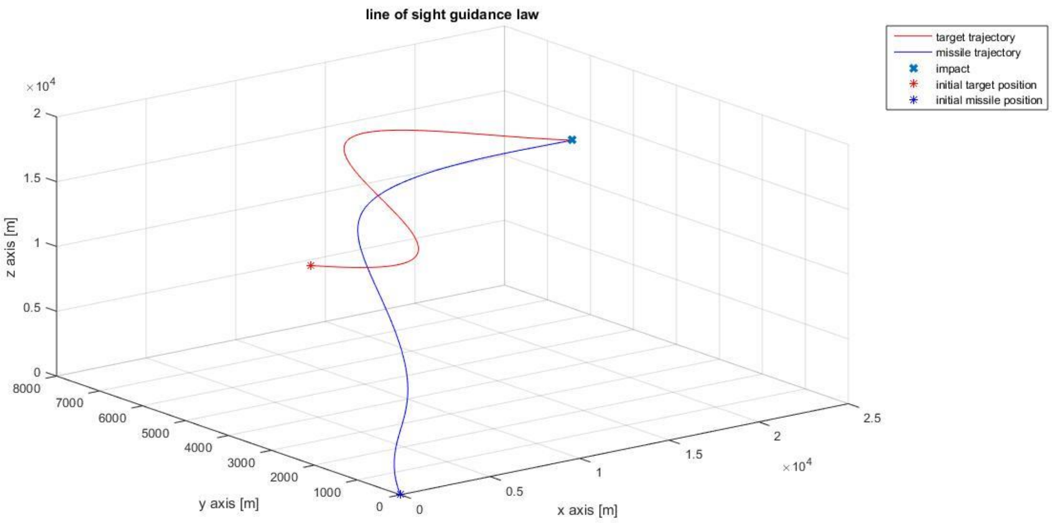

Image 46: Line of Sight, Case 2, trajectory plot, vm= 475 m/s

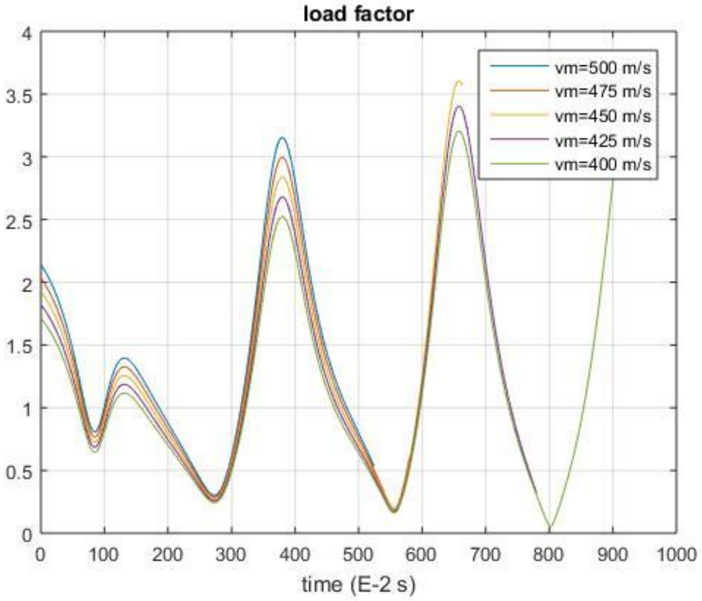

Image 47: Line of Sight, Case 2, load factor plot for different velocities

As it was mentioned, Line of Sight Guidance is sensitive to manoeuvre changes, therefore in the first part, where the missile flying at 500 m/s has not impacted already, we see the same trend as in Pure Pursuit: greater velocities provide higher load factors. However, lower velocities require more manoeuvres, and the closer to the objective, the greater the lateral acceleration peaks, which lead to a load factor increase. However, analysing Case 1 and 2, it can be concluded that Line of Sight Guidance presents lower load factors than Pure Pursuit and is not as sensitive to target's manoeuvering, as consequence of implementing a more advanced pursuit strategy rather than just point in all moment the target, which makes that the missile has to react to target's manoeuvres with a high turn rate.

4.6. Wind gusts effects.

Up to this point, we have been working in an environment where no wind gusts have been taken into account. However, since the environment in which missile works is on Earth, atmospheric phenomena must be taken into account.

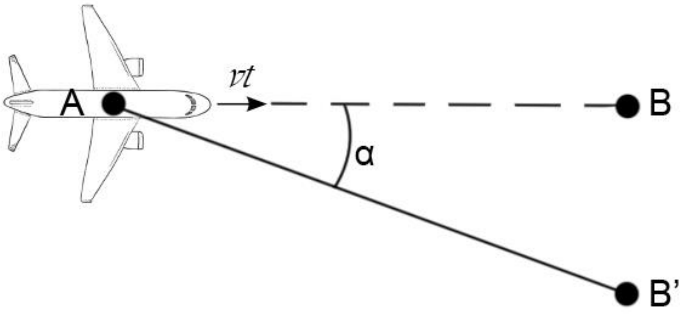

Image 48: Wind drift effect

When a body is flying from one point (A) to another (B), in a region where a crosswind acts on the body, if no correction is performed, a wind drift takes place deviating the body an angle (α) from the desired path. In the end, if there is no trajectory correction by the pilot/autopilot the final point (B') is deviated from the intended point (B). Therefore, in order to compute the effect of the wind, the final velocity vector has to be calculated as

$$\vec{v} = \vec{v}_i + \vec{v}_{wind}$$

Where "i" stands for the subdindex "m" for the missile and "t" for the target. Therefore, trajectories would vary with respect to the previous ones, depending on the modelled wind. For representing the effect of it, $\vec{v}_{wind}$ has been considered to be $\vec{v}_{wind}$ =(0,15,0) m/s.

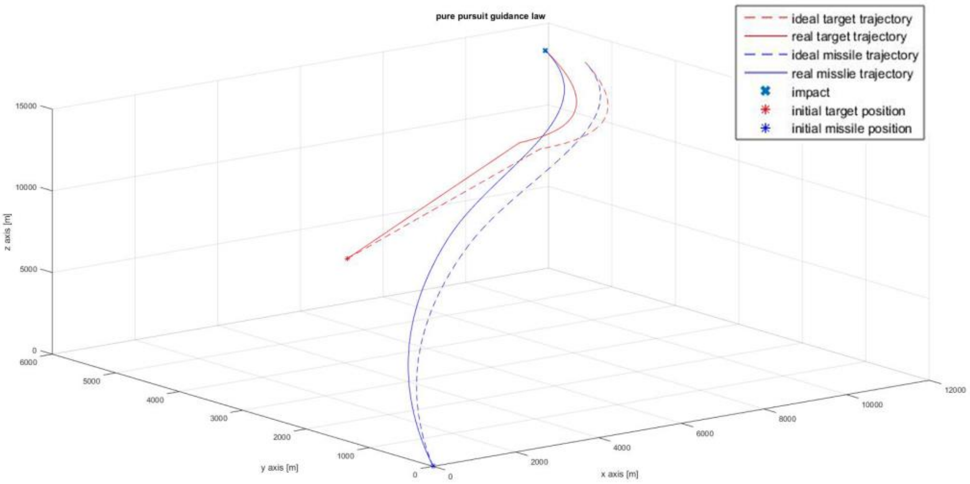

Image 49: Case 1, wind drift effect example

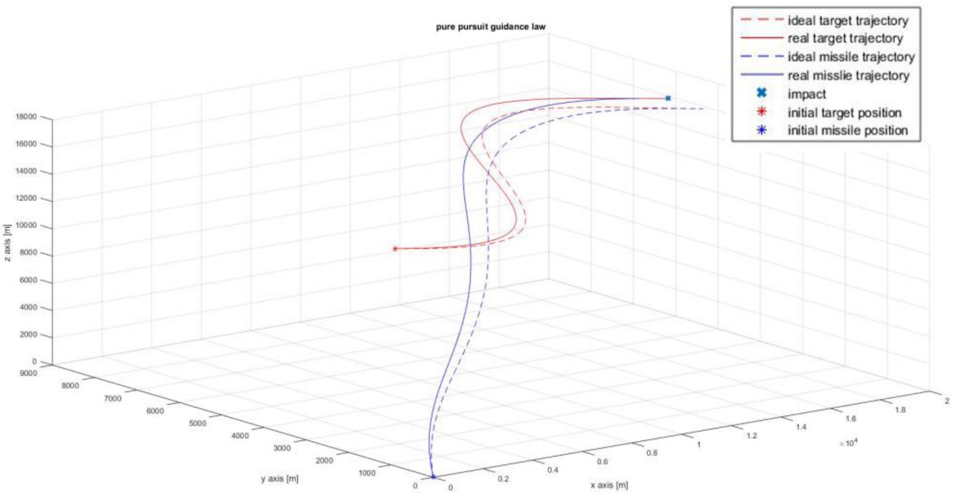

Image 50: Case 2, wind drift effect example

It is seen that progressively, the current position of both missile and target differs more from the one where no wind was considered. In addition, it is also needed to estimate, how the wind can influence the load factor:

$$a_x \approx \frac{\Delta V_x}{\Delta t} = \frac{\Delta V_x}{t_2 - t_1} = \frac{\left(v_{m_{x2}} + v_{wind_{x2}}\right) - \left(v_{m_{x1}} + v_{wind_{x1}}\right)}{t_2 - t_1}$$

$$a_y \approx \frac{\Delta V_y}{\Delta t} = \frac{\Delta V_y}{t_2 - t_1} = \frac{\left(v_{m_{y2}} + v_{wind_{y2}}\right) - \left(v_{m_{y1}} + v_{wind_{y1}}\right)}{t_2 - t_1}$$

$$a_z \approx \frac{\Delta V_z}{\Delta t} = \frac{\Delta V_z}{t_2 - t_1} = \frac{\left(v_{m_{z2}} + v_{wind_{z2}}\right) - \left(v_{m_{z1}} + v_{wind_{z1}}\right)}{t_2 - t_1}$$

Therefore, there are the following cases:

- If $v_{wind_{i2}} = v_{wind_{i1}}$ (where sub index “i” stands for x, y and z directions), which means that the wind velocity is constant, there is a deviation from the trajectory, but not a change in load factor with respect to the ideal case where no wind was being taken into account.
- If $v_{wind_{i2}} > v_{wind_{i1}}$, which means that there is a sudden wind gust, load factor will always increase as consequence of the sudden increase in normal acceleration.
- If $v_{wind_{i2}} < v_{wind_{i1}}$, which means that the sudden wind gust is on instant t_2 lower, load factor will decrease, however, it will still be greater than the ideal case with no wind.

4.7. Results comparison.

In this section, the objective is to compare results from the previous modelled guidance laws (time to impact and maximum load factor), so that it is possible to reach conclusions.

4.7.1. Missile velocity 500 m/s

	Target movement	Time to impact (seconds)	Maximum load factor
Pure Pursuit Guidance Law			
Case 1	Linear	-	0.9858
	Parabolic	41.90	3.6926
Case 2	Sinusoidal	54.81	4.5887
Proportional Navigation Guidance Law			
Case 1	Linear	-	0
	Parabolic	36.70	1.8133
Case 2	Sinusoidal	53.90	3.5129
Line of Sight Guidance Law			
Case 1	Linear	-	1.9257
	Parabolic	38.10	1.9201
Case 2	Sinusoidal	52.70	3.1531

Table 1: Results for vm=500 m/s

4.7.2. Missile velocity 475 m/s

	Target movement	Time to impact (seconds)	Maximum load factor
Pure Pursuit Guidance Law			
Case 1	Linear	-	0.9255
	Parabolic	47.62	3.0983
Case 2	Sinusoidal	61.21	8.8037
Proportional Navigation Guidance Law			
Case 1	Linear	-	0
	Parabolic	41.20	1.8121
Case 2	Sinusoidal	59.50	3.5151
Line of Sight Guidance Law			
Case 1	Linear	-	1.8294
	Parabolic	42.90	1.7960
Case 2	Sinusoidal	58.30	2.9954

Table 2: Results for vm=475 m/s

4.7.3. Missile velocity 450 m/s

	Target movement	Time to impact (seconds)	Maximum load factor
Pure Pursuit Guidance Law			
Case 1	Linear	-	0.8745
	Parabolic	55.23	2.6310
Case 2	Sinusoidal	69.50	8.6034
Proportional Navigation Guidance Law			
Case 1	Linear	-	0
	Parabolic	46.90	1.8110
Case 2	Sinusoidal	68.70	3,5972*
Line of Sight Guidance Law			
Case 1	Linear	-	1.7331
	Parabolic	49.10	1.6876
Case 2	Sinusoidal	66.60	3.6048

Table 3: Results for vm=450 m/s

*peak value at impact no considered, as when impact, load factor is not determinant

4.7.4. Missile velocity 425 m/s

	Target movement	Time to impact (seconds)	Maximum load factor
Pure Pursuit Guidance Law			
Case 1	Linear	-	0.8259
	Parabolic	65.85	2.2479
Case 2	Sinusoidal	80.06	4.3845
Proportional Navigation Guidance Law			
Case 1	Linear	-	0
	Parabolic	54.80	1.8085
Case 2	Sinusoidal	83.20	3.5434
Line of Sight Guidance Law			
Case 1	Linear	-	1.6368
	Parabolic	57.70	1.6030
Case 2	Sinusoidal	78.30	3.4045

Table 4: Results for vm=425 m/s

4.7.5. Missile velocity 400 m/s

	Target movement	Time to impact (seconds)	Maximum load factor
Pure Pursuit Guidance Law			
Case 1	Linear	-	0.7773
	Parabolic	81.74	1.9312
Case 2	Sinusoidal	95.42	6.9927
Proportional Navigation Guidance Law			
Case 1	Linear	-	0
	Parabolic	66.40	1.8085
Case 2	Sinusoidal	102.30	3.5433
Line of Sight Guidance Law			
Case 1	Linear	-	1.5405
	Parabolic	70.20	1.5469
Case 2	Sinusoidal	95.50	3.6417

Table 5: Results for vm=400 m/s

4.8. Results Discussion

Along the project, the behaviour of the classical guidance laws studied has been shown. All the graphs and charts as well as the previous tables allow to reach the following conclusions:

In Pure Pursuit Guidance, the guidance strategy is very easy, since the missile is always pointing towards its target, in such a way that the missile is obliged to head towards the target continuously.

On the one hand, we see that in Case 1, the high load factors jumps are found on the trajectory when the target manoeuvres changing from a linear trajectory to a parabolic one, which is a clear indication that this guidance law is sensitive to manoeuvre changes of the target. Moreover, analysing this case at different velocities: 500 m/s, 475 m/s, 450 m/s, 425 m/s, 400 m/s. the following maximum loads factors are obtained respectively: 3.6926, 3.0983, 2.6310, 2.2479, 1.9312. It is clear that a change in velocity magnitude of the missile has a big impact on the results, as the difference between flying at 500 m/s and 400 m/s means a difference around 1.70 in maximum load factor.

On the other, when analysing Case 2 maximum load factors (4.5887, 8.8037, 8.6034, 4.3845, 6.9927) do not follow the previous trend. This is explained because, as it was seen on Image 33, when flying at lower velocities the missile needs to manoeuvre more times, and therefore maximum peaks in load factor are found in different manoeuvres in each of the cases. It is obvious that, for the same manoeuvre, lower velocities will carry lower load factors, as it was shown in Case 1, however, since flying at 400 m/s, requires more manoeuvres than flying at 500 m/s, in one of those extra manoeuvres, due to the higher proximity of the missile to the target, the maximum load factor reached is greater. To sum up, although velocity influences the results, the main factor that influences this guidance law is the amount of manoeuvres that the missile has to perform in order to impact. In addition, the problem with Pure Pursuit Guidance, that avoids it from being used to collide with fast movable targets, is that it approaches very fast to the target trajectory, therefore manoeuvering can be translated into very high lateral accelerations. That is why this guidance is used in easy cases where the target remains fixed or moves slowly.

With respect to Proportional Navigation guidance, it is also a simple mechanized strategy, therefore it is widely used. As it was explained, the objective in this guidance law is to approximate the real trajectory to the instantaneous ideal collision path, which is a linear trajectory, therefore when implementing this guidance law, a calculator is required in order to compute the instantaneous point of impact.

In Case 1, it is noticed that when the target follows the linear trajectory, the missile follows also a straight path, and therefore in all cases load factor is 0, since the real trajectory corresponds with the ideal collision path, where the missile suffers no lateral accelerations. It must be highlighted, that this guidance law is not as sensitive to target manoeuvres (changes from linear to parabolic path) as Pure Pursuit, since when the missile flies at 500 m/s, in the manoeuvre instant maximum load factor in Pure Pursuit was 3.6926 while in Proportional Navigation is 1.8133, almost a reduction of 50%.

In Case 2, it is also evident that reduction since in Pure Pursuit the missile would bear a 4.5887 times gravity while in Proportional Navigation 3.5129. It is noticed also, that in Proportional Navigation velocity practically does not influence the load factor (with the exception of the sudden increase when impact, which can be neglected as in that instant collision takes places and therefore the missile does not need to fly anymore).The maximum load factors experienced in Case 1 (1.8133, 1.8121, 1.8110, 1.8085, 1.8085) and Case 2 (3.5129, 3.5151, 3,5972, 3.5434, 3.5433) show that there is no significant change to consider that velocity has an impact, which is positive from a design point of view.

If we consider Case 2, for missile velocity of 500 m/s and 400 m/s, we find that the two first peaks of the load factor are found at 11,0 and 37,6 seconds. If we analyse the angle between the two consecutive velocity vectors when peak value is reached we find that at v_m=500 m/s the angle values are 0,3939º and 0,3949º respectively, and for v_m=400 m/s, 0,4973º and 0,4977º. So although when using Proportional Navigation Guidance, the same impact of missile velocity change applies, the variation of angles is so small that is possible to neglect the impact of velocity on load factor. Moreover, with respect to the sudden increase on the final curse of the trajectory, this is normally not determinant because the increase is not very dramatic. In a nutshell, it could be concluded that this guidance law is very suitable for fast movable targets.

Regarding Line of Sight law, LOS guidance is the guidance technique used; therefore the missile would need an external source to provide the target position to the missile. Analysing the results it is noticed the same trend as in Pure Pursuit: when the missile has to manoeuvre to adapt the new trajectory of the target there is a sudden load factor increase. In addition, lower velocities carry lower load factors, unless there are multiple target manoeuvres, then, the missile which is flying at lower velocity takes more time to impact that the one who flies at higher velocities, therefore, it has to perform more manoeuvres and so on there are new peaks in the load factor, which means that the slower missile can bear in the end greater lateral accelerations. The main difference, is that this guidance law is less sensitive to the aforementioned parameters than Pure Pursuit.

In Case 1, when flying at 500 m/s, the missile bears a maximum load factor of 1.9201 when manoeuvering, which is a reduction around the 50% with respect to the maximum value in Pure Pursuit. Furthermore, this result is very similar to the one of Proportional Navigation (1.8133), and therefore when flying at lower velocities, the maximum load factor is even lower than the one presented by Proportional Navigation, since in Line of Sight law the maximum load factor is lowered with velocity reduction while in Proportional Navigation it remained constant.

In Case 2, however, Proportional Navigation loads factor, (3.5129, 3.5151, 3,5972, 3.5434, 3.5433) thanks to remaining constant the load factor independently the flight velocity and manoeuvres performed, presents in some cases better results than Line of Sight Guidance (3.1531, 2.9954, 3.6048, 3.4045, 3.6417). In addition with respect to those of Pure Pursuit (4.5887, 8.8037, 8.6034, 4.3845, 6.9927), the results are extremely improved, as a consequence of heading the missile towards a different point rather than trying to keep it fix to the line of sight missile-target. Therefore, Line of Sight Guidance, is widely used when the target moves fast or performs multiple manoeuvres.

Another important aspect, is the time it takes the missile to collide with its target, especially when it is the case of a movable target. Taking into account the results, for Case 1 and 2, for the different missile velocities (500 m/s, 475 m/s, 450 m/s , 425 m/s, 400 m/s) in the three guidance laws, results can be plotted:

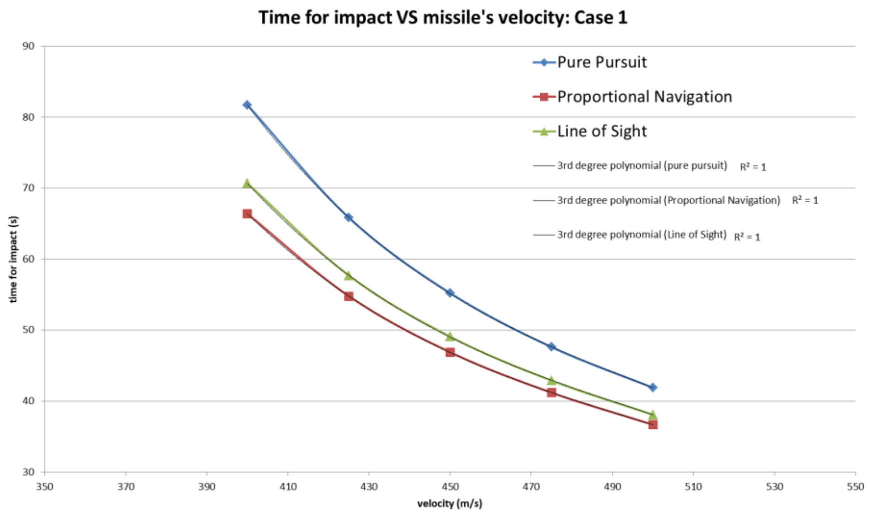

Image 51: Time for impact vs missile's velocity: Case 1

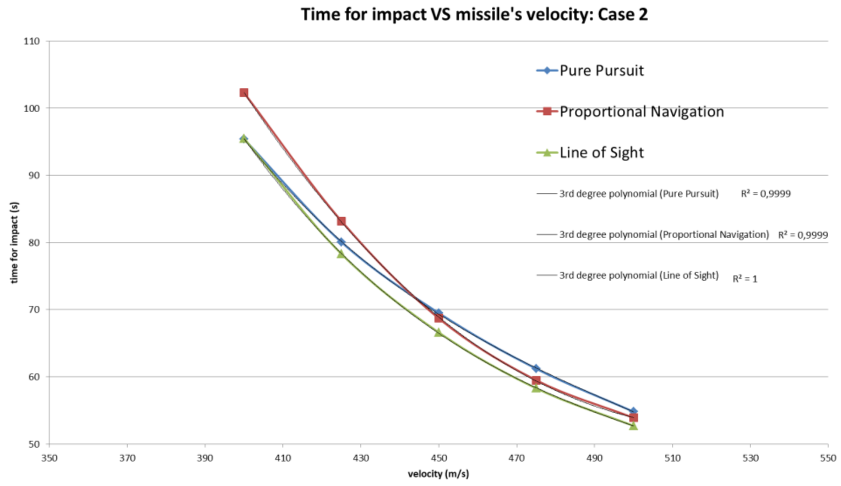

Image 52: Time for impact vs missile's velocity: Case 2

The first point to highlight, is that the flight time can be modelled with accuracy as a third degree polynomial where the independent variable would be the missile velocity, and the equation would be upper bounded by the maximum flight speed that the missile is able to reach and lower bounded by the minimum flight speed in order to impact (the condition that the lower bound must satisfy is $v_m >> v_t$).

Regarding which guidance law is optimum from the consideration of impact time, it should be mentioned that it is not possible to obtain an absolute criteria, like when comparing with respect to the load factor. When flying at constant speed, flight time will be a matter of the travelled distance. As it was studied on section 4.3.2. The behaviour of the missile's trajectory when applying Proportional Navigation guidance, matches the movement of the target. Therefore, on a linear target's path, Proportional Navigation will always take lower time for impact since the missile will follow a linear path too, while applying the other two guidance laws the trajectories are curvilinear. Nevertheless, when the target trajectory is not straight but curvilinear (for example Case 2), the same trend does not apply. In order to understand why so, we focuse on the XY projection of the different trajectories.

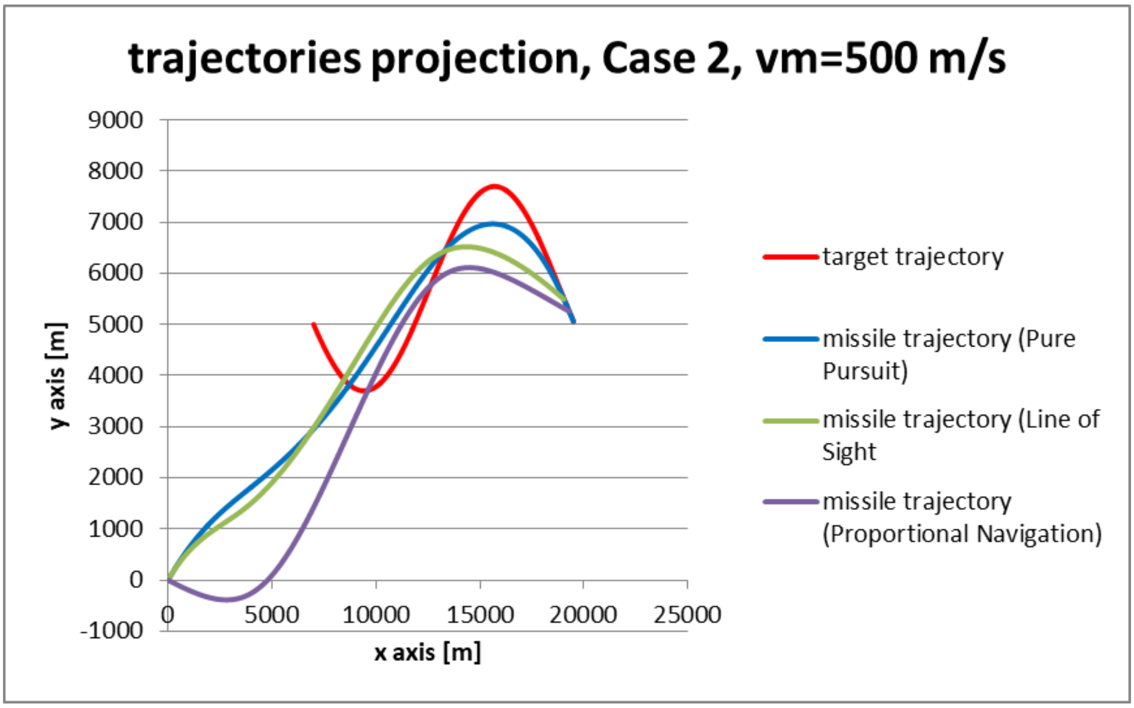

Image 53: XY trajectories projection, Case 2, vm= 500 m/s

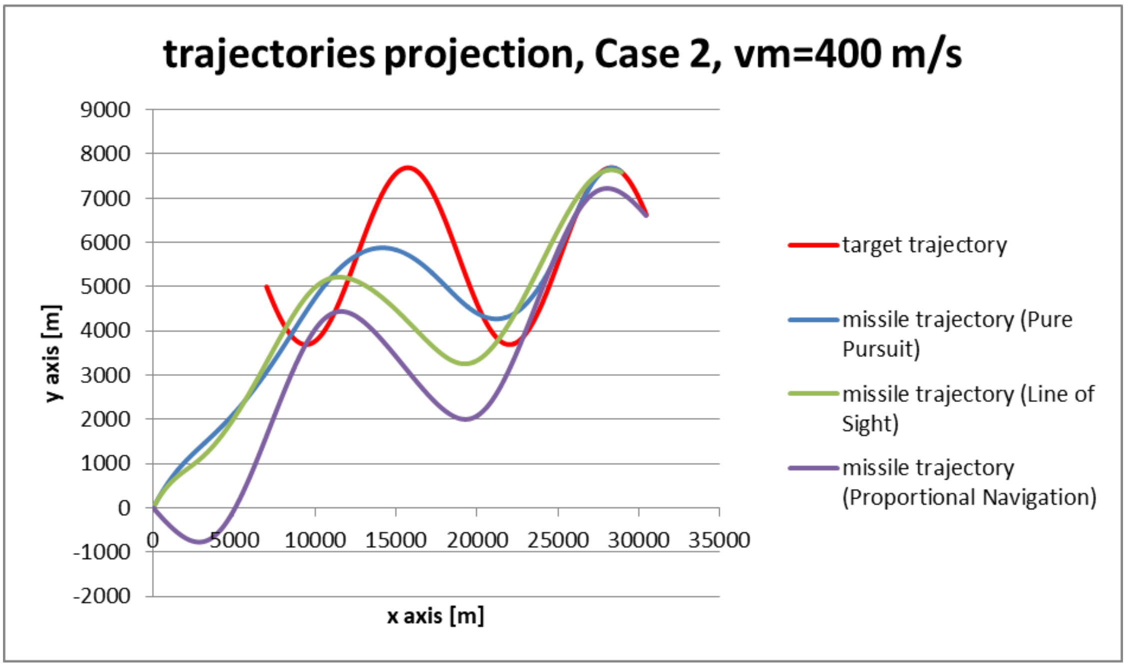

Image 54: XY trajectories projection, Case 2, vm= 400 m/s

It is seen in the projection of the trajectory when applying Pure pursuit and Line of Sight, that the missiles behaves similarly, because both are "pursuing" the target, the only difference is that the missile is pointing always the target, in such a way that the missile velocity vector is aligned with the missile-target line of sight, while in Line of Sight guidance, the missile is forced to have the same angular rate with respect to the launch platform so that it keeps aligned with the launch platform-target line of sight, which is a more advanced pursuit strategy. Translated into the path the missile travels, both trajectories are curvilinear and very similar between them from a behaviour point of view, since both are "pursuing" the target, as comented. It is also seen that trajectories are closer between those two guidance laws if the velocity is lower and they differ more when velocity is greater, which means that impact time will be more similar at low speeds than at high ones. In addition, when applying Pure Pursuit guidance, the missile, in the last phase, pursues the target imitating the target's movement, therefore impact takes place at the back of the target, while in Line of Sight the impact takes place laterally, which is fastest. In a few words, Line of sight Guidance will take lower time to collide with its target than Pure Pursuit.

However, in the case of applying Proportional Navigation, it is not possible to predict as easily if it will collide with its target fastest or slower than using the other guidance strategies. As we mentioned, when using Proportional Navigation, the missile path has the same behaviour as the target, and therefore for a target's sinusoidal path, the missile would follow also a sinusoidal trajectory until impact, which is translated into a high numer of manoeuvres and travelled distance prior impact, and so on, using the other two strategies, the missile will collide before with the target since they are pursuing the target rather than trying to follow an ideal collision trajectory, which supposes a lower travelled distance, as it can be seen on Image 54.

5. CONCLUSIONS: SELF-ANALYSES.

To conclude the work, it is very important to be critical with the results of it and analyse if the objectives have been achieved. In my case, I am aware that in order to calculate the trajectories several assumptions and simplifications have been made. Due to those simplifications, it could be state that with the Matlab code, what I have obtain is a first approach to ideal classical Guidance Laws, since the Dead Reckoning technique works ideally, which in real case there would have been positioning errors as well as signal errors.

In addition, since velocities are constant the problem has consisted of a kinematic one without taking into account the acting forces. From my point of view, that would be the factor that differs most from reality. It is true that when designing a missile the goal is to obtain a high speed flying device, so that its velocity is always greater than the target's one, in such a way that if the target accelerates, the missile does not need to accelerate as there will be an impact anyway. However, since I have worked with a supersonic missile, so that its results evidence clearly the main trends of these guidance laws, to properly compare them, the missile in reality will not start at the supersonic speed, there will be an accelerating time after the lunching, until reaching design velocity. Nevertheless, the majority of the trajectory would have been made in reality at constant velocity. If instead of working with a supersonic cases, I had worked with small velocities, the acceleration time after the launching will have been smaller and the results in this first instants, would be closer to reality. However, in reality missiles are normaly launched with no initial guidance so that during the acceleration time, they follow a linear path and therefore load factors are null, and once the missile has high velocity guidance starts. In my case, I am focused on a parametric study of the load factor and, since structurally the most important point is the maximum lateral accelerations achieved, in this first instants they would have been lower than those we have obtained, therefore, no modelling this first part is not critical. Moreover, for the scope of the project which is to compare the trajectories between guidance laws, maintaining a constant missile speed simplifies the computational power of the calculations as well as the programming procedure, without reducing the veracity of the results.

Apart from those simplifications, it can be considered that the project has been performed successfully because its objectives have been achieved. It has been possible to obtain an algorithm to compute any trajectory independently from the target's trajectory or its velocity, in such a way, that the missile path can be modelled. It has also been studied how velocity affects different guidance laws in order to know which one is suitable for each occasion and which one presents better results. It has been understood from a theoretical point of view, how wind gusts may affect the trajectory as well as the load factor. Therefore, the main goals proposed in the beginning of the project have been achieved successfully.

Finally, it must be highlighted the importance of the project, since it allows to calculate the trajectory of a missile towards one target and know its load factor, which would be the normal acceleration that the structure would need to stand, which will allow to choose the specific materials for its construction, and therefore reduce weight, obtain materials that could be cheaper, etc. Out of the scope of the project are the aerodynamic forces acting on the body and the compressibility phenomena, since as it was mentioned, the project has remained a pure kinematic problem, nevertheless, in reality, a study of the aerodynamics of a missile is as important as a study of the structural accelerations suffered.

6. ANNEX.

6.1. Matlab Code.

```
clear all
%% case establishment
guidance_law=input('Select Guidance Law: 1 for Pure Pursuit, 2 for
Proportional Navigation, 3 for Line of Sight : ');
target_trajectory=input('Select target trajectory: 1 for linear +
parabolic, 2 for sinusoidal: ');
wind_effect=input('Take into account wind effect: 1 for yes, 0 for no:
');

%% generic data
vt=300; %(m/s)
vm=500; %(m/s)
xm0=0; % 'x' initial missile position (m)
ym0=0;%'y' initial missile position (m)
zm0=0; %'z' initial missile position (m)
if target_trajectory==1
    xt0=1000; %'x' initial target position (m)
    yt0=2000; %'y' initial target position (m)
    zt0=10000; %'z' initial target position (m)
elseif target_trajectory==2
    xt0=7000; %'x' initial target position (m)
    yt0=5000; %'y' initial target position (m)
    zt0=10000; %'z' initial target position (m)
end

rmt0=[xt0-xm0,yt0-ym0,zt0-zm0]; %missile-target vector position (m)
r0=norm(rmt0); %% missile-target distance module (m)

if target_trajectory==1
    vt_vector=vt*[1,0,1]/sqrt(1^2+1^2); % (m/s)
elseif target_trajectory==2
    m=cos(xt0/2000);
    vt_vector=vt*[1,m,0.5]/sqrt(1^2+0.5^2+m^2); % (m/s)
end
if guidance_law==1
    vm_vector=vm*rmt0/r0; % (m/s)
elseif guidance_law==2
    syms t

time=solve(sqrt((rmt0(1)+vt_vector(1)*t)^2+(rmt0(2)+vt_vector(2)*t)^2+
(rmt0(3)+vt_vector(3)*t)^2)==vm*t);
    time=double(time); % (s)
    vm_vector=[xt0+vt_vector(1)*time-xm0, yt0+vt_vector(2)*time-ym0,
zt0+vt_vector(3)*time-zm0]/time; % (m/s)
elseif guidance_law==3
    alfa=0; % (rad)
end


%% steering signal generation
if guidance_law==1
    evaluation_time=0.01; % (s)
elseif guidance_law==2 || guidance_law==3
    evaluation_time=0.1; % (s)
end
```

```
%% vectors determination
if guidance_law==1 || guidance_law==2
%vector position
y=zeros(1,7);
y(1,1)=xt0; % (m)
y(1,2)=yt0; % (m)
y(1,3)=zt0; % (m)
y(1,4)=xm0; % (m)
y(1,5)=ym0; % (m)
y(1,6)=zm0; % (m)
y(1,7)=r0; % (m)

%vector velocity
v=zeros(1,3);
v(1,1)=vm_vector(1); %vx (m/s)
v(1,2)=vm_vector(2); %vy (m/s)
v(1,3)=vm_vector(3); %vz (m/s)

elseif guidance_law==3
    y=zeros(1,8);
%state vector
y(1,1)=xt0; %target x position (m)
y(1,2)=yt0; %target y position (m)
y(1,3)=zt0; %target z position (m)
y(1,4)=vm*evaluation_time*rmt0(1)/r0; %missile x position (m)
y(1,5)=vm*evaluation_time*rmt0(2)/r0; %missile y position (m)
y(1,6)=vm*evaluation_time*rmt0(3)/r0; %missile z position (m)
y(1,8)=alfa; % (rad)
rm=norm([y(1,4) y(1,5) y(1,6)]); %missile-base line of sight (m)

%vector velocity
v=zeros(1,3);
v(1,1)=(y(1,4)-xt0)/evaluation_time; %vx (m/s)
v(1,2)=(y(1,5)-yt0)/evaluation_time; %vy (m/s)
v(1,3)=(y(1,6)-xt0)/evaluation_time; %vz (m/s)
end

%vector acceleration
acc=zeros(1,3); % (m/s^2)

%vector load factor
n=zeros(1,1);

%parameters for programming
i=1;
p=1;

%% wind drift establishment
if wind_effect==0
    wind_drift=[0,0,0]; % (m/s)
elseif wind_effect==1
    wind_drift=[0,10,0]; % (m/s)
end
%% guidance law
if target_trajectory==1
    if guidance_law==1
        for k=1:evaluation_time:1000
    i=i+1;
    %% target_displacement
    vt_vector_disp=vt_vector+wind_drift; % (m/s);
```

```
    t_position=vt_vector_disp*evaluation_time+[xt0,yt0,zt0]; %(m)

    %variables update
    xt0=t_position(1,1); % (m)
    yt0=t_position(1,2); % (m)
    zt0=t_position(1,3); % (m)
    y(i,1)= xt0; % (m)
    y(i,2)=yt0; % (m)
    y(i,3)=zt0; % (m)
    %% missile displacement
    vm_vector_disp=vm_vector+wind_drift; % (m/s);
    m_position=vm_vector_disp*evaluation_time+[xm0,ym0,zm0]; % (m)

    %variables update
    xm0=m_position(1,1); % (m)
    ym0=m_position(1,2); % (m)
    zm0=m_position(1,3); % (m)
    rmt0=[xt0-xm0,yt0-ym0,zt0-zm0];%missile-target vector position (m)
    r0=norm(rmt0); % (m)
    vm_vector=vm*rmt0/r0; % (m/s)
    y(i,4)= xm0; % (m)
    y(i,5)=ym0; % (m)
    y(i,6)=zm0; % (m)
    y(i,7)=r0; % (m)

  %% maneouver
    if r0<5000 % (m)
        if p==1
            a=1/10000;
            b=-2*a*xt0;
            p=0;
            ii=i;
        end
    m=2*a*xt0+b;
    vt_vector=vt*([1,m,0]/sqrt(m^2+1^2)); %(m/s)
    end
%% impact detection
        if r0<1 % (m)
            break
        end

 %% velocity and acceleration calculation
    v(i,1)=vm_vector(1)+wind_drift(1); %(m/s)
    v(i,2)=vm_vector(2)+wind_drift(2); %(m/s)
    v(i,3)=vm_vector(3)+wind_drift(3); %(m/s)
    acc(i-1,1)=(v(i,1)-v(i-1,1))/evaluation_time; %(m/s^2)
    acc(i-1,2)=(v(i,2)-v(i-1,2))/evaluation_time; %(m/s^2)
    acc(i-1,3)=(v(i,3)-v(i-1,3))/evaluation_time; %(m/s^2)
    n(i-1,1)=sqrt((acc(i-1,1))^2+(acc(i-1,2)^2+(acc(i-1,3))^2))/9.81;
end
    elseif guidance_law==2
        for k=1:evaluation_time:1000
    i=i+1;
    %% target_displacement
    vt_vector_disp=vt_vector+wind_drift; % (m/s)
    t_position=vt_vector_disp*evaluation_time+[xt0,yt0,zt0]; %(m)

    %variables update
    xt0=t_position(1,1); % (m)
    yt0=t_position(1,2); % (m)
```

```
    zt0=t_position(1,3); % (m)
    y(i,1)= xt0; % (m)
    y(i,2)=yt0; % (m)
    y(i,3)=zt0; % (m)
    %% missile displacement
    vm_vector_disp=vm_vector+wind_drift; % (m/s)
    m_position=vm_vector_disp*evaluation_time+[xm0,ym0,zm0]; % (m)

    %variables update
    xm0=m_position(1,1); %(m)
    ym0=m_position(1,2); % (m)
    zm0=m_position(1,3); % (m)
    rmt0=[xt0-xm0,yt0-ym0,zt0-zm0];%missile-target line of sight
    r0=norm(rmt0); % (m)
    syms t

time=solve(sqrt((rmt0(1)+vt_vector(1)*t)^2+(rmt0(2)+vt_vector(2)*t)^2+
(rmt0(3)+vt_vector(3)*t)^2)==vm*t);
    time=double(time); % (s)
    vm_vector=[xt0+vt_vector(1)*time-xm0, yt0+vt_vector(2)*time-ym0,
zt0+vt_vector(3)*time-zm0]/time; % (m/s)
    y(i,4)= xm0; % (m)
    y(i,5)=ym0; % (m)
    y(i,6)=zm0; % (m)
    y(i,7)=r0; % (m)

  %% maneouver
    if r0<5000 % (m)
        if p==1
            a=1/10000;
            b=-2*a*xt0;
            p=0;
            ii=i;
        end
    m=2*a*xt0+b;
    vt_vector=vt*([1,m,0]/sqrt(m^2+1^2)); %(m/s)
    end

%% impact detection
    if r0<20 % (m)
        break
    end
          %% velocity and acceleration calculation
    v(i,1)=vm_vector(1)++wind_drift(1); %(m/s)
    v(i,2)=vm_vector(2)++wind_drift(2); %(m/s)
    v(i,3)=vm_vector(3)++wind_drift(3); %(m/s)
    acc(i-1,1)=(v(i,1)-v(i-1,1))/evaluation_time; %(m/s^2)
    acc(i-1,2)=(v(i,2)-v(i-1,2))/evaluation_time; %(m/s^2)
    acc(i-1,3)=(v(i,3)-v(i-1,3))/evaluation_time; %(m/s^2)
    n(i-1,1)=sqrt((acc(i-1,1))^2+(acc(i-1,2)^2+(acc(i-1,3))^2))/9.81;

        end
    elseif guidance_law==3
        for k=1:evaluation_time:1000
    i=i+1;
    %% target_displacement
    t_position1=[xt0,yt0,zt0]; % (m)
    vt_vector_disp=vt_vector+wind_drift; % (m/s)
    t_position2=vt_vector_disp*evaluation_time+t_position1; % (m)
```

```
alfa=acosd(dot(t_position1,t_position2)/(norm(t_position1)*norm(t_posi
tion2))); % (m)
    %variables update
    xt0=t_position2(1); % (m)
    yt0=t_position2(2); % (m)
    zt0=t_position2(3); % (m)
    y(i,1)= xt0; % (m)
    y(i,2)=yt0; % (m)
    y(i,3)=zt0; % (m)
    y(i,8)=alfa; % (rad)
    %% missile displacement
    syms c
    m_c=solve((vm*evaluation_time)^2==rm^2+c^2-2*rm*c*cosd(alfa));
    m_c=double(m_c(2));
    rm=m_c; % (m)

m_position2=rm*t_position2/norm(t_position2)+wind_drift*evaluation_tim
e;  % (m)
    %variables update
    xm0=m_position2(1);% (m)
    ym0=m_position2(2);% (m)
    zm0=m_position2(3);% (m)
    rmt0=[xt0-xm0,yt0-ym0,zt0-zm0];%missile-target line of sight
    r0=norm(rmt0);% (m)
    y(i,4)= xm0;% (m)
    y(i,5)=ym0;% (m)
    y(i,6)=zm0;% (m)
    y(i,7)=r0; % (m)
 %% maneouver
    if r0<5000 % (m)
        if p==1
            a=1/10000;
            b=-2*a*xt0;
            p=0;
            ii=i;
        end
    m=2*a*xt0+b;
    vt_vector=vt*([1,m,0]/sqrt(m^2+1^2)); % (m/s)
    end
%% impact detection
    if r0<20 % (m)
        break
    end
              %% velocity and acceleration calculation
    v(i,1)= (y(i,4)-y(i-1,4))/evaluation_time; % (m/s)
    v(i,2)= (y(i,5)-y(i-1,5))/evaluation_time; % (m/s)
    v(i,3)= (y(i,6)-y(i-1,6))/evaluation_time;% (m/s)

    acc(i-1,1)=(v(i,1)-v(i-1,1))/evaluation_time; % (m/s^2)
    acc(i-1,2)=(v(i,2)-v(i-1,2))/evaluation_time; % (m/s^2)
    acc(i-1,3)=(v(i,3)-v(i-1,3))/evaluation_time; % (m/s^2)
    n(i-1,1)=sqrt((acc(i-1,1))^2+(acc(i-1,2)^2+(acc(i-1,3))^2))/9.81;
        end
    end
    %% anomality remove in trajectory change instant
acc(ii,1)=acc(ii+1,1); % (m/s^2)
acc(ii,2)=acc(ii+1,2); % (m/s^2)
acc(ii,3)=acc(ii+1,3); % (m/s^2)
n(ii,1)=n(ii+1,1);
%% displayed information
```

```
nmax1= max(n(2:ii-1));
nmax2=max(n(ii:length(n)));
time_impact=length(y)*evaluation_time;
disp('maximum load factor on target linear trajectory: ')
disp(max(nmax1));
disp('maximum load factor on target parabolic trajectory: ')
disp(nmax2);
disp('time to impact: ')
disp(time_impact);
save state_vector_500.txt y -ascii
save load_factor_500.txt n -ascii
elseif target_trajectory==2
    if guidance_law==1
        for k=1:evaluation_time:1000
    i=i+1;
    %% target_displacement
    m=cos(xt0/2000);
    vt_vector=vt*[1,m,0.5]/sqrt(1^2+m^2+0.5^2); % (m/s)
    vt_vector_disp=vt_vector+wind_drift; % (m/s)
    t_position=vt_vector_disp*evaluation_time+[xt0,yt0,zt0]; % (m)

    %variables update
    xt0=t_position(1,1); % (m)
    yt0=t_position(1,2); % (m)
    zt0=t_position(1,3); % (m)
    y(i,1)= xt0; % (m)
    y(i,2)=yt0; % (m)
    y(i,3)=zt0; % (m)
    %% missile displacement
    vm_vector_disp=vm_vector+wind_drift; % (m/s)
    m_position=vm_vector_disp*evaluation_time+[xm0,ym0,zm0]; % (m)

    %variables update
    xm0=m_position(1,1); % (m)
    ym0=m_position(1,2); % (m)
    zm0=m_position(1,3); % (m)
    rmt0=[xt0-xm0,yt0-ym0,zt0-zm0];%missile-target vector position (m)
    r0=norm(rmt0); % (m)
    vm_vector=vm*rmt0/r0; % (m/s)
    y(i,4)= xm0; % (m)
    y(i,5)=ym0; % (m)
    y(i,6)=zm0;% (m)
    y(i,7)=r0;% (m)

%% impact detection
        if r0<1
            break
        end

 %% velocity and acceleration calculation
    v(i,1)=vm_vector(1)+wind_drift(1); % (m/s)
    v(i,2)=vm_vector(2)+wind_drift(2); % (m/s)
    v(i,3)=vm_vector(3)+wind_drift(3); % (m/s)
    acc(i-1,1)=(v(i,1)-v(i-1,1))/evaluation_time; % (m/s^2)
    acc(i-1,2)=(v(i,2)-v(i-1,2))/evaluation_time; % (m/s^2)
    acc(i-1,3)=(v(i,3)-v(i-1,3))/evaluation_time; % (m/s^2)
    n(i-1,1)=sqrt((acc(i-1,1))^2+(acc(i-1,2)^2+(acc(i-1,3))^2))/9.81;
end
    elseif guidance_law==2
        for k=1:evaluation_time:1000
    i=i+1;
```

```
    %% target_displacement
    m=cos(xt0/2000);
    vt_vector=vt*[1,m,0.5]/sqrt(1^2+0.5^2+m^2); % (m/s)
    vt_vector_disp=vt_vector+wind_drift; % (m/s)
    t_position=vt_vector_disp*evaluation_time+[xt0,yt0,zt0]; % (m)

    %variables update
    xt0=t_position(1,1); % (m)
    yt0=t_position(1,2); % (m)
    zt0=t_position(1,3); % (m)
    y(i,1)= xt0; % (m)
    y(i,2)=yt0; % (m)
    y(i,3)=zt0; % (m)
    %% missile displacement
    vm_vector_disp=vm_vector+wind_drift; % (m/s)
    m_position=vm_vector_disp*evaluation_time+[xm0,ym0,zm0]; % (m)

    %variables update
    xm0=m_position(1,1); % (m)
    ym0=m_position(1,2);% (m)
    zm0=m_position(1,3);% (m)
    rmt0=[xt0-xm0,yt0-ym0,zt0-zm0];%missile-target line of sight (m)
    r0=norm(rmt0); % (m)
    syms t

time=solve(sqrt((rmt0(1)+vt_vector(1)*t)^2+(rmt0(2)+vt_vector(2)*t)^2+
(rmt0(3)+vt_vector(3)*t)^2)==vm*t);
    time=double(time);% (s)
    vm_vector=[xt0+vt_vector(1)*time-xm0, yt0+vt_vector(2)*time-ym0,
zt0+vt_vector(3)*time-zm0]/time; % (m/s)
    y(i,4)= xm0; % (m)
    y(i,5)=ym0; % (m)
    y(i,6)=zm0; % (m)
    y(i,7)=r0; % (m)

%% impact detection
    if r0<20 % (m)
        break
    end
          %% velocity and acceleration calculation
    v(i,1)=vm_vector(1)+wind_drift(1); % (m/s)
    v(i,2)=vm_vector(2)+wind_drift(2); % (m/s)
    v(i,3)=vm_vector(3)+wind_drift (3); % (m/s)
    acc(i-1,1)=(v(i,1)-v(i-1,1))/evaluation_time; % (m/s^2)
    acc(i-1,2)=(v(i,2)-v(i-1,2))/evaluation_time; % (m/s^2)
    acc(i-1,3)=(v(i,3)-v(i-1,3))/evaluation_time; % (m/s^2)
    n(i-1,1)=sqrt((acc(i-1,1))^2+(acc(i-1,2)^2+(acc(i-1,3))^2))/9.81;

        end
    elseif guidance_law==3
        for k=1:evaluation_time:100
    i=i+1;
    %% target_displacement
    t_position1=[xt0,yt0,zt0]; % (m)
    m=cos(xt0/2000);
    vt_vector=vt*[1,m,0.5]/sqrt(1+0.5^2+m^2); % (m/s)
    vt_vector_disp=vt_vector; % (m/s)
    t_position2=vt_vector_disp*evaluation_time+t_position1; % (m)

alfa=acosd(dot(t_position1,t_position2)/(norm(t_position1)*norm(t_posi
tion2))); % (rad)
```

```
    %variables update
    xt0=t_position2(1); % (m)
    yt0=t_position2(2); % (m)
    zt0=t_position2(3); % (m)
    y(i,1)= xt0; % (m)
    y(i,2)=yt0; % (m)
    y(i,3)=zt0; % (m)
    y(i,8)=alfa; % (rad)
    %% missile displacement
    syms c
    m_c=solve((vm*evaluation_time)^2==rm^2+c^2-2*rm*c*cosd(alfa));
    m_c=double(m_c(2)); % (m)
    rm=m_c; % (m)

m_position2=rm*t_position2/norm(t_position2)+wind_drift*evaluation_tim
e; % (m)
    %variables update
    xm0=m_position2(1); % (m)
    ym0=m_position2(2); % (m)
    zm0=m_position2(3); % (m)
    rmt0=[xt0-xm0,yt0-ym0,zt0-zm0];%missile-target line of sight % (m)
    r0=norm(rmt0); % (m)
    y(i,4)= xm0;% (m)
    y(i,5)=ym0; % (m)
    y(i,6)=zm0;% (m)
    y(i,7)=r0;% (m)
%% impact detection
    if r0<20 % (m)
        break
    end
              %% velocity and acceleration calculation
    v(i,1)= (y(i,4)-y(i-1,4))/evaluation_time; % (m/s)
    v(i,2)= (y(i,5)-y(i-1,5))/evaluation_time; % (m/s)
    v(i,3)= (y(i,6)-y(i-1,6))/evaluation_time; % (m/s)

    acc(i-1,1)=(v(i,1)-v(i-1,1))/evaluation_time; % (m/s^2)
    acc(i-1,2)=(v(i,2)-v(i-1,2))/evaluation_time; % (m/s^2)
    acc(i-1,3)=(v(i,3)-v(i-1,3))/evaluation_time;% (m/s^2)
    n(i-1,1)=sqrt((acc(i-1,1))^2+(acc(i-1,2)^2+(acc(i-1,3))^2))/9.81;
        end
%% displayed information
    end
nmax=max(n(2:end));
time_impact=length(y)*evaluation_time;
disp('maximum load factor: ')
disp(nmax);
disp('time to impact: ')
disp(time_impact);
save state_vector_500.txt y -ascii
save load_factor_500.txt n -ascii
end

%% plot results
%% velocity plot
figure
subplot(1,3,1);
plot(v(2:length(v),1));
title ('vx');
if guidance_law==1
    xlabel('time (E-2 s)');
else
```

```
  xlabel('time (E-1 s)');
end
ylabel('m/s');
grid on
subplot(1,3,2);
plot(v(2:length(v),2));
title ('vy');
if guidance_law==1
    xlabel('time (E-2 s)');
else
  xlabel('time (E-1 s)');
end
ylabel('m/s');
grid on
subplot(1,3,3);
plot(v(2:length(v),3));
title ('vz');
if guidance_law==1
    xlabel('time (E-2 s)');
else
  xlabel('time (E-1 s)');
end
ylabel('m/s');
grid on

%acceleration plot
figure
subplot(1,3,1);
plot(acc(2:length(acc),1));
title ('ax');
if guidance_law==1
    xlabel('time (E-2 s)');
else
  xlabel('time (E-1 s)');
end
ylabel('m/s^2');
grid on
subplot(1,3,2);
plot(acc(2:length(acc),2));
title ('ay');
if guidance_law==1
    xlabel('time (E-2 s)');
else
  xlabel('time (E-1 s)');
end
ylabel('m/s^2');
grid on
subplot(1,3,3);
plot(acc(2:length(acc),3));
title ('az');
if guidance_law==1
    xlabel('time (E-2 s)');
else
  xlabel('time (E-1 s)');
end
ylabel('m/s^2');
grid on

%load factor plot
figure
plot(n(2:length(n)));
```

```
title ('load factor');
if guidance_law==1
    xlabel('time (E-2 s)');
else
  xlabel('time (E-1 s)');
end
grid on

%trajectory plot
figure
plot3(y(1:i,1),y(1:i,2),y(1:i,3),'r');
hold on
plot3(y(1:i,4),y(1:i,5),y(1:i,6),'b');
hold on
plot3(xt0,yt0,zt0,'x', 'LineWidth',3);
hold on
plot3(y(1,1),y(1,2),y(1,3),'r*')
hold on
plot3(0,0,0,'b*')
grid on
xlabel('x axis [m]');
ylabel('y axis [m]');
zlabel('z axis [m]');
legend('target trajectory','missile trajectory','impact', 'initial
target position','initial missile position');
title ('line of sight guidance law');
```

Some simplifications of the code:

For Pure Pursuit the results are updated every 0.1 seconds, while for Proportional Navigation and Line of Sight guidance, the results are updated every 0.1 seconds, in order to reduce the computational power. This fact influences the impact codition, since as the missile is flying at very high velocities in a small time interval, therefore in order to reduce the distance at which it is considered that impact takes place it is needed to reduce the time interval, which supposes high computational power.

For Case 1, the movement consists of a comibination of a linear plus a parabolic trajectory. The vertex of the parabola is located at the point where the target manoeuvres in order to be a real trajectory. However, from the point of view of the missile, when applying Pure Pursuit and Line of Sight guidance, when using *solve* Matlab function, there is a discontinuity in the algorithm, which results in a jump of the calculated parameters, and therefore the results obtained on that instant are not correct. In order to solve that, we remove that result and we consider the results in the first part (linear target's trajectory), and then in the second part (parabolic target's trajectory), as they were independent trajectories.

The difference between results, would be the real change in results when manoeuvering.

For the load factor, the value between intant 0 and first instant is also removed, because it supposes to change from no velocity to maximum velocity instantaneously which, as it was explained on section 5 is a rough simplification. Therefore, the result on that instant is not correct, because in reality there is an acceleration time until reaching maximum velocity. Therefore, we are not considering the result on that instant.

7. BIBLIOGRAPHY

- *Misiles*, Escuela Técnica Superior de Ingenieros Aeronáuticos, Madrid, Octubre 1998
- *Fundamentals of Naval Weapons Systems*, Federation of American Scientists
- *Missile Guidance and Control Systems*, George M. Siouris
- *Sistema Electrónico de Comunicación de un Misil*, Vicente Vazquez Moreno
- *Mechanical Gyroscopes,* Warsaw University of Technology, Janusz Gajda
- *Attitude determination using inertial sensors*, Warsaw University of Technology, Przemysław Bibik
- *Missile Guidance Laws*, National Programme on Technology Enhanced Learning
- *Aircraft structures for engineering students*, T.H.G. Megson