Fixed Wing Aircraft Collision Avoidance Using Collision Cone Theory

Andrew T. Beary III
University of Texas, Arlington, 76013
atb7256@mavs.uta.edu
August 2022

Supervising Committee:

Chakravarthy, Animesh - Supervising Professor
Wang, Shuo Linda
Bowling, Alan

Abstract

This thesis develops the guidance and control components of an aircraft collision avoidance system, using a collision cone-based approach. These guidance and control laws are developed by considering the nonlinear relative kinematic equations of the aircraft that are on a collision course, as well as the nonlinear flight dynamics of the aircraft performing the collision avoidance maneuver. The longitudinal and lateral-directional axes of the aircraft are considered separately. The developed guidance and control laws are subsequently integrated with the nonlinear aircraft dynamics in a MATLAB/Simulink framework, and simulations performed to demonstrate the working of the overall system.

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

Motivation

Whether a flight vehicle is manned or unmanned, it is essential for vehicle and mission integrity that the aircraft be able to avoid collisions, be it with the ground, a stationary object, or a moving object such as another aircraft. Should collision occur, the minimal and best result is expensive aircraft damage, while the worst outcomes could be mission failure, loss of craft, loss of crew, collateral damage, or even some combination thereof. Obviously, these are all undesirable results even in the best case, and so strategies must be explored for how to mitigate collisions.

Aircraft do occasionally crash into obstacles like buildings, mountains, or even other aircraft. Ref. [12] discusses how two US military aircraft collided during a mid-air refueling operation gone wrong on September 29th, 2020, with everyone on board both planes luckily surviving. However, one aircraft was entirely destroyed when the pilot ejected, with the other aircraft managing to land with heavy damage. On the civilian side, and much more recently, Ref. [14] shows that four people died when two aircraft collided with each other in mid-air in Nevada on July 7th, 2022. Another civilian craft barely clipped a building with the wing of the aircraft in California on June 10th, 2022, and crashed into a field, killing the pilot in the process according to Ref. [5]. Each and every aircraft collision is potentially devastating to all those involved, and so methods to avoid collisions are important to investigate.

Generally, there exist separation standards, which, when practiced properly, "...ensures safe separation from the ground, from other aircraft and from protected airspace" according to Ref. [16]. These standards are as they sound; industry standard operating procedures for how far apart aircraft should fly from any possible obstacle in order to minimize the possibility of collisions. There even exist airborne separation assurance systems (ASAS), which are long-range on-board systems used to keep track of separation between aircraft, warning the pilot when such separation becomes shorter than recommended.

Also, irrespective of any separation standards, there are first-line defenses against aircraft collision consisting of relatively simple, human based methods, such as the aircraft pilot simply seeing something in the path of the aircraft and then turning the craft to avoid crashing. A pilot may also receive early warning from air traffic control, or from

other ground-based equipment, that another aircraft is present in the same airspace that may present the possibility of collision, along with recommendations for changes to flight path.

However, besides these, other independent systems do exist that detect an incoming collision, and act to warn the pilot of the incoming danger and advise on optimal flight paths. An airborne collision avoidance system (ACAS) is built into aircraft to fulfill this independent collision detection requirement. It is a short-range system designed to mitigate aircraft on aircraft collision, and it "interrogates the ... transponders of nearby aircraft ('intruders') and from the replies tracks their altitude and range and issues alerts to the pilots, as appropriate", according to Ref. [2]. If collision is still imminent after a warning is given, then such systems may, in some cases with certain aircraft, even act outside of the pilot's control to maneuver the craft to reduce the risk of collision as much as possible. Though, this seems to be implemented only in the minority of cases.

There are even similar systems designed to avoid collision with the ground, called ground collision warning systems (GCWS) or ground proximity warning systems (GPWS) which use a radar altimeter to detect proximity to the ground and "provides a distinctive warning to pilots", from Ref. [19]. Such systems have been in development since the 1970's, and can even be angled to look in front of the aircraft and warn of flying into mountains or buildings. These are quite common on commercial airliners, and variations can be found on military craft, but with added complexity to account for more drastic or extreme operating conditions.

Most of the systems and methods described so far to avoid collision explicitly only seek to warn the pilot of possible collision, relying on them to enact a change in flight path to avoid the collision. However, a human element will always introduce unpredictable possible sources of failure, as the pilot could choose to ignore warnings, could possibly not notice them, or could be otherwise incapacitated. Therefore, automatic systems are also important avenues to explore when discussing collision avoidance.

One such automatic aircraft collision mitigation method utilizes what is known as the collision cone, which is first introduced in Ref. [4], and is described there as being "... used to predict the possibility of collision between two objects and to design collision avoidance strategies." For an example in simple terms, take a point object and a

sphere. If a vector is drawn from the point object to all of the points on the sphere that it could collide with if it moved along that vector, then the shape of that collection of vectors is a cone. If the actual velocity vector of the point object is within this collision cone, then collision will occur. This collision cone can also be projected in two dimensions, with this example then becoming a collision between a point and a circle. The cone can also be applied when both objects are moving, in which case relative velocity vectors are used. The conditions for determining collision based on relative velocity vectors indicate whether a certain trajectory is within the collision cone or not, and it is obviously desired that a velocity vector inside the collision cone be steered out of the cone so that the two objects can avoid collision.

The collision cone approach has been used in multiple contexts in multiple publications, from collision avoidance between a robot and an irregularly shaped obstacle in the original Ref. [4], to aircraft trajectories in Refs. [3], [11], and [20] through [23], to cars with automatic driver collision avoidance assistance in Refs. [8] through [10]. These differences in application will of course lead to variations in certain equations; however, it generally holds that the underlying principles apply in some form or another regardless of applications to 2-D or 3-D movement, or with regular or irregular shaped objects, or other such contextual variations. The ultimate goal remains to first determine if collision is forthcoming, then enact a maneuver to avoid collision.

To this end, this thesis examines two-dimensional collision scenarios between two fixed-wing aircraft, using collision cone theory to determine when the two aircraft are on a collision course. The scenarios are conducted in two different frames, either viewing the aircraft along the longitudinal axis or along the lateral-directional axis. Guidance laws are developed which are applied to one of the aircraft in order to change its flight trajectory, and these guidance laws are integrated in succession with control laws and nonlinear aircraft dynamics to create a simulation framework that deflects the aircraft's control surface(s) appropriately and results in the two aircraft avoiding collision.

Thesis Organization

In this work, two different 2-D frames are considered: the longitudinal frame and the lateral-directional frame. Chapter 2 describes the 2-D definition of the collision cone. The simulations run in the longitudinal reference frame are described fully in chapter 3, and those of the lateral-directional reference frame are featured in chapter 4. Both

chapters 3 and 4 begin with describing the derivations of the guidance laws and control laws used for each respective frame, as well as the respective nonlinear dynamics used for the chosen aircraft, and finally the complete simulation framework in Simulink. Then, those chapters conclude with the results of the simulations, and descriptive comments about the plots therein. Chapter 5 presents the overall conclusions, and possible directions for future work using the content developed here.

2. BACKGROUND

This chapter begins with the description of collision cone theory in two dimensions. Starting with a case where two non-maneuvering aircraft are represented as point objects, the criteria for collision are established. Then, both aircraft are represented as circles, and a new set of collision criteria are found, corresponding with the collision cone theory subsequently used in this work. Following this, the specific aircraft chosen for the conducted simulations is revealed.

Collision Cone Theory

Consider two aircraft A and B, indicated by red dots in Fig. (1) below. Craft A and craft B are both point objects, and the distance between the two craft is labelled as r, while the angle of the line of sight between the two craft from a reference x-axis is called θ_{LOS}. Each craft has a velocity vector, V_T and V_B, which each act at flight path angles of γ_A and γ_B, respectively.

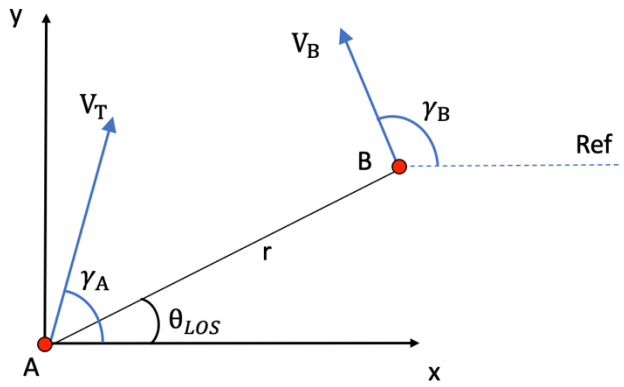

Figure 1: Engagement Geometry

There are two relative velocity components to be considered; let V_r be the rate of change of distance r, and let V_θ be the component relative velocity normal to line AB. As defined in Ref. [4], these are

$$V_r = \dot{r} = V_B \cos(\gamma_B - \theta_{LOS}) - V_A \cos(\gamma_A - \theta_{LOS}) \quad (1)$$

$$V_\theta = r\dot{\theta} = V_B \sin(\gamma_B - \theta_{LOS}) - V_A \sin(\gamma_A - \theta_{LOS})$$

According to Ref. [13], the conditions required for collision to occur between craft A and craft B (assuming constant velocities) are for the rate of change of the distance between the two craft to be decreasing, i.e. $V_r < 0$, and the line of sight to not rotate in inertial space, i.e. $V_\theta = 0$. With both of these conditions being true, collision between craft A and B will occur. These collision conditions are indicated by a red line in a plot of V_r versus V_θ seen in Fig. (2).

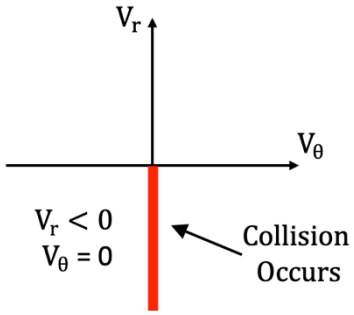

Figure 2: Collision Conditions in (V_θ, V_r) Space

Now, consider that craft A and craft B are no longer point objects, but are instead circles of a radii R_A and R_B respectively, as is shown in Fig. (3). Every other parameter remains the same.

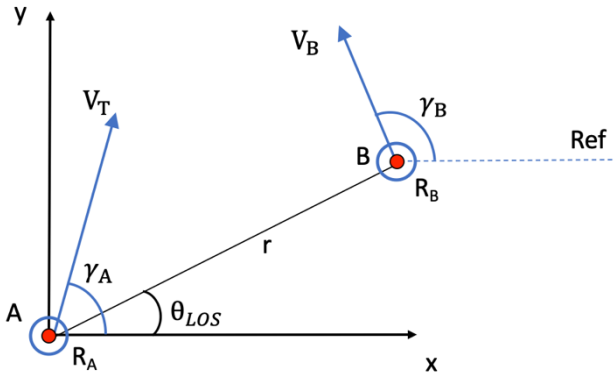

Figure 3: Modified Engagement Geometry

To determine the collision conditions considering the combined size of both aircraft, a miss-distance term must first be derived. This is the closest that craft A gets to craft B, and is found by first starting with Eqs. (1), and recognizing that they are two state equations that govern engagement geometry for two non-maneuvering aircraft. From Ref. [13], there are two other state equations to recognize, which are then (ignoring a_A and a_B terms)

$$\dot{V}_r = \frac{{V_\theta}^2}{r}$$

$$\dot{V}_\theta = -\frac{V_\theta V_r}{r}$$

Then

$\rightarrow$ $r\dot{V}_r = {V_\theta}^2$ Multiply by V_r

$\rightarrow$ $r\dot{V}_\theta = -V_\theta V_r$ Multiply by V_θ

Add together and simplify to get

$$r\dot{V}_r V_r + r\dot{V}_\theta V_\theta = 0$$

Integrate to get

$$\frac{{V_\theta}^2}{2} + \frac{{V_r}^2}{2} = c^2,\ c = \text{constant}$$

Substitute values for time t=0 to get c. Then, the trajectories are

$${V_\theta}^2 + {V_r}^2 = {V_{\theta 0}}^2 + {V_{r0}}^2$$

Where the subscript 0 indicates the respective initial value. At time of closest approach, $V_r = 0$ and t = t_m, previous equation then becomes

$${V_{\theta m}}^2 = {V_{\theta 0}}^2 + {V_{r0}}^2$$

Where the subscript *m* indicates the respective value at time of closest approach. From derivation earlier, we know

$$\frac{-V_{r0}}{r} = \frac{-\dot{r}_0}{r} = \frac{V_\theta \dot{V}_\theta}{{V_{\theta 0}}^2}$$

Integrate both sides and set them equal to each other to get

$$\frac{{r_0}^2}{2r^2} = \frac{{V_\theta}^2}{2{V_{\theta 0}}^2}$$

At time of closest approach, r = r_m and ${V_\theta}^2 = {V_{\theta m}}^2$, which yields

$$\left(\frac{r_0}{r_m}\right)^2 = \frac{{V_{\theta m}}^2}{{V_{\theta 0}}^2}$$

Substitute ${V_{\theta m}}^2$ from earlier equation to get

$${r_m}^2 = \frac{{r_0}^2 * {V_{\theta 0}}^2}{{V_{\theta 0}}^2 + {V_{r0}}^2}$$

Which finally is the desired miss distance term

$$r_m = \sqrt{\frac{{r_0}^2 * {V_\theta}^2}{{V_\theta}^2 + {V_r}^2}} \tag{2}$$

Logically, if the two aircraft get closer than the combined sizes of the aircraft, (R_A + R_B), then the two craft have collided. Therefore, the collision conditions are now when $V_r < 0$ and when $r_m \leq (R_A + R_B)$. To show this in the (V_θ, V_r) space, Eq. (2) is set equal to (R_A + R_B), and V_θ is solved for in terms of V_r.

$$V_\theta = \pm \frac{V_r \sqrt{{r_0}^2 - (R_A + R_B)^2}}{(R_A + R_B)} \tag{3}$$

Now, the two lines of Eq. (3) can be plotted to show the limits of the area within which collision will occur, shown in Fig. (4).

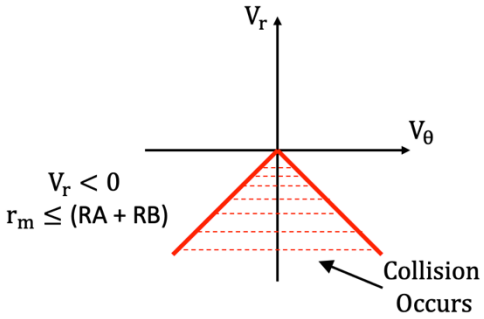

Figure 4: Collision Region

Eqs. (3) are the solid red lines in Fig. (4), and the dotted red lines denote a shaded area showing that any conditions between or on the two solid red lines will result in collision. If the trajectory of the aircraft is inside the above collision region, then its relative velocity vector is within the collection of vectors which lead to collision, which is known as the collision cone. In summary, the collision cone is represented in the (V_θ, V_r) space by:

$$\frac{r_0^2 * V_\theta^2}{V_\theta^2 + V_r^2} \leq (R_A + R_B)^2 \, , V_r < 0$$

For a pair of aircraft on a collision course as is the case in this thesis, the trajectory in the (V_θ, V_r) space will begin inside the collision cone. As time progresses, craft A performs a maneuver so that the (V_θ, V_r) trajectory leaves the collision cone, and thus craft A changes its flight path to avoid a collision with craft B.

3. COLLISION AVOIDANCE DESIGN IN THE LONGITUDINAL AXIS

a. APPROACH

In this chapter, the guidance law is first derived, which synthesizes the desired acceleration required for collision avoidance. Then, the control law is designed, which converts this commanded acceleration into the corresponding control surface deflection, which in this case, is the aircraft's elevator. These are integrated into a Simulink model, along with the aircraft nonlinear dynamics. Finally, simulation results are presented and discussed.

Overall Guidance and Control Architecture

The following Fig. (5) shows the architecture of the guidance and control components of the collision avoidance system in the longitudinal axis.

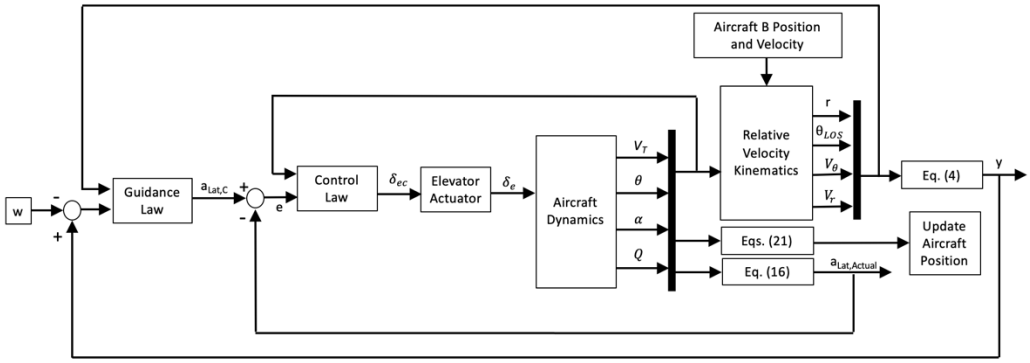

Figure 5: Longitudinal Axis Guidance and Control Architecture

The guidance law generates an output of the commanded lateral acceleration, which is then compared to the actual lateral acceleration. The difference of these two terms is used as the input to the control law, along with the state values from the aircraft dynamics. This control law outputs a commanded elevator deflection. The elevator actuator converts this commanded elevator deflection value into an actual elevator deflection, which is then input into the aircraft dynamics. The dynamics equations result in an output of the state values, which are fed into the relative velocity kinematics, as well as the equation to generate the actual aircraft lateral acceleration, and the equations that update the aircraft position. The relative velocity kinematics are also used to find a term y, which is based on miss distance arguments, and is then compared to a reference input w and fed into the guidance law along with the relative velocity kinematics. All of these variables, equations, and laws are derived or defined in the following sections.

Guidance Law Derivation

First, the conditions under which two aircraft collide have to be determined. Based on collision cone theory, define Eq. (4) as

$$y = r_m^2 - (R_A + R_B)^2 \tag{4}$$

Where R_A, R_B, and r_m are all previously defined. If y = 0, then A will graze craft B, and if y < 0, A and B will collide. We now use dynamic inversion to design the guidance law which is represented as a commanded latax that, if applied, will achieve collision avoidance. An error term e is now defined as

$$e = y - w \tag{5}$$

Where w is some distance factor greater than or equal to zero. Now, as long as y > w > 0, craft A will miss craft B by the distance factor w. From Eq. (5), it can be shown that

$$\dot{e} = \dot{y} \tag{6}$$

With

$$\dot{y} = \frac{dy}{dt} = \frac{\partial y}{\partial r}\frac{dr}{dt} + \frac{\partial y}{\partial V_r}\frac{dV_r}{dt} + \frac{\partial y}{\partial V_\theta}\frac{dV_\theta}{dt} = \frac{\partial y}{\partial r}\dot{r} + \frac{\partial y}{\partial V_r}\dot{V}_r + \frac{\partial y}{\partial V_\theta}\dot{V}_\theta \tag{7}$$

In Eq. (7), the state equations used are

$$\dot{r} = V_r$$

$$\dot{V}_r = \frac{{V_\theta}^2}{r} - a_{Long}\cos(\gamma_A - \theta_{LOS}) - a_{Lat,C}\cos(\delta_A - \theta_{LOS}) \tag{8}$$

$$\dot{V}_\theta = -\frac{V_\theta V_r}{r} - a_{Long}\sin(\gamma_A - \theta_{LOS}) - a_{Lat,C}\sin(\delta_A - \theta_{LOS})$$

Where γ_A is the flight path angle for craft A, and $\delta_A = \gamma_A + {}^{\pi}/_{2}$. The line of sight angle between craft A and craft B is θ_{LOS}, and is illustrated at the beginning of the *Simulation Results* section of this chapter. The longitudinal acceleration a_{Long} is equal to $\dot{V}_T$, which is the rate of change of the forward velocity of the aircraft. This $\dot{V}_T$ term is one of the nonlinear equations of motion, and is defined in the next subsection. The commanded lateral acceleration (latax) $a_{Lat,C}$ is the guidance law term that is being solved for.

Plugging Eqs. (8) into Eq. (7), with terms without $a_{Lat,C}$ or a_{Long} cancelling out, results in

$$\dot{e} = \dot{y} = \frac{\partial y}{\partial V_r}\left(-a_{Long}\cos(\gamma_A - \theta_{LOS}) - a_{Lat,C}\cos(\delta_A - \theta_{LOS})\right) + \frac{\partial y}{\partial V_\theta}\left(-a_{Long}\sin(\gamma_A - \theta_{LOS}) - a_{Lat,C}\sin(\delta_A - \theta_{LOS})\right) \tag{9}$$

In order to drive y to w, e from Eq. (5) must be driven to 0. To this end, we choose

$$\dot{e} = -k_g e \tag{10}$$

Which is a first order exponential decay, with k_g being some arbitrary positive guidance gain, chosen sufficiently large to ensure that e decays to zero before the time to closest approach. Note however, that k_g should not be excessively large as this can lead to acceleration saturation and/or control surface saturation.

Substituting Eq. (10) into Eq. (9), and solving for $a_{Lat,C}$,

$$a_{Lat,C} = \frac{k_g(y-w) + a_{Long}\left(\frac{\partial y}{\partial V_r}\cos(\gamma_A - \theta_{LOS}) + \frac{\partial y}{\partial V_\theta}\sin(\gamma_A - \theta_{LOS})\right)}{\frac{\partial y}{\partial V_\theta} sin(\delta_A - \theta_{LOS}) + \frac{\partial y}{\partial V_r} cos\,(\delta_A - \theta_{LOS})} \tag{11}$$

This Eq. (11) is the longitudinal guidance law, with

$$\frac{\partial y}{\partial V_\theta} = \frac{2r^2 {V_r}^2 V_\theta}{\left({V_r}^2 + {V_\theta}^2\right)^2} \tag{12}$$

and

$$\frac{\partial y}{\partial V_r} = \frac{-2r^2 {V_\theta}^2 V_r}{\left({V_r}^2 + {V_\theta}^2\right)^2} \tag{13}$$

Eqs. (4) and (5), and Eqs. (11) through (13) will be utilized in the longitudinal simulation. The program finds V_r from Eq. (1) and uses Eq. (4) to determine if the trajectory of aircraft A is inside the collision cone. If both $V_r < 0$ and $y < w$, then the program applies the guidance law Eqs. (11) - (13), until either y is larger than w (as described just under Eq. (5)) or $V_r > 0$, meaning that collision will no longer occur and the trajectory is outside the collision cone.

Nonlinear Dynamics

First, the aircraft states are defined to be V_T = True Airspeed, α = Angle of Attack, θ = Pitch Attitude Angle, Q = Pitch Rate, and δ_e = Elevator Deflection. These are all in reference to craft A, since that craft will be the focus of this simulation. Velocity V_T, angle of attack α, and pitch attitude angle θ are shown in Fig. (6) below, along with the flight path angle γ_A. The red line is an extension from the nose of the craft, to better show the direction it is facing.

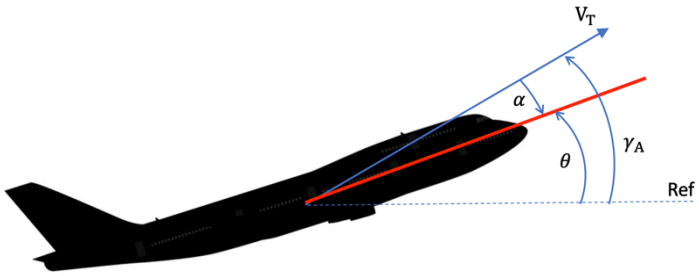

Figure 6: Longitudinal Dynamics State Variables

Arrow direction indicates positive notation. From Fig. (6), the flight path angle γ_A can be seen to equal $\theta - \alpha$.

Using these states, the decoupled longitudinal equations of motion are derived in Ref. [17] as

$$\begin{aligned} \dot{V}_T &= [F_T \cos\alpha - D - mg\sin\gamma]/m \\ \dot{\alpha} &= [-F_T \sin\alpha - L + mg\cos\gamma]/(mV_T) + Q \\ \dot{\theta} &= Q \\ \dot{Q} &= M/I_{yy} \\ \dot{\delta}_e &= -(1/\tau)\delta_e + (1/\tau)\delta_{ec} \end{aligned} \tag{14}$$

The actual elevator deflection and the commanded elevator deflection are δ_e and δ_{ec}, respectively, and the above equations also include first-order elevator actuator dynamics. All of the other ancillary terms are defined as

$$\begin{aligned} L &= \bar{q}SC_L \\ C_L &= 0.25 + 4.58\alpha \\ D &= \bar{q}SC_D \\ C_D &= 0.038 + 0.053C_L^2 \\ F_T &= (338.02 + 1.56V_T - 0.008V_T^2) \\ M &= \bar{q}Sc[C_M + 0.5(c/V_T)C_{mQ}Q] \\ C_M &= 0.015 - 0.75\alpha - 0.9\delta_e \\ C_{MQ} &= -12 \\ \frac{1}{\tau} &= 20.2 \end{aligned} \tag{15}$$

Where $\bar{q} = 0.5\rho {V_T}^2$, ρ being the air density (assumed to be 0.002377 slugs/ft^3). To rest of the terms required for Eqs. (14) and (15) are dependent on the aircraft used, which is described in the next section.

Zivko 33% Scale Edge 540T

The specific aircraft used in all of the following simulations is the 33% scale version of the Zivko Edge 540T, made by Aeroworks, and seen below in Fig. (7).

Figure 7: Zivko 33% Scale Edge 540T by Aeroworks, From Ref. [17]

This is a relatively small and aerobatic airplane. Information regarding the full scale Zivko Edge 540T, including properties like length, wingspan, etc., can be found in Ref. [25], while parameters of this specific 33% Zivko Scale Edge 540T aircraft are defined in Ref. [17] to be: weight = m*g = 29 lb., y-axis moment of inertial I_{yy} = 1.5 slug*ft^2, and wing area S = 13 ft^2. Wing chord length is estimated as being 33% of the wing chord length of the full-scale aircraft from Ref. [7], and is found to be c = 1.41 ft. These are the rest of the terms required to fully define Eqs. (14) and (15).

Control Law Derivation

We now use dynamic inversion to design a control law that will convert the commanded acceleration determined by the guidance law into a corresponding elevator deflection of the aircraft. In a very similar process to the derivation of the guidance law Eq. (11), the control law starts with the definition of a new y term,

$$y_{zp} = n_{zp} = \frac{\bar{q}S(C_L\cos(\alpha)+C_D\sin(\alpha))}{mg} + \frac{15M}{gI_{yy}} \tag{16}$$

In which n_{yp} is the acceleration at the pilot's seat of the aircraft, and is also derived in Ref. [17]. Following the same process which yielded Eq. (7), the following is found. Define the error term as e = $a_{Lat,C} - n_{zp}$, and differentiate the same to obtain

$$\dot{e} = \dot{y}_{zp} = \frac{\partial y}{\partial V_T}\dot{V}_T + \frac{\partial y}{\partial \alpha}\dot{\alpha} + \frac{\partial y}{\partial Q}\dot{Q} + \frac{\partial y}{\partial \delta_e}\dot{\delta}_e \tag{17}$$

where we have neglected the influence of the derivative of $a_{Lat,C}$. Now, the error term e should follow the dynamics $\dot{e} = -k_c * e$, where $k_c > 0$ is a controller gain. Substituting Eq. (10) into Eq. (17),

$$e = \frac{-1}{k_c}\left(\frac{\partial y}{\partial V_T}\dot{V}_T + \frac{\partial y}{\partial \alpha}\dot{\alpha} + \frac{\partial y}{\partial Q}\dot{Q} + \frac{\partial y}{\partial \delta_e}\dot{\delta}_e\right) \tag{18}$$

Finally, substituting Eqs. (14) into Eq. (18), and solving for δ_{ec} yields

$$\delta_{ec} = \frac{\tau\left(k_c e - \frac{\partial y}{\partial V_T}\dot{V}_T - \frac{\partial y}{\partial \alpha}\dot{\alpha} - \frac{\partial y}{\partial Q}\dot{Q} - \frac{\partial y}{\partial \delta_e}\frac{1}{\tau}\delta_e\right)}{\frac{\partial y}{\partial \delta_e}} \tag{19}$$

This Eq. (19) is the control law, where the partial derivative terms (also from Ref. [17]) are

$$\frac{\partial y}{\partial V_T} = \frac{\rho V_T S}{mg}(C_L \cos(\alpha) + C_D \sin(\alpha)) + \frac{15}{gI_{yy}}\left(\rho V_T S c C_M + 0.25 S c^2 Q \rho C_{MQ}\right)$$

$$\frac{\partial y}{\partial \alpha} = \frac{\bar{q}S}{mg}((C_L + 4.58)\cos(\alpha) - 0.515 C_L \sin(\alpha)) - \frac{15}{gI_{yy}}(0.75\bar{q}Sc) \tag{20}$$

$$\frac{\partial y}{\partial Q} = \frac{15}{gI_{yy}}\frac{Sc^2\bar{q}C_{MQ}}{2V_T}$$

$$\frac{\partial y}{\partial \delta_e} = -\frac{15}{gI_{yy}}(0.9\bar{q}Sc)$$

Thus, all of the components of the longitudinal simulation are defined, and the Simulink model can now be developed.

Simulink Model

In the model seen in Fig. (8), for each time step, the simulation generates a latax in the guidance block from the guidance law Eq. (11). The error is then found to be difference of the commanded and the actual latax, and is input into the controller block. This is fed into the control law Eq. (19), which outputs a commanded elevator deflection. This δ_{ec} is the input of the nonlinear equations of motion Eqs. (14) in the dynamics block, which then outputs the states. These states are then fed back into each of the previous blocks, as well as used to calculate the acceleration at the pilot's seat n_{zp}, which is Eq. (16). The dynamics block also updates the position of the aircraft by solving the following Eqs. (21),

$$\begin{aligned} \dot{x}_A &= V_T \cos(\gamma_A) \\ \dot{y}_A &= V_T \sin(\gamma_A) \\ \dot{x}_B &= V_B \cos(\gamma_B) \\ \dot{y}_B &= V_B \sin(\gamma_B) \end{aligned} \tag{21}$$

Where V_B is the speed of aircraft B and γ_B is the angle of attack of craft B. The integral of these Eqs. (21) are then taken in Simulink, which yields the new inertial frame positions of each aircraft, (x_A, y_A) and (x_B, y_B). The results of this simulation are discussed in Section 3b of this thesis.

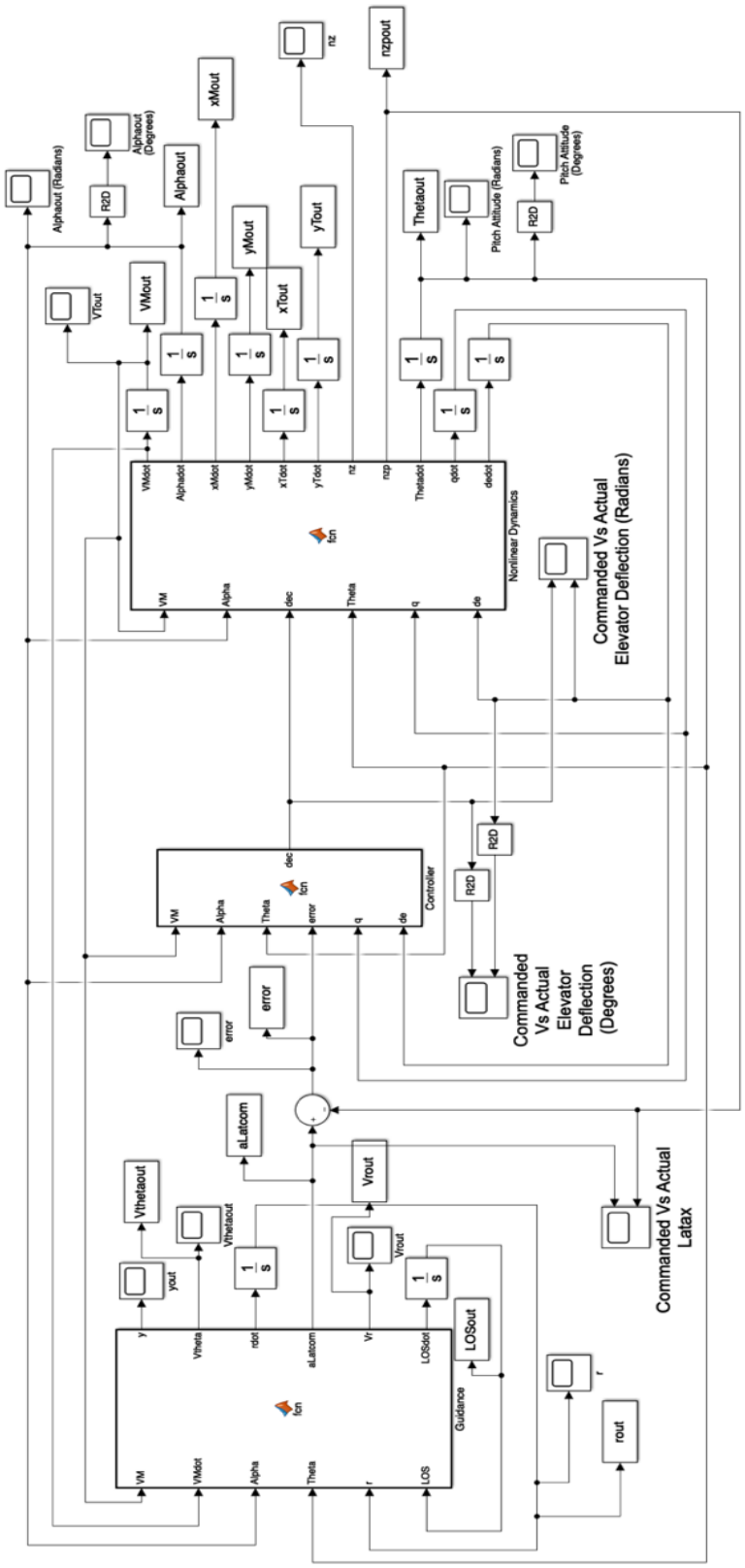

Figure 8: Longitudinal Simulink Model

b. SIMULATION RESULTS

Initial Collision Scenario

Aircraft A is assumed to be at trim initially, and so the equilibrium values of these equations of motion must be found. Also, to ensure that the craft is flying at the desired orientation of being level while at trim, the γ angle must initially equal zero, meaning that the condition of α and θ equaling each other must be met. Therefore, the Matlab function 'fmincon' is used with the above condition, which yields the following initial trim values:

$V_T = 289.7165$ (ft/s)
$\alpha = -2.5751$ (deg)
$\theta = -2.5751$ (deg)
$Q = 0$ (deg/s)
$\delta_e = 3.1008$ (deg)

Fig. (9) below details the geometry and initial conditions of the scenario used in the longitudinal simulation. Many specifics of this scenario were chosen to ensure that the level flying craft A is on a collision course with craft B, which occurs at the red X mark. Craft A is labelled "A" at the origin of the inertial frame, while craft B is at point B. The circles around craft A and craft B are chosen to both have a radius of $R_A = R_B = 15$ ft. The aircraft are initially a distance of r = 1500 ft. apart, with a line of sight angle θ_{LOS} = - 20 degrees. V_T and V_B are the true velocity vectors of craft A and B, respectively, and γ_A and γ_B are the angles from horizontal reference at which the aircraft velocity vectors act, known as the flight path angles. The velocity of craft A is previously defined as 289.7165 ft/sec, while the other craft is slower by an arbitrary factor of 1.2, resulting in a speed of 241.4304 ft/sec.

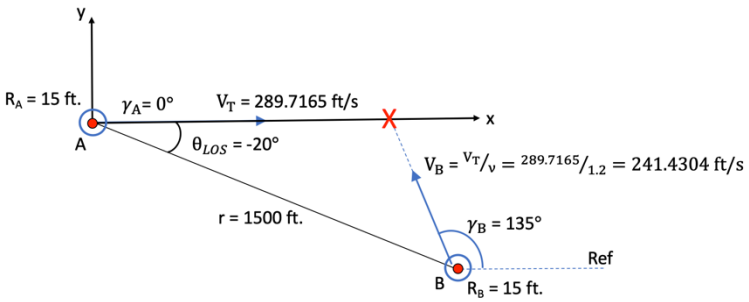

Figure 9: Initial Longitudinal Collision Scenario

Running the longitudinal simulation with a constant commanded acceleration of 1 g, which is the latax required to maintain level flight, it is validated in Fig. (10) below that craft A and B do indeed collide using this initial scenario. A closeup is provided in Fig. (11). The simulation run time is 3.055 seconds.

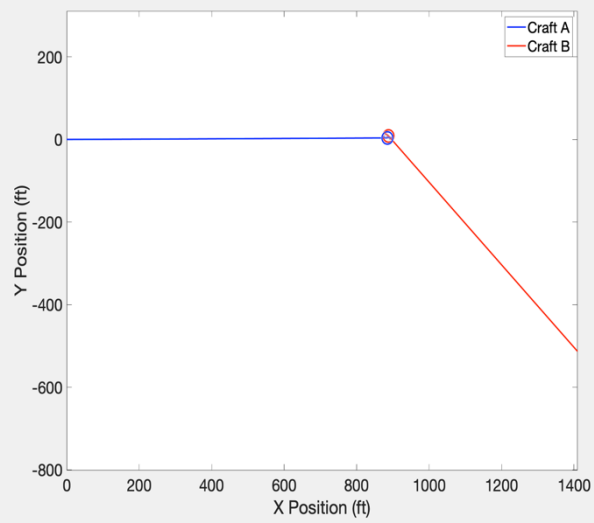

Figure 10: Initial Longitudinal Collision Scenario in Simulation

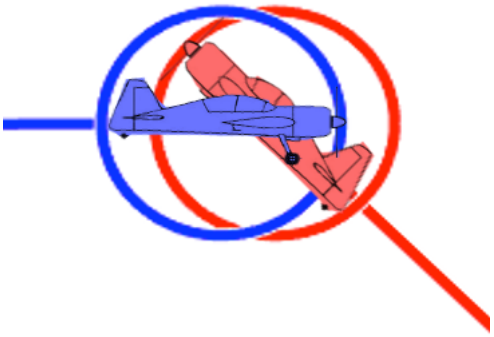

Figure 11: Collision Closeup

The path of craft A is represented by the blue line, with the aircraft initially at the origin and the craft itself being the blue circle. The path of craft B is represented by the red line, with the craft itself being the red circle. Craft B begins at the bottom right of the plot, and clearly ends the simulation overlapping with the blue circle, indicating that the craft will collide if craft A does not change its initial flight path. As Fig. (11) shows, inside of each circle is the longitudinal view of a Zivko aircraft from Ref. [1].

In accordance with collision cone theory, the y value from Eq. (4) is found to initially be y = -819.1297 ft^2, while the initial V_r value from Eqs. (1) is initially V_r = -491.0547 ft/sec. As the value of y is less than w = 0, and since the value of V_r is negative, the criteria for being inside the collision cone are met, confirming that craft A begins and remains on a collision course with craft B.

Simulation With Applied Latax Required for Collision Avoidance

Now, the simulation is rerun with the commanded latax applied to craft A that is required for it to just barely avoid collision with craft B. The distance factor w from the guidance law Eq. (11) is set to w = 0 for a near miss, the guidance gain k_g = 3, and the controller gain k_c = 25. The simulation run time is still 3.055 seconds. The resulting trajectory in the inertial frame is seen below in Fig. (12), and a zoomed in plot of the near miss is provided in Fig. (13) for convenience.

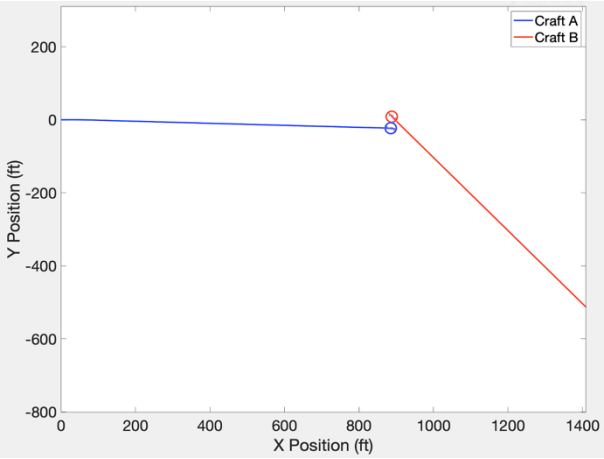

Figure 12: Longitudinal Simulation, Near Miss

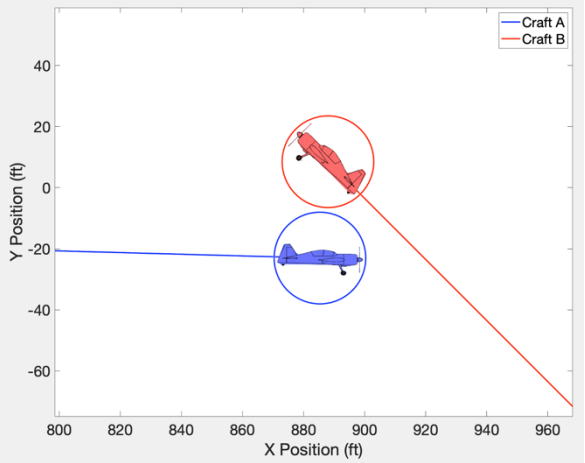

Figure 13: Longitudinal Simulation, Near Miss Closeup

The closest that the circle around aircraft A gets to the circle around craft B is 4.8124 inches. The commanded and actual lateral acceleration in g's required to achieve this is shown in Fig. (14) below.

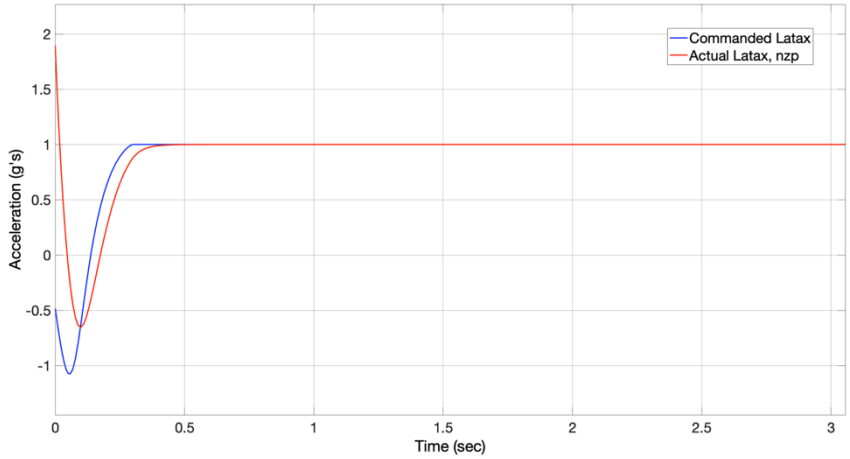

Figure 14: Commanded and Actual Latax

Since an applied acceleration of 1 g is required to keep craft A flying level, any acceleration less than 1 g drops the aircraft down in the inertial frame, which is what happens in this case.

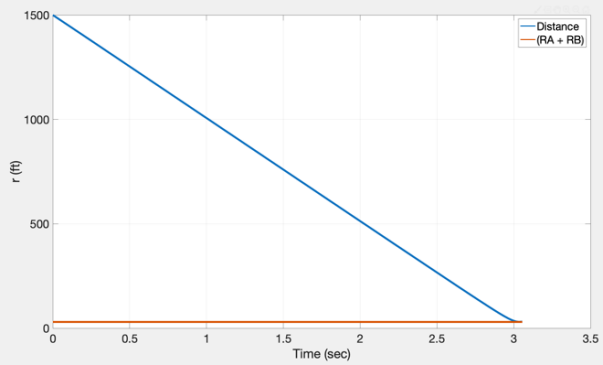

Figure 15: Distance Between Aircraft

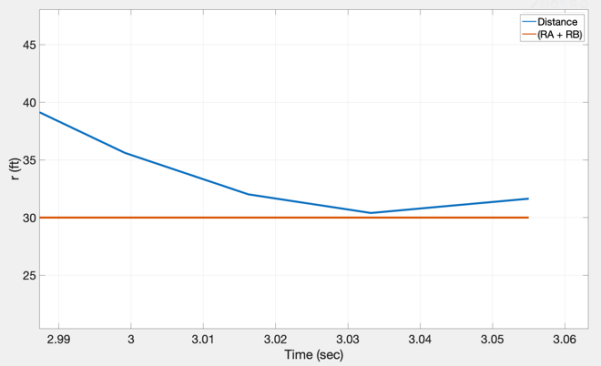

Figure 16: Distance Between Aircraft, Close Up

In Fig. (15), the blue line shows the distance r between the two aircraft throughout the simulation, and Fig. (16) is a close up showing that the distance between the craft does not go below the orange line representing the radii (R_A + R_B), meaning that the craft do not collide.

The following Fig. (17) is the flight path angle of aircraft A, γ_A, in degrees. The negative values show that the craft angles downward in the inertial frame.

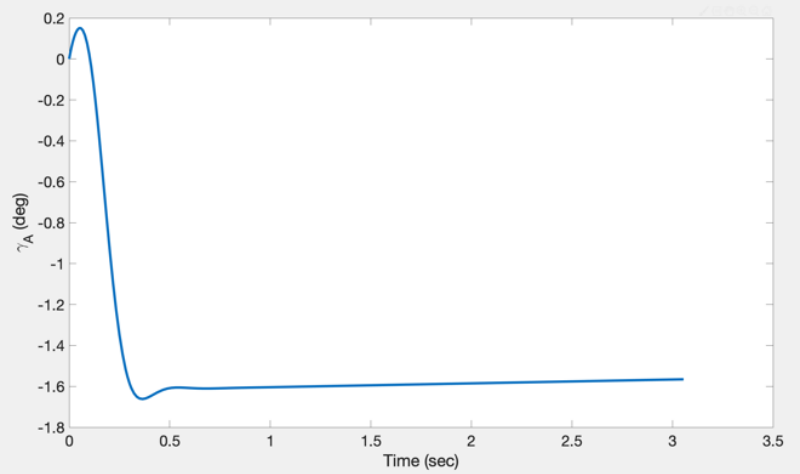

Figure 17: γ_A, Flight Path Angle of Aircraft A

The y value from the guidance block, Eq. (4), is shown in Fig. (18).

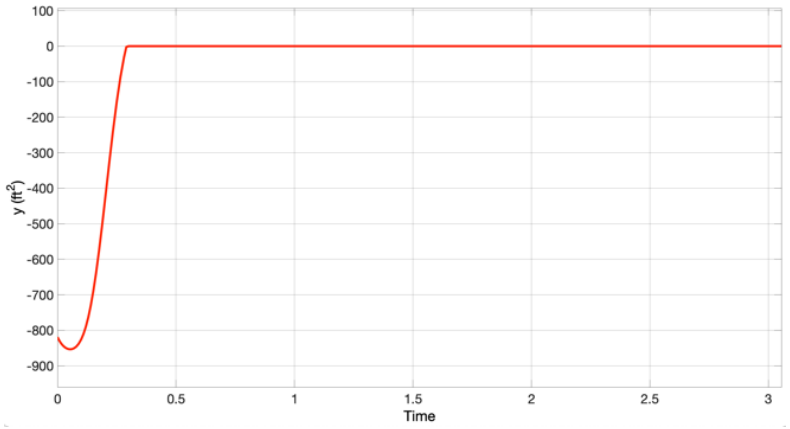

Figure 18: Guidance Block Value y

Note that all y values above w = 0 have been set to equal w. The y values later in the simulation can get positively large; however, the only y values we are interested in are when y is less than w, since that is when craft A is on a collision course with craft B. So, all y values above y = w have been set equal to w to cut down on unnecessary information and to better show the y values below w. Time on the x axis is plotted in seconds.

The plots of V_θ and V_r separately plotted each with respect to time (in seconds) are Fig. (19) and Fig. (20), respectively.

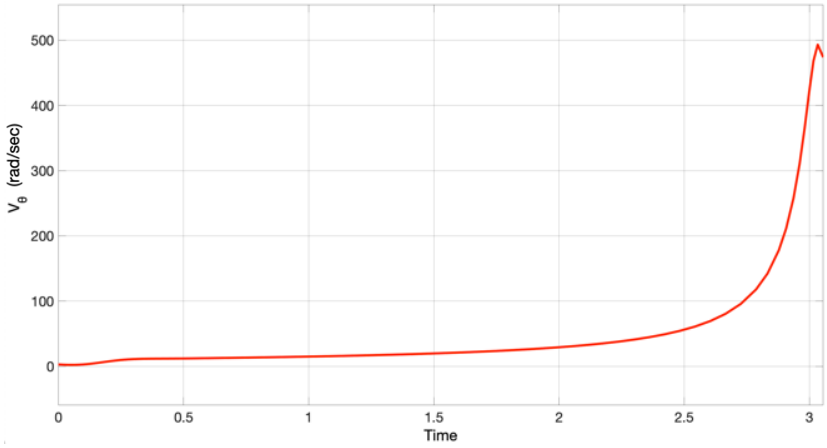

Figure 19: Plot of V_θ

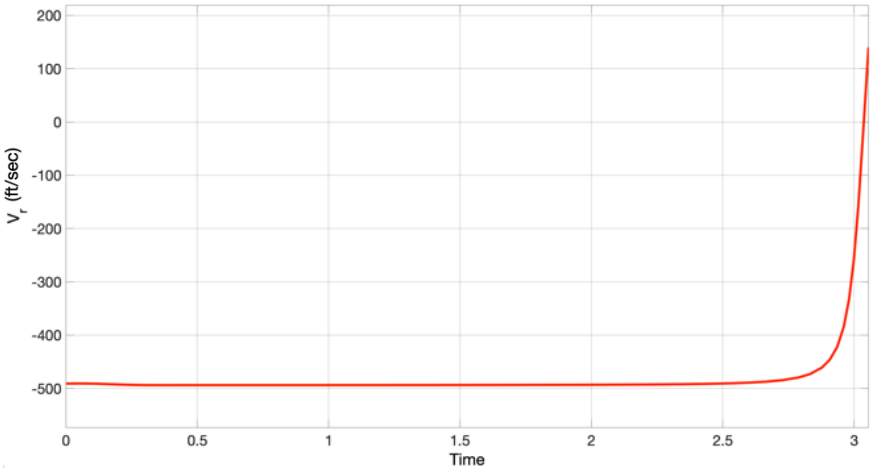

Figure 20: Plot of V_r

Recall that the guidance law applies the latax to craft A when both y < w and $V_r < 0$. From Figs. (18) and (20) above, it is clear that the y value is the main determining factor regarding the latax application, as it becomes larger than w much sooner than V_r becomes positive.

Finally, the plots of the state variables versus simulation time in seconds are seen in the following figures, starting with the velocity of craft A in Fig. (21).

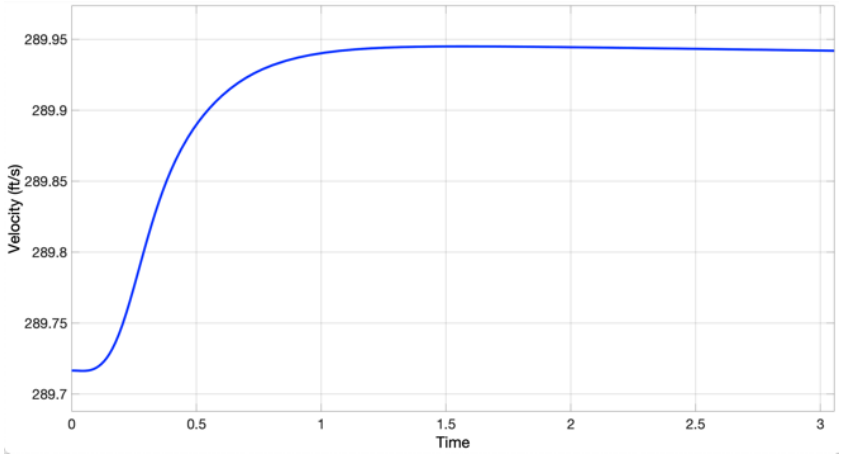

Figure 21: True Velocity of Craft A

Since craft A tilts downward slightly, moving with the direction of the acceleration due to gravity, it gains a slight bit of speed at around 0.21 ft/sec.

Fig. (22) is the both the commanded elevator deflection δ_{ec} and the actual elevator deflection δ_e in degrees for craft A.

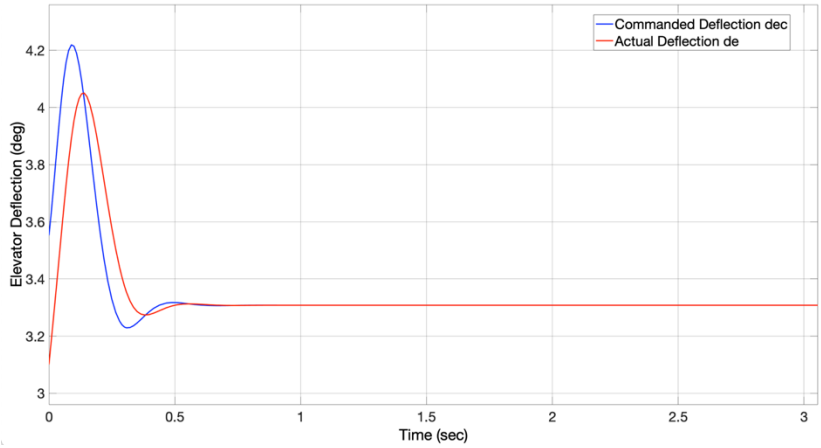

Figure 22: Commanded Elevator Deflection δ_{ec} and Actual Elevator Deflection δ_e

Fig. (23) below is the angle of attack α, in degrees, of craft A throughout the simulation.

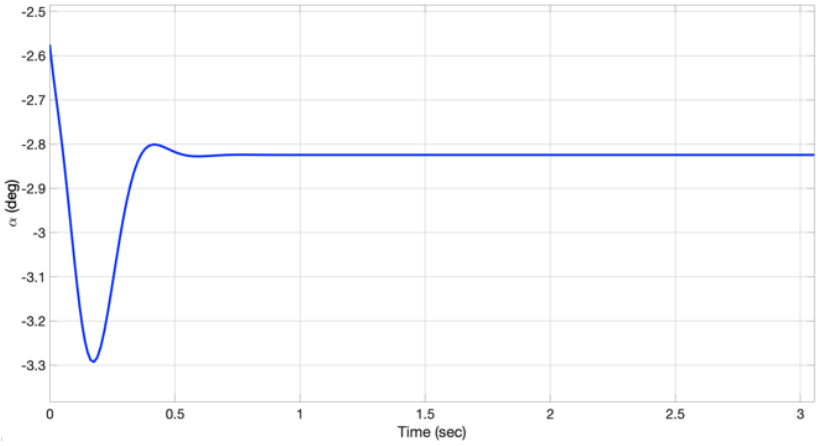

Figure 23: Aircraft A Angle of Attack α

Next is the pitch attitude angle θ, also in degrees, in Fig. (24).

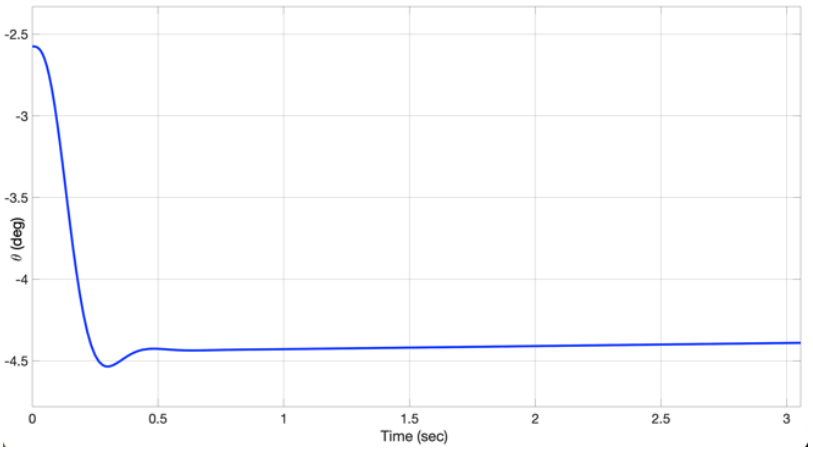

Figure 24: Aircraft A Pitch Attitude Angle θ

Finally, the last state variable to look at is the pitch rate Q, in deg/sec, seen in Fig. (25).

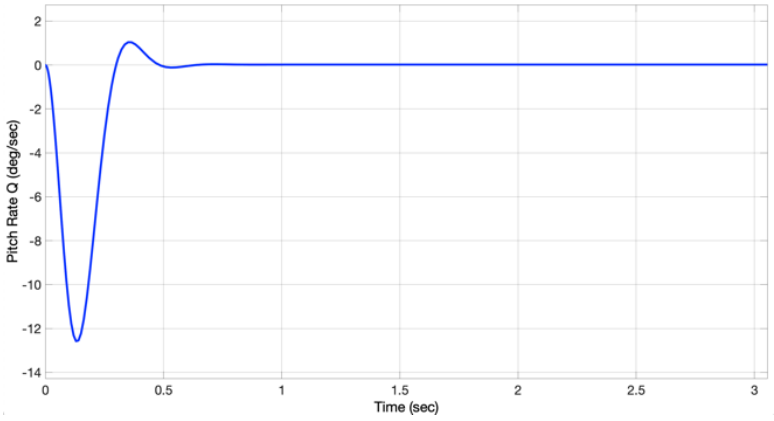

Figure 25: Aircraft A Pitch Rate Q

Simulation Two, Near Miss from Closer Initial Distance

In order to see what the program does when a larger change in initial trajectory is required for aircraft A, the same simulation is run again, except with the initial distance between the two craft r = 600 ft. instead of the previous r = 1500 ft. Every other initial value is the same as the previous simulation, as dictated in Fig. (9). With a commanded acceleration of 1 g so that craft A maintains a level flight path, Fig. (26) below shows the aircraft still collide as before, except now much sooner in the simulation at a runtime of 1.21 seconds.

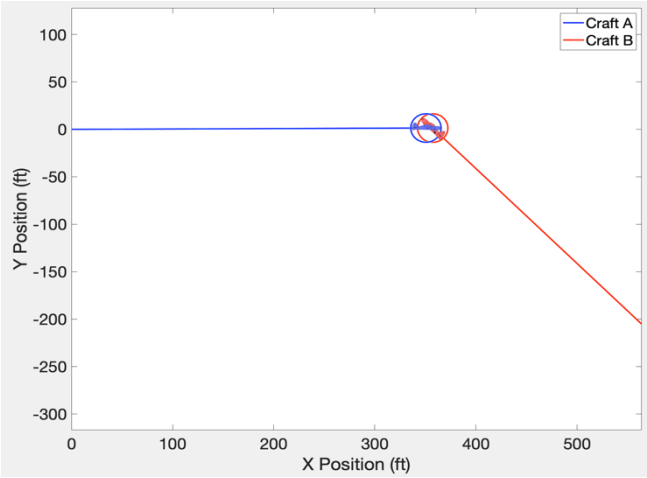

Figure 26: Longitudinal Simulation Two with Closer Initial Distance, Collision

Exactly as under Fig. (10), the path of craft A is still represented by the blue line, with the aircraft initially at the origin and the craft itself being the blue circle. The path of craft B is represented by the red line, with the craft itself being the red circle. It begins at the bottom right of the plot, and clearly ends the simulation overlapping with the right end of the blue line, indicating that the craft will collide if craft A does not change its initial flight path. With an applied acceleration required to avoid this collision, the trajectory now looks like Fig. (27). The distance factor w from the guidance law Eq. (11) is still set to w = 0 for a near miss, the guidance gain is now k_g = 2.05, and the controller gain is still k_c = 25. The simulation run time is still 1.21 seconds.

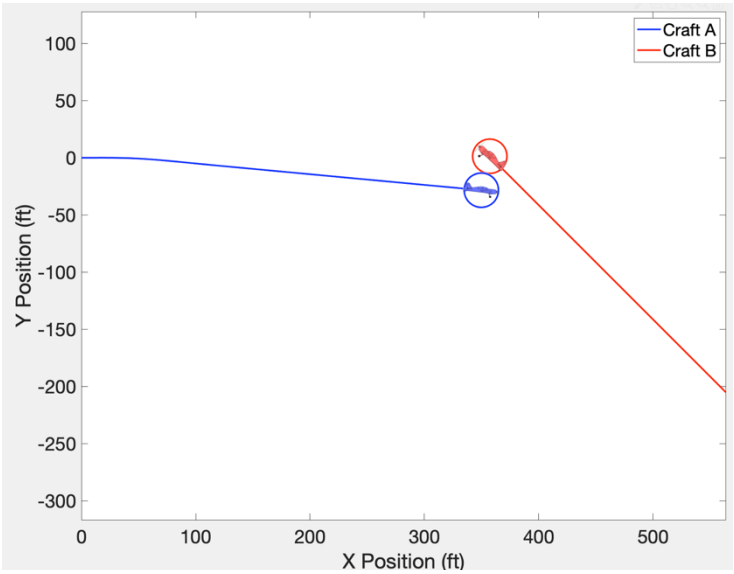

Figure 27: Longitudinal Simulation Two from Closer Initial Distance, Near Miss

The closest that aircraft A gets to craft B is 5.6895 inches. The commanded and actual lateral acceleration in g's required to achieve this is shown in Fig. (28) below.

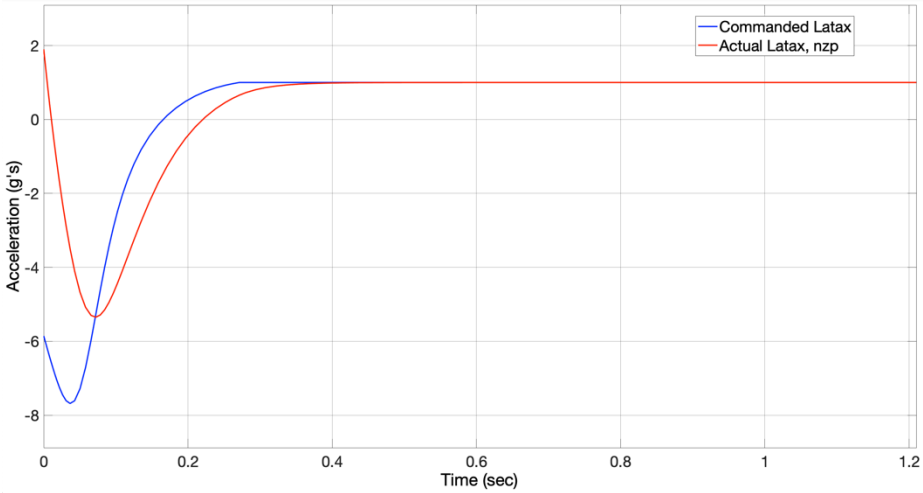

Figure 28: Simulation Two Commanded and Actual Latax

Comparing this to Fig. (15), is can be seen that reducing the initial distance between the two craft results in much larger requirements on acceleration in order for craft A to avoid collision with craft B.

Corroborating this result, the plot of the distance between the two craft is below in Fig. (29), with another closeup provided in Fig. (30) for convenience.

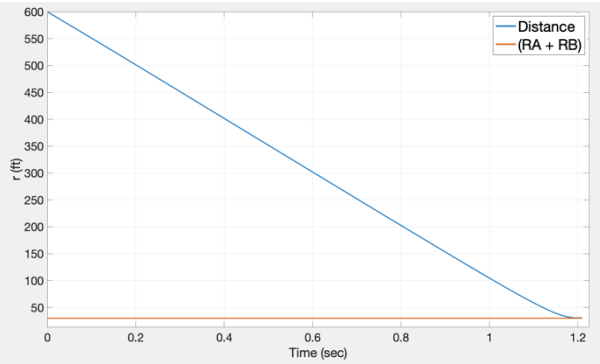

Figure 29: Distance Between the Two Craft, Simulation Two

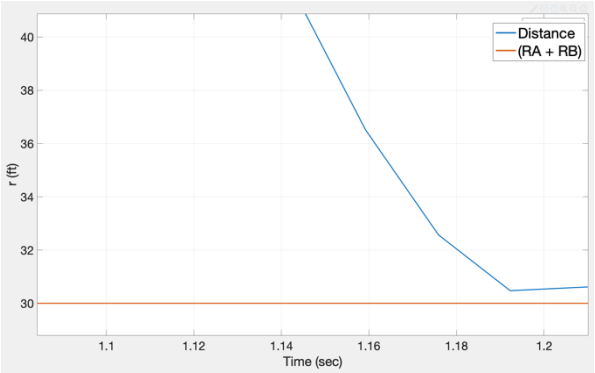

Figure 30: Closeup of Figure 29

The following Fig. (31) is the flight path angle of aircraft A, γ_A, in degrees.

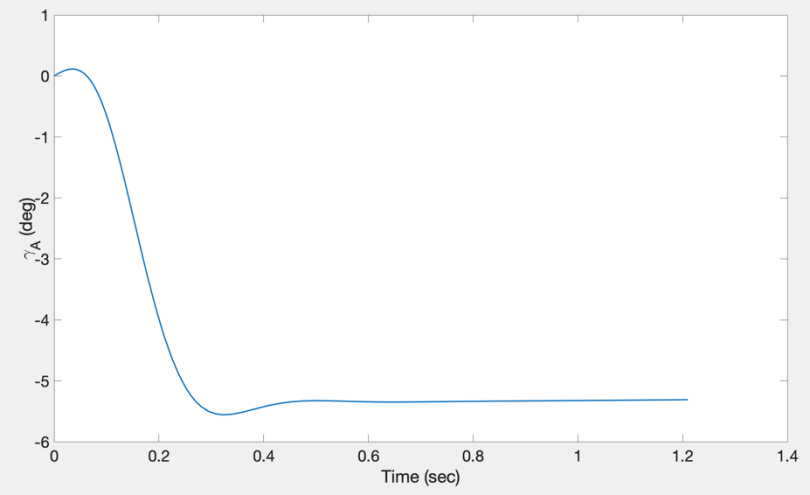

Figure 31: γ_A, Flight Path Angle of Aircraft A for Simulation Two

The y value from the guidance block, Eq. (4), is shown in Fig. (32).

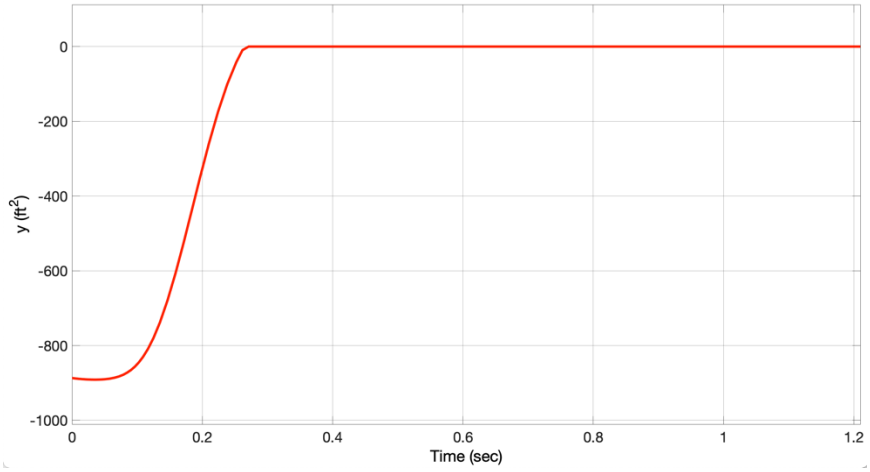

Figure 32: Guidance Block Value y for Simulation Two

Note again that all y values above w = 0 have been set to equal w. The y values later in the simulation can get positively large; however, the only y values we are interested in are when y is less than w, since that is when craft A is on a collision course with craft B. So, all y values above y = w have been set equal to w to cut down on unnecessary information and to better show the y values below w.

The plots of V_θ and V_r separately plotted each with respect to time are Fig. (33) and Fig. (34), respectively.

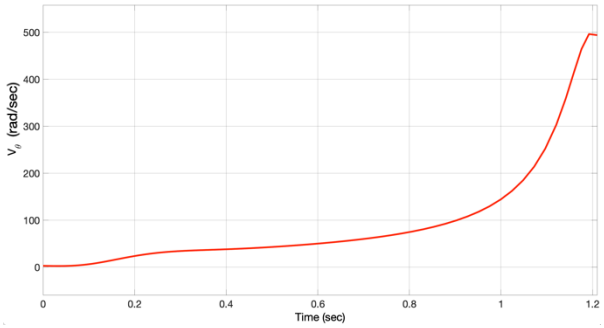

Figure 33: Plot of V_θ for Simulation Two

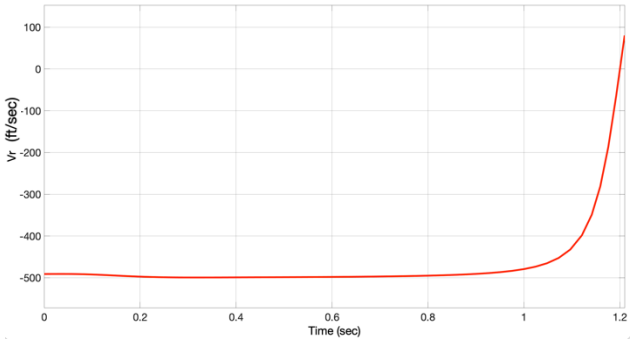

Figure 34: Plot of V_r for Simulation Two

Much like the previous simulation, looking at Figs. (32) and (34) shows that the driving factor of latax application conditions is again the y value.

Finally, the plots of the state variables versus simulation time in seconds are seen in the following figures, starting with the velocity of craft A in Fig. (35).

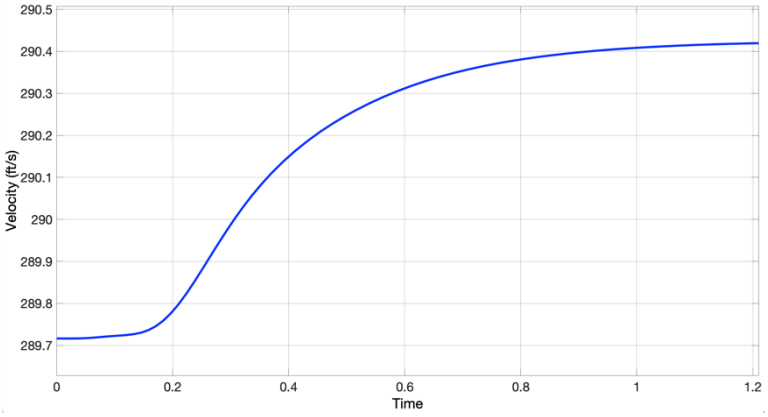

Figure 35: True Velocity of Craft A for Simulation Two

Fig. (36) below is the both the commanded elevator deflection δ_{ec} and the actual elevator deflection δ_e in degrees for craft A. In the legend, δ_{ec} is denoted as dec, and δ_e is denoted as de. In the legend, δ_{ec} is again denoted as dec, and δ_e is again denoted as de.

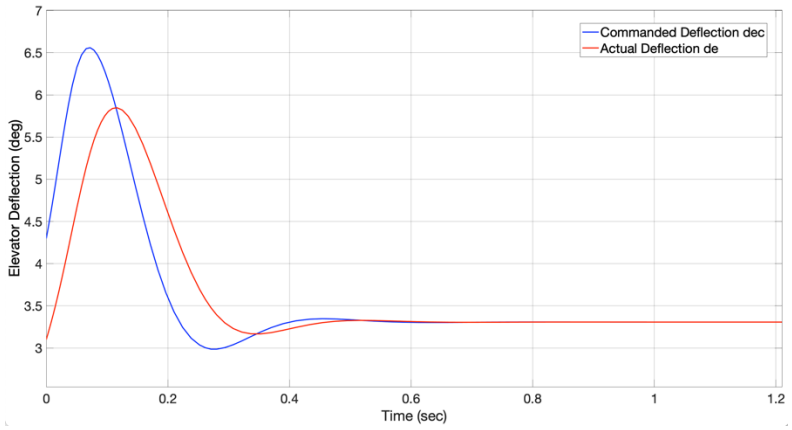

Figure 36: Commanded Elevator Deflection δ_{ec} and Actual Elevator Deflection δ_e for Simulation Two

Fig. (37) below is the angle of attack α, in degrees, of craft A throughout the simulation.

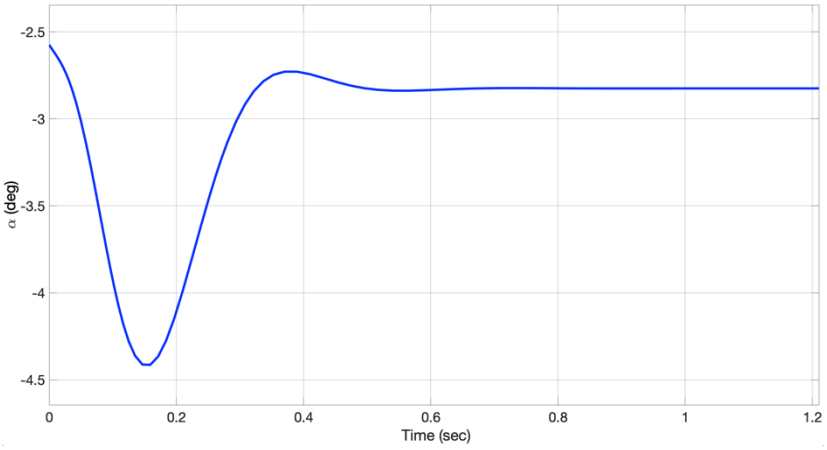

Figure 37: Aircraft A Angle of Attack α for Simulation Two

Next is the pitch attitude angle θ, also in degrees, in Fig. (38).

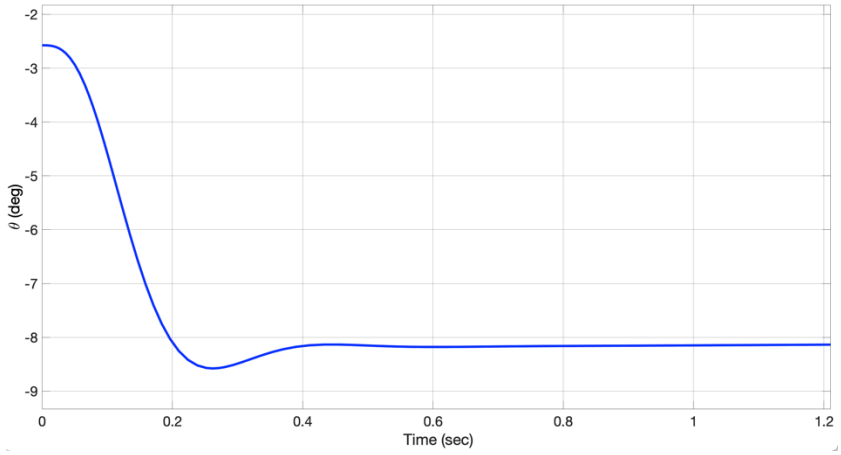

Figure 38: Aircraft A Pitch Attitude Angle θ for Simulation Two

Finally, the last state variable to look at is the pitch rate Q, in deg/sec, seen in Fig. (39).

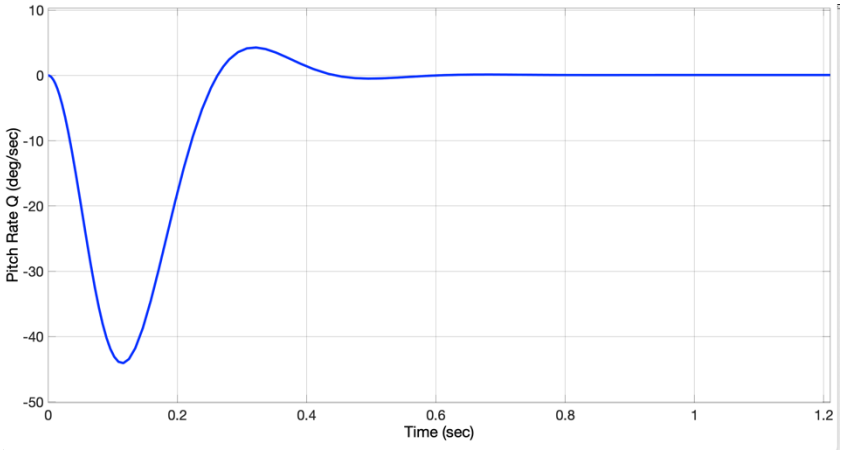

Figure 39: Aircraft A Pitch Rate Q for Simulation Two

4. COLLISION AVOIDANCE DESIGN IN THE LATERAL DIRECTIONAL AXIS

a. APPROACH

This chapter has a very similar structure to the previous chapter, beginning with the derivation guidance law, which again synthesizes the desired acceleration required for collision avoidance. Then, the control law is designed, which converts this commanded acceleration into the corresponding deflections of the control surfaces, which in this case, are the aircraft's aileron and rudder. These laws are both integrated into a Simulink model, along with the aircraft nonlinear dynamics. Finally, simulation results are presented and discussed.

Overall Guidance and Control Architecture

The following Fig. (40) shows the logic of how the architecture for the simulations in the lateral-directional axis are designed.

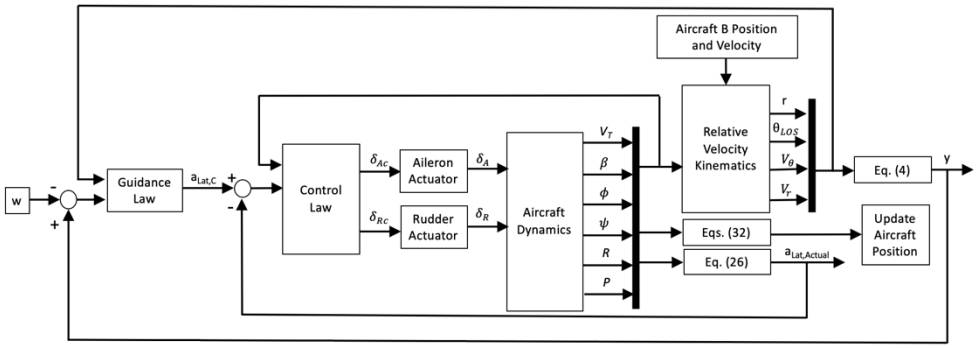

Figure 40: Lateral-Directional Axis Guidance and Control Architecture

The reasoning used here is very similar to that which is used for the longitudinal version in Fig. (6). Once again, the guidance law generates an output of the commanded lateral acceleration, which is then compared to the actual lateral acceleration. The difference of these two terms is used as the input to the control law, along with the state values from the aircraft dynamics. This control law outputs a commanded aileron deflection and a commanded rudder deflection. The aileron actuator converts this commanded aileron deflection value into an actual aileron deflection, and the rudder actuator does the same for the rudder deflection terms. These actual deflection values are then used as the input into the aircraft dynamics. The dynamics equations result in an output of the state values, which are fed into the relative velocity kinematics, as well as the equation to generate the actual aircraft lateral acceleration, and the equations that update the aircraft position. The relative velocity kinematics are also used to find a term y, which

is based on miss distance arguments, and is then compared to an input term w and fed into the guidance law along with the relative velocity kinematics. Once again, all of these variables, equations, and laws are derived or defined in the following sections.

Guidance Law Derivation

The lateral axis guidance law is very similar to the longitudinal guidance law, except with changes to certain angles used in certain equations. Eqs. (4) through (7) are still the same for this derivation, while Eqs. (8) are replaced with Eqs. (22) below.

$$\dot{r} = V_r$$

$$\dot{V}_r = \frac{{V_\theta}^2}{r} - a_{Long}\cos(\kappa_A - \theta_{LOS}) - a_{Lat,C}\cos(\delta_A - \theta_{LOS}) \tag{22}$$

$$\dot{V}_\theta = -\frac{V_\theta V_r}{r} - a_{Long}\sin(\kappa_A - \theta_{LOS}) - a_{Lat,C}\sin(\delta_A - \theta_{LOS})$$

Where the flight path angle of craft A in the lateral axis is kappa κ_A, and δ_A now equals $\delta_A = \kappa_A + {}^{\pi}/_{2}$. The V_θ and V_r terms are the same as in Eq. (4), except now the flight path angle for each craft is changed from γ_A and γ_B to κ_A and κ_B, respectively. The θ_{LOS} term is still the line of sight angle, only now in the lateral frame. Following the same process as before, plugging Eqs. (22) into Eq. (7), with terms without $a_{Lat,C}$ or a_{Long} cancelling out, results in

$$\dot{e} = \dot{y} = \frac{\partial y}{\partial V_r}\left(-a_{Long}\cos(\kappa_A - \theta_{LOS}) - a_{Lat,C}\cos(\delta_A - \theta_{LOS})\right)$$

$$+\frac{\partial y}{\partial V_\theta}\left(-a_{Long}\sin(\kappa_A - \theta_{LOS}) - a_{Lat,C}\sin(\delta_A - \theta_{LOS})\right) \tag{23}$$

In order to drive y to w, the e term from Eq. (5) must be driven to 0 using the same process as before. To this end, Eq. (10) is once again used. Substituting Eq. (10) into Eq. (23), and solving for $a_{Lat,C}$,

$$a_{Lat,C} = \frac{k_g(y-w) + a_{Long}\left(\frac{\partial y}{\partial V_r}\cos(\kappa_A - \theta_{LOS}) + \frac{\partial y}{\partial V_\theta}\sin(\kappa_A - \theta_{LOS})\right)}{\frac{\partial y}{\partial V_\theta}sin(\delta_A - \theta_{LOS}) + \frac{\partial y}{\partial V_r}cos\,(\delta_A - \theta_{LOS})} \tag{24}$$

This Eq. (24) is the lateral guidance law, with Eqs. (12) and (13) still applying.

Equations (4), (24), (12), and (13) will be utilized in the lateral simulation. The program again uses Eqs. (1) and (4) to determine if aircraft A and B will collide, and if they will, then it applies the guidance law Eq. (24), with (12) and (13), until the trajectory is outside the collision cone. Once y is larger than w or V_r is larger than 0, the commanded acceleration is returned to the baseline value of zero g's.

Nonlinear Dynamics

The lateral states are defined to be V_T = True Airspeed, β = Sideslip Angle, ϕ = Roll Angle, ψ = Yaw Angle, P = Roll Rate, R = Yaw Rate, δ_A = Aileron Deflection, and δ_R = Rudder Deflection. Velocity V_T, sideslip angle β, and yaw angle ψ are shown in Fig. (41) below, along with the directional flight path angle κ_A.

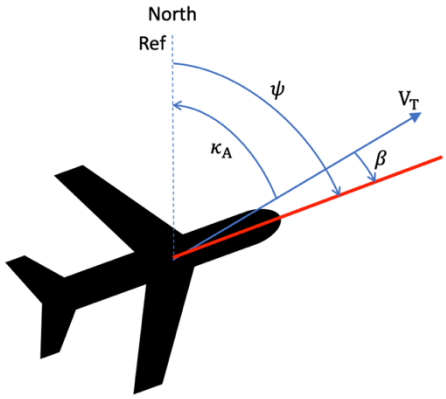

Figure 41: Lateral-Directional Dynamics State Variables

Like in Fig. (6), the red line is an extension from the nose of the craft, to better show the direction it is facing, and arrow direction indicates positive notation. It can therefore be seen that the directional axis flight path angle $\kappa_A = -(\psi - \beta) = \beta - \psi$.

Using these state variables, the lateral equations of motion are derived in Refs. [17] and [6] as

$$\dot{V}_T = (F_T \cos(\alpha) - D - mgsin(\gamma))/m$$

$$\dot{\phi} = P + \tan(\theta)(Qsin(\emptyset) + Rcos(\emptyset))$$

$$\dot{\psi} = (Qsin(\emptyset) + Rcos(\emptyset))/\cos(\theta) \tag{25}$$

$$\dot{\delta}_A = -(1/\tau)\delta_A + (1/\tau)\delta_{AC}$$

$$\dot{\delta}_R = -(1/\tau)\delta_R + (1/\tau)\delta_{RC}$$

$$\dot{\beta} = \frac{g}{V_T}\sin(\emptyset)\cos(\theta) - Rcos(\alpha) + Psin(\alpha) + \frac{\bar{q}S}{mV_T}C_{y_\beta}\beta$$

$$\dot{R} = \frac{I_{XX} - I_{YY}}{I_{ZZ}}PQ + \frac{I_{XZ}}{I_{ZZ}}\dot{P} - \frac{I_{XZ}}{I_{ZZ}}QR + \frac{n}{I_{ZZ}} + \frac{\bar{q}Sb^2}{2V_T I_{ZZ}}(C_{n_R}R + C_{n_P}P)$$

$$\dot{P} = \frac{I_{YY} - I_{ZZ}}{I_{XX}}QR + \frac{I_{XZ}}{I_{XX}}\dot{R} + \frac{I_{XZ}}{I_{XX}}PQ + \frac{l}{I_{XX}} + \frac{\bar{q}Sb^2}{2V_T I_{XX}}(C_{l_R}R + C_{l_P}P)$$

The actual aileron deflection and the commanded aileron deflection are δ_A and δ_{AC}, respectively, while the actual rudder deflection and the commanded rudder deflection are δ_R and δ_{RC}, respectively. The other terms that have not previously been defined are: wingspan b = 8.61 ft., which is 33% of the wingspan of the full-size plane from Ref. [19], x-axis moment of inertia I_{XX} = 0.32 slug*ft^2, z-axis moment of inertia I_{ZZ} = 2.3 slug*ft^2, and cross-product of inertia I_{XZ} = 0.026 slug*ft^2, all from Ref. [17].

Other important terms in Eqs. (25) are the yawing moment *n*, C_{n_R}, and C_{n_P} terms from the $\dot{R}$ equation, the rolling moment *l,* C_{l_R}, and C_{l_P} terms from the $\dot{P}$ equation, and the C_{y_β} term from the $\dot{\beta}$ equation. These are all estimated from Ref. [15].

$$n = \bar{q}SbC_n\,, C_n = .12\beta - .011\delta_A - .09\delta_R$$

$$l = \bar{q}SbC_l\,, C_l = -.125\beta + .04\delta_A + .022\delta_R$$

$$C_{n_R} = -0.19,\ C_{n_P} = 0.02$$

$$C_{l_R} = 0.13,\ C_{l_P} = -0.4$$

$$C_{y_\beta} = -0.8$$

Control Law Derivation

Following similar logic as used in the previous derivations, a new y equation is defined from Ref. [18] to be the acceleration at the pilot's seat, $n_{yp.}$

$$y_{yp} = n_{yp} = n_y + \frac{X_p\dot{R} - Z_p\dot{P}}{g} \tag{26}$$

With n_y coming from Ref. [24].

$$n_y = (Dsin(\beta) - F_T\cos(\alpha)sin(\beta))/(mg) \tag{27}$$

The drag term D and thrust term F_T in Eq. (27) are the same as previously defined for Eqs. (15). In Eq. (26), the X_p term is the distance of the pilot's seat from the aircraft center of gravity along the x-axis. From figure (42) below of the Zivko aircraft from the side, it is observed that the pilot's seat is approximately at the same point on the x-axis as the center of the fuselage.

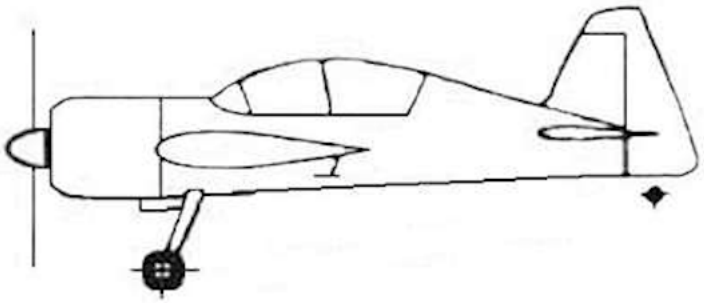

Figure 42: Side-View of Zivko 540T, From Ref. [1]

Therefore, it is estimated from Ref. [1] that X_p is equal to the distance of the effective center of the fuselage from the aircraft's center of gravity along the x-axis, so X_p = -1.5 ft. Similarly, Z_p is the distance of the pilot's seat from the aircraft center of gravity along the z-axis, and it is assumed that Z_p = 0 ft.

Once again, the error term used here is the difference between the commanded and actual accelerations, or $a_{Lat,C}$ from Eq. (24) minus n_{yp} from Eq. (26). Using the same dynamic inversion process regarding the error term as Eqs. (5), (6), and (7), it is found that

$$\dot{e} = \dot{y}_{yp} = \frac{dy}{dt} = \frac{\partial y}{\partial V_T}\frac{dV_T}{dt} + \frac{\partial y}{\partial \beta}\frac{d\beta}{dt} + \frac{\partial y}{\partial R}\frac{dR}{dt} + \frac{\partial y}{\partial P}\frac{dP}{dt} + \frac{\partial y}{\partial \delta_A}\frac{d\delta_A}{dt} + \frac{\partial y}{\partial \delta_R}\frac{d\delta_R}{dt}$$

$$= \frac{\partial y}{\partial V_T}\dot{V}_T + \frac{\partial y}{\partial \beta}\dot{\beta} + \frac{\partial y}{\partial R}\dot{R} + \frac{\partial y}{\partial P}\dot{P} + \frac{\partial y}{\partial \delta_A}\dot{\delta}_A + \frac{\partial y}{\partial \delta_R}\dot{\delta}_R \tag{28}$$

Manipulating Eq. (28) to isolate the commanded deflection terms δ_{AC} and δ_{RC} from the $\dot{\delta}_A$ and $\dot{\delta}_R$ terms on one side, and substituting the error term in, along with the controller gain k_c, the result is

$$\frac{\partial y}{\partial \delta_A}\delta_{AC} + \frac{\partial y}{\partial \delta_R}\delta_{RC} = -\tau\left(k_c e + \frac{\partial y}{\partial V_T}\dot{V}_T + \frac{\partial y}{\partial \beta}\dot{\beta} + \frac{\partial y}{\partial R}\dot{R} + \frac{\partial y}{\partial P}\dot{P} - \frac{1}{\tau}\frac{\partial y}{\partial \delta_A}\delta_A - \frac{1}{\tau}\frac{\partial y}{\partial \delta_R}\delta_R\right) \tag{29}$$

However, now there are two unknowns to solve for (δ_{AC} and δ_{RC}), yet only one Eq. (29). To solve this, δ_{RC} is set to equal some fraction of δ_{AC},

$$\delta_{RC} = k_{AR}\delta_{AC} \tag{30}$$

With k_{AR} being the ratio between the commanded aileron and rudder deflections. Setting the right-hand-side of Eq. (29) equal to a new term RHS, substituting in Eq. (30), then solving for δ_{AC}, Eq. (29) becomes

$$\delta_{AC} = \frac{RHS}{\frac{\partial y}{\partial \delta_A} + \frac{\partial y}{\partial \delta_R}k_{AR}} \tag{31}$$

Thus, Eq. (31) is the lateral control law used, and along with Eqs. (30) and the right-hand-side of Eq. (29), the desired commanded aileron and rudder deflections can be determined.

Simulink Model

The lateral Simulink model can be seen in Fig. (43). Similarly to the longitudinal Simulink model, for each time step, the lateral simulation generates a latax in the guidance block from the guidance law Eq. (24). The error is then found to be difference of the commanded and the actual latax, and is input into the controller block. This is fed into the control law Eq. (31), which outputs a commanded aileron deflection. This δ_{Ac} is then used with Eq. (30) to get the rudder deflection as well, and these two values are input into the nonlinear equations of motion Eqs. (25) in the dynamics block, which then outputs the states. These states are then fed back into each of the previous blocks, as well as used to calculate the acceleration at the pilot's seat n_{yp}, which is Eq. (26). The dynamics block also updates the position of the aircraft by solving the following Eqs. (32),

$$\begin{aligned} \dot{x}_A &= V_T \cos(\kappa_A) \\ \dot{y}_A &= V_T \sin(\kappa_A) \\ \dot{x}_B &= V_B \cos(\kappa_B) \\ \dot{y}_B &= V_B \sin(\kappa_B) \end{aligned} \tag{32}$$

Where V_B is the speed of aircraft B and κ_B is the directional axis flight path angle of craft B. The integral of these Eqs. (32) are then taken in Simulink, which yields the new inertial frame positions of each aircraft, (x_A, y_A) and (x_B, y_B). The results of this simulation are discussed in Section 4b of this thesis.

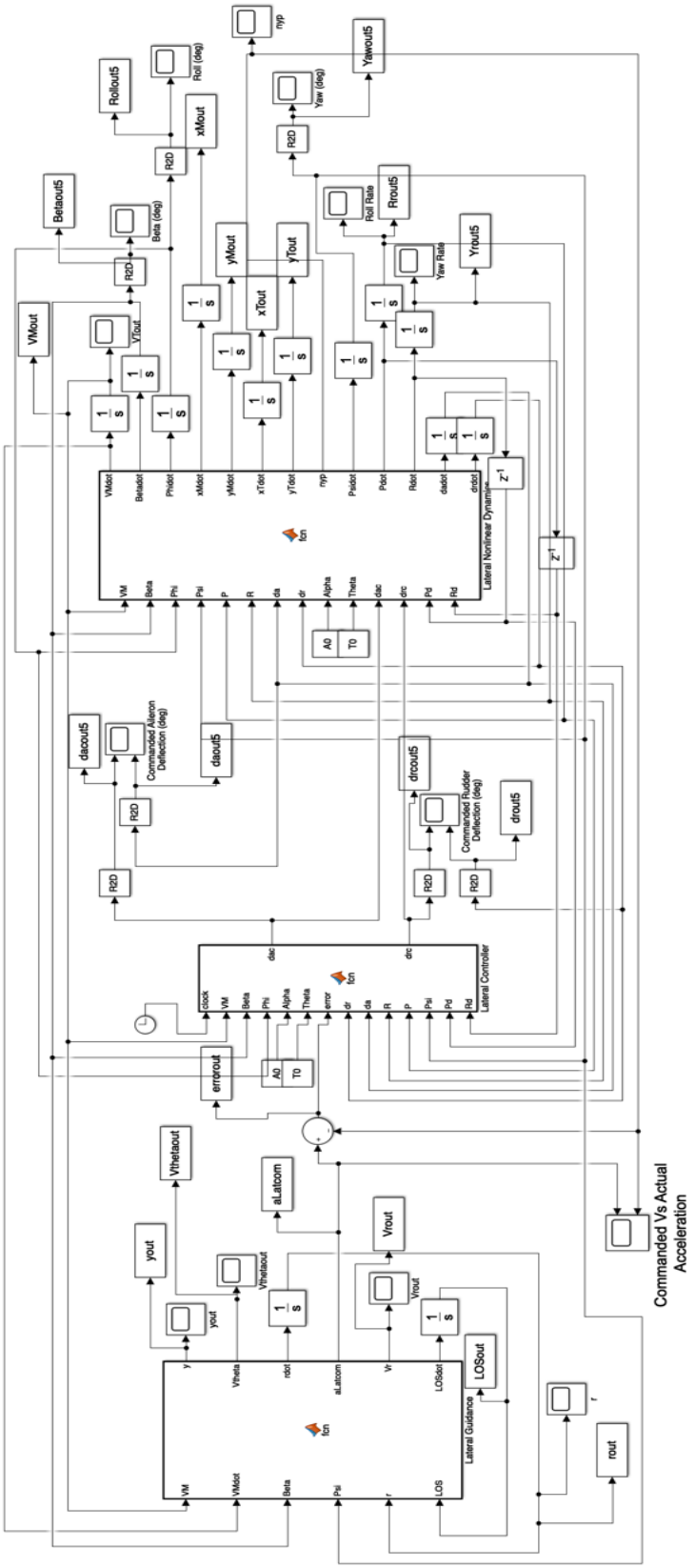

Figure 43: Lateral Simulation Simulink Model

b. SIMULATION RESULTS

Initial Collision Scenario

Fig. (44) below details the geometry and initial conditions of the scenario used in the lateral simulation. Like the longitudinal scenarios, many specifics of this scenario were chosen to ensure that the level flying craft A collided with craft B, which again occurs at the red X mark. The circles around craft A and craft B are chosen to both have a radius of $R_A = R_B = 15$ ft. All inertial frame angles here are defined with respect to North as the reference, with positive angles being left of North (i.e., angling toward West is positive). The starting point of craft A is defined as the origin, with distances more North and West being positive. The aircraft are initially a distance of r = 1500 ft. apart, with a line of sight angle θ_{LOS} = - 50 degrees. V_T and V_B are the true velocity vectors of craft A and B, respectively, and κ_A and κ_B are the angles from the chosen North reference at which the aircraft velocity vectors act, known as the directional axis flight path angles. The velocity of craft A is still the trim velocity, previously defined as 289.7165 ft/sec, while the other craft is again slower by an arbitrary factor of 1.2, resulting in a speed of 241.4304 ft/sec. For the lateral nonlinear dynamics equations, the α and θ terms (and therefore γ) used for the duration of the lateral simulations are the initial trim values introduced in the longitudinal section, which are $\alpha = \theta = -2.5751°$, and $\gamma = 0$. Since the pitch angle is constant, the pitch rate Q is equal to zero.

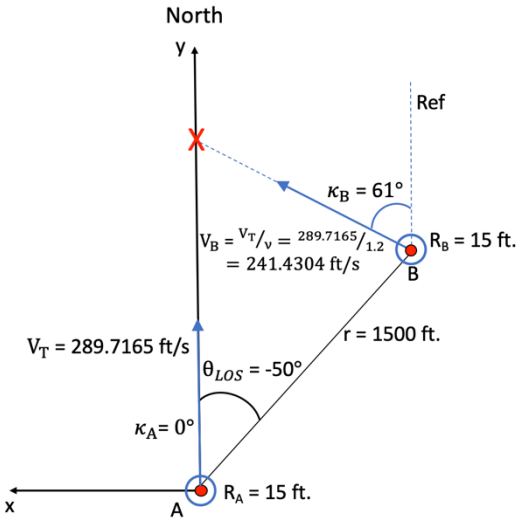

Figure 44: Initial Lateral Collision Scenario

Running the simulation with an applied acceleration of zero shows that the aircraft do indeed collide under the initial conditions, as can be seen in Fig. (45) below.

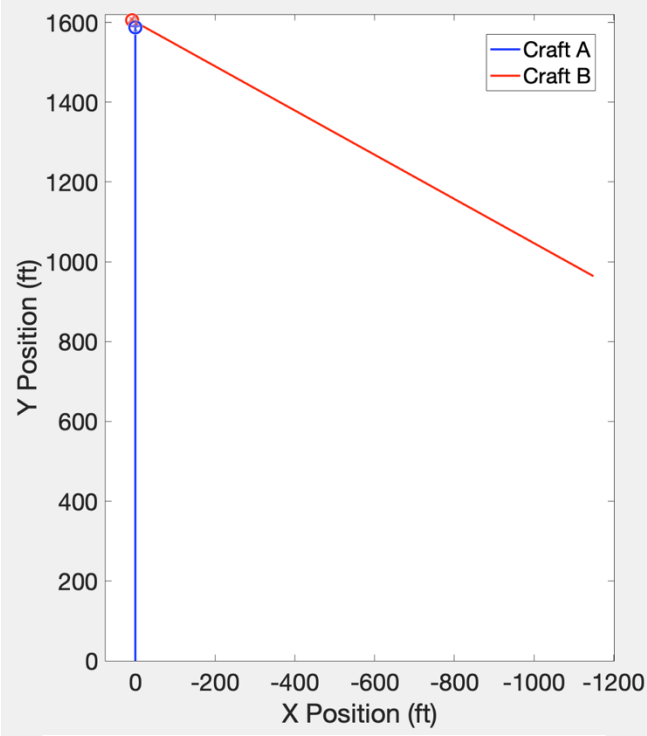

Figure 45: Initial Lateral Collision Trajectory

Craft A starts at the origin and moves North, with its position being the blue circle at the top of the blue line; while the red circle is craft B, which starts the simulation on the far right of the plot and moves up and left. It can be seen above that the two craft overlap, indicating collision, which occurs at a simulation runtime of 5.48 seconds.

In accordance with collision cone theory, the y value from Eq. (4) is found to initially be y = -538.1810 ft^2, while the initial V_r value from Eqs. (1) is initially V_r = -272.7471 ft/sec. As the value of y is less than w = 0, and since the value of V_r is negative, the criteria for being inside the collision cone are met, confirming that craft A begins and remains on a collision course with craft B.

Simulation With Applied Latax Required for Collision Avoidance

Now, the simulation is rerun with the commanded latax applied to craft A that is required for it to just barely avoid collision with craft B. The distance factor w from the guidance law Eq. (24) is set to w = 0 for a near miss, the guidance gain k_g = 0.9, and the controller gain k_c = 25. The ratio between the commanded rudder and aileron

deflection (from Eq. (30)) $k_{AR} = 0.4235$, which is approximately the 'ideal' value, and is discussed more later in this section. The simulation run time is still 5.48 seconds. The resulting trajectory in the inertial frame is seen below in Fig. (46).

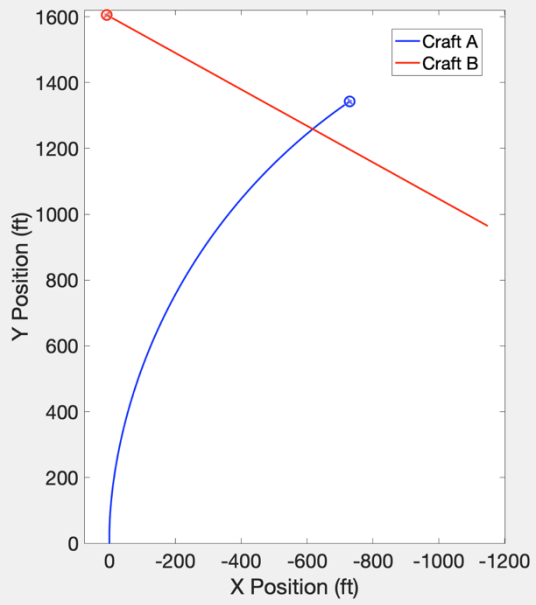

Figure 46: Lateral Simulation with Latax Applied

Obviously, Fig. (46) does not show a near miss between the two aircraft. To examine why, it can help to look to the longitudinal simulations in the previous chapter of this report. In those longitudinal simulations, an upward acceleration of 1 g is required to keep the aircraft flying level, and that constant of 1 g is what the acceleration returns to after craft A has changed flight path, as can be seen in figures (15) and (28). This 1 g of acceleration applied after the aircraft drops down serves to stop craft A from turning further down than is required, keeping it flying in a straight line from then on, as can be seen in figures (13), (14), and (27), which is how craft A barely passes by craft B. Craft A drops down to miss Craft B, then an upward acceleration of 1 g is applied to stop the craft from turning further down than necessary.

However, this operational logic does not directly translate to the lateral frame. In order to keep craft A flying in a straight line in the lateral frame, an acceleration of 0 g's is required. Any positive acceleration applied to craft A will turn it to the left, and a negative acceleration will turn it to the right. Like the longitudinal simulations, this baseline acceleration of 0 g's is what the acceleration value returns to after the required turn is made, as can be seen below in Fig. (47).

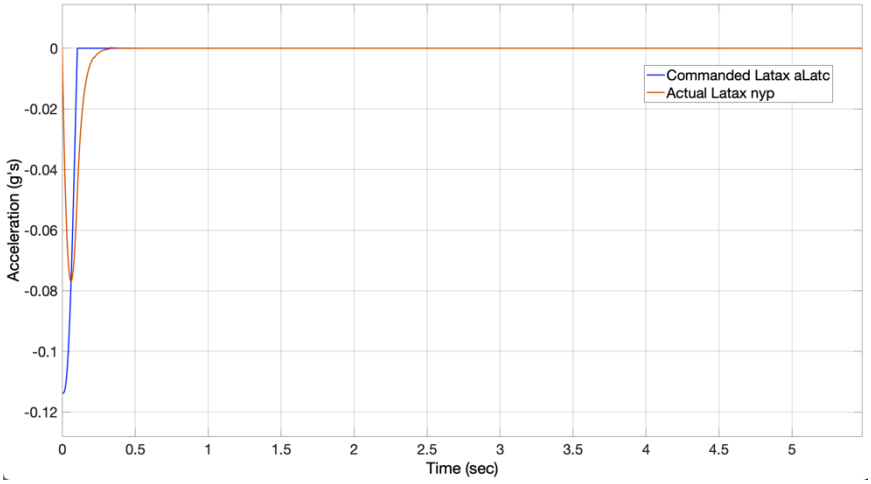

Figure 47: Lateral Simulation Commanded and Actual Accelerations

Except now, in this lateral frame, simply returning the applied acceleration to zero doesn't counteract the ongoing turn of the aircraft. Note how, in the longitudinal simulations, in figures (22) and (36), the elevator deflection first increases to drop the nose of the craft down, then returning to the 1 g acceleration causes the elevator to decrease its deflection, lifting the nose back up. As can be seen in figures (48) and (49) below, returning the acceleration to zero does nothing to return the deflection of the aileron or rudder to zero, instead leaving them constant in that deflected state.

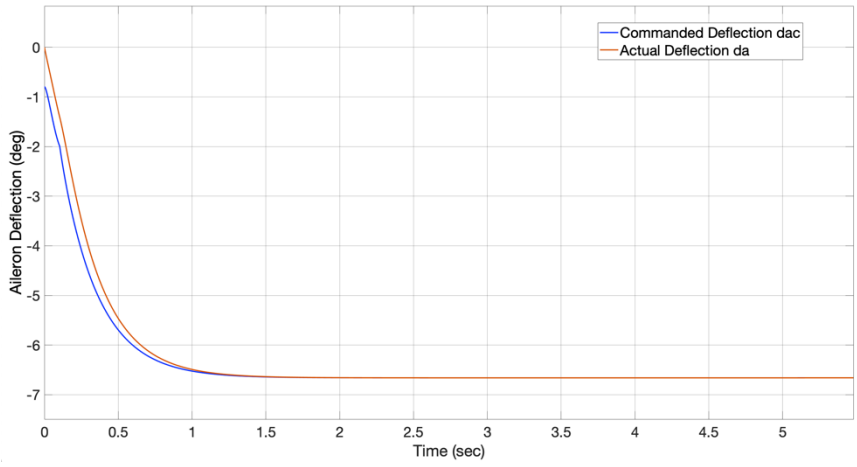

Figure 48: Lateral Simulation Aileron Deflection

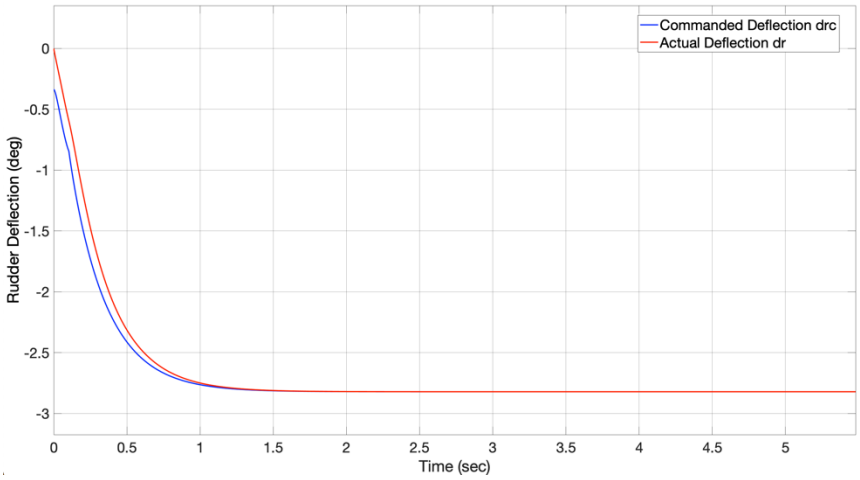

Figure 49: Lateral Simulation Rudder Deflection

In this lateral simulation, since returning to the 'baseline' acceleration of 0 g's does not apply a counteracting force to straighten up the aircraft again, the rudder and aileron stay deflected, and so of course the craft keeps turning even after there is no more acceleration being applied.

Therefore, in order to stop craft A from turning more than necessary to barely avoid craft B, a counteracting acceleration must be applied to craft A to get the aileron and rudder deflections to return to zero. While there must certainly be an elegant way to determine what magnitude of counteracting acceleration is required to return aileron and rudder deflections to zero after turning, a simpler method is to simply directly make the aileron and rudder deflection zero once craft A has turned enough to miss craft B. This simpler method is the one employed here, and is done by first looking at Fig. (47) above; once the applied acceleration n_{yp} (red line) is very close to zero after the initial decrease then increase (say, approaching within -0.005 g's of zero), then at that time the commanded aileron deflection δ_{Ac} and commanded rudder deflection δ_{Rc} in the control block are both set equal to zero. Using this method, the aileron and rudder deflections now look like figures (50) and (51) below.

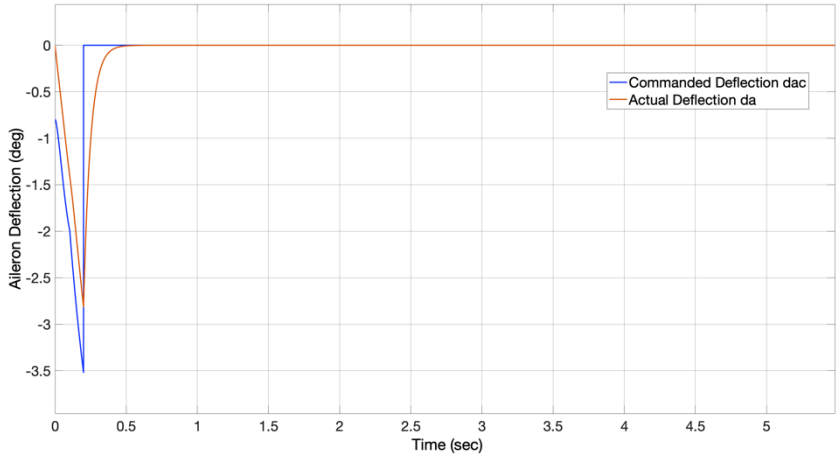

Figure 50: Modified Lateral Simulation Commanded and Actual Aileron Deflection

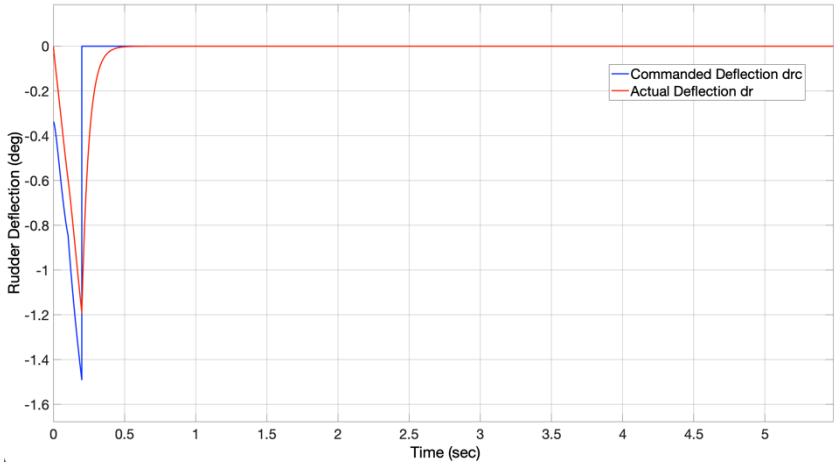

Figure 51: Modified Lateral Simulation Commanded and Actual Rudder Deflection

With the rudder and aileron being forced to zero, aircraft A still experiences a new resulting applied acceleration, as can be seen below in Fig. (52). This new acceleration is the counteracting acceleration required to stop the aircraft from turning farther than is necessary to barely miss craft B.

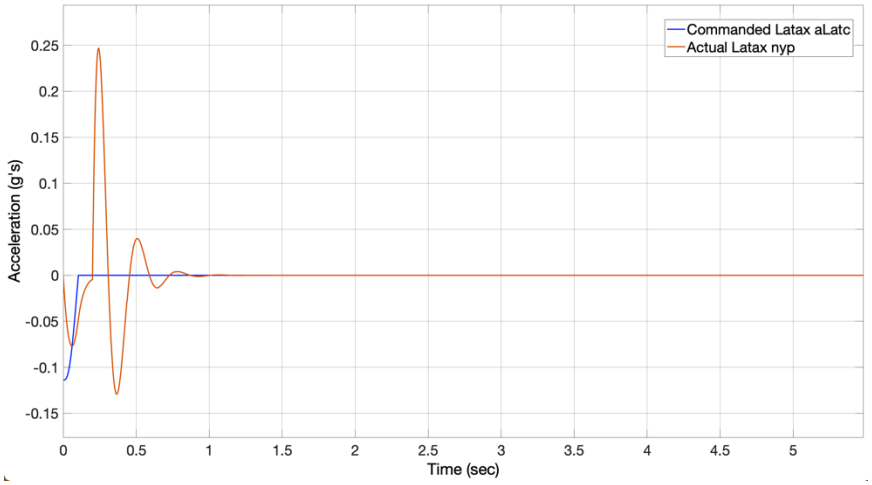

Figure 52: Modified Lateral Simulation Commanded and Actual Acceleration

Comparing Fig. (52) to Fig. (47), it can be seen that the negative commanded acceleration is still the same as before, as is the actual latax for that section. The part of the applied acceleration that suddenly becomes positive, then oscillates a bit before settling back at zero, is the counteracting acceleration that begins once the commanded aileron and rudder deflections are set to zero. With all this, the trajectory of the two aircraft in the inertial frame now looks like Fig. (53), with a closeup for convenience in Fig. (54).

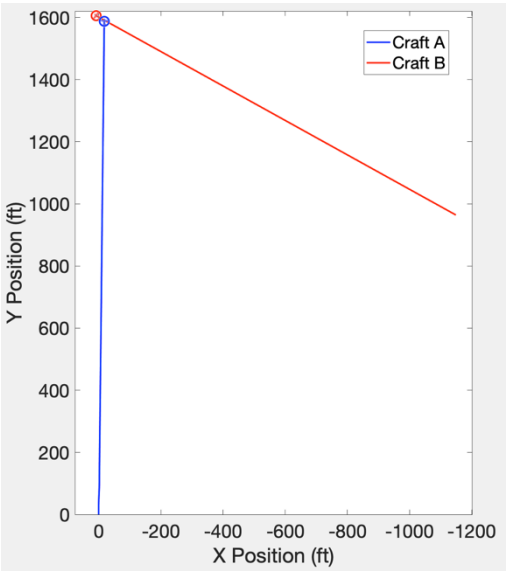

Figure 53: Modified Lateral Simulation Trajectory

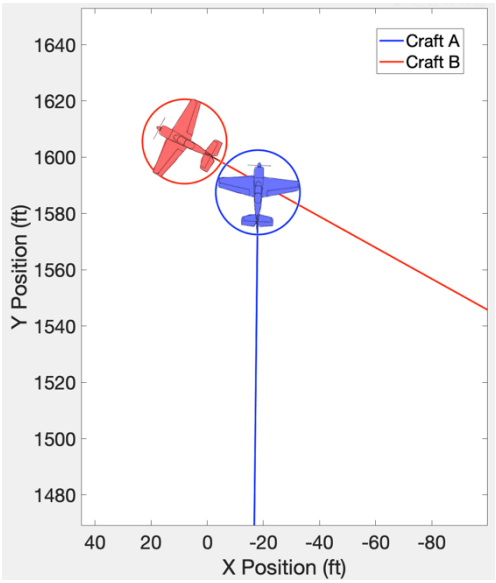

Figure 54: Modified Lateral Simulation Trajectory Closeup

The closest distance between the circle around craft A and the circle around craft B is 5.8837 inches. As Fig. (54) shows, inside of each circle is the lateral view of a Zivko aircraft from Ref. [1]. Now that the simulation seems to be

working as intended, all of the rest of the figures in this section are for this modified lateral simulation, with the next being the plot of the distance between the two craft is below in Fig. (55), with another closeup provided in Fig. (56) for convenience.

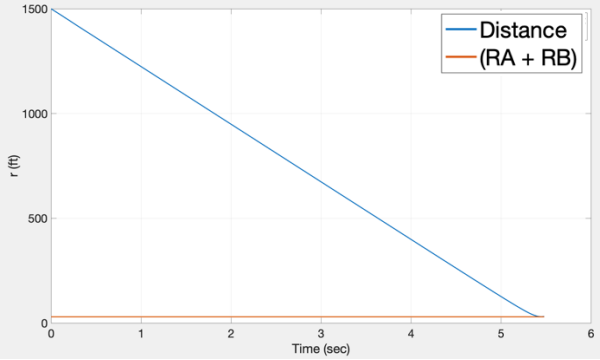

Figure 55: Distance Between Aircraft, Lateral Simulation

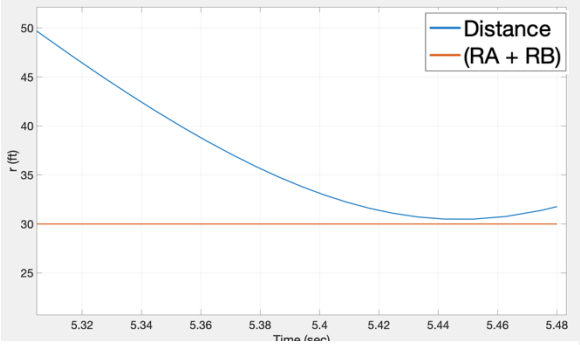

Figure 56: Distance Between Aircraft Closeup

Clearly, the distance between the two craft does not cross the orange line representing the combined sizes of the aircraft, again confirming that the two craft do not collide. The next plot in Fig. (57) shows the flight path angle of craft A.

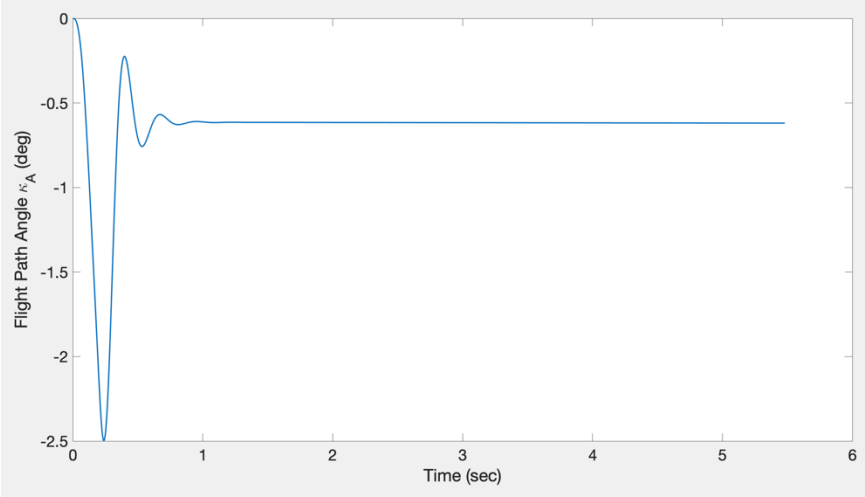

Figure 57: Lateral Simulation Flight Path Angle of Aircraft A

Reflecting the oscillation seen in the counteracting acceleration from Fig. (52), the flight path angle for craft A also oscillates a bit before settling to a constant value. This oscillation of flight path angle is relatively small enough and for a short enough duration to not easily be noticed in the trajectory of craft A in Fig. (53).

The y value from the guidance block Eq. (4) is shown in Fig. (58), again following the convention of setting the larger positive values of y equal to w.

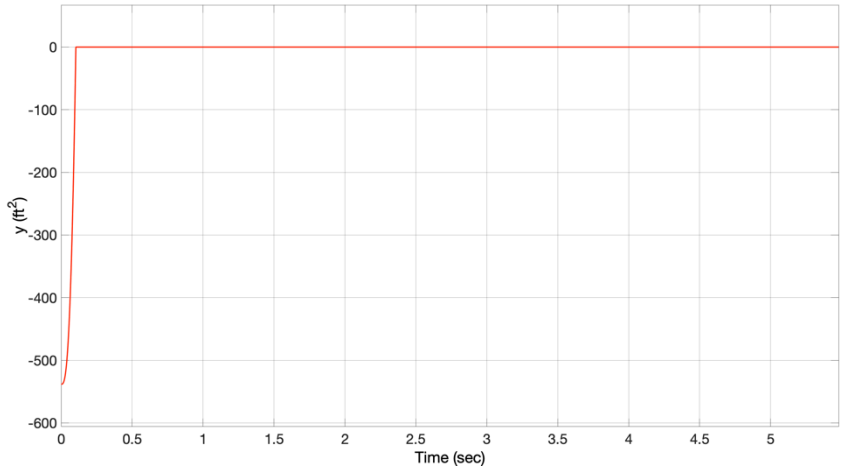

Figure 58: Guidance Block y Value

The plots of V_θ and V_r separately plotted each with respect to time are Fig. (59) and Fig. (60), respectively.

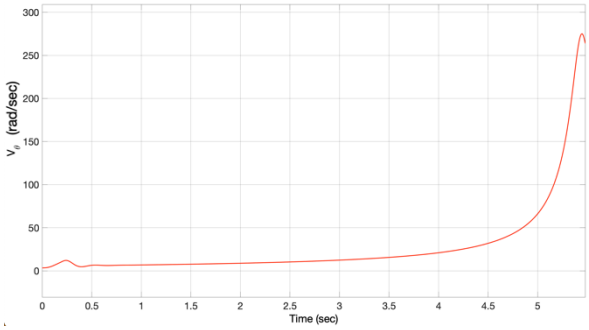

Figure 59: Plot of V_θ for Lateral Simulation

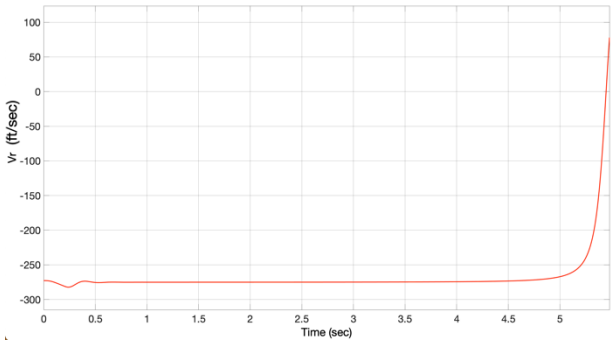

Figure 60: Plot of V_r for Lateral Simulation

Clearly, there is a slight fluctuation present on the left side in both of the above plots, which is likely due to the imposed modifications on this lateral simulation as compared to the longitudinal simulations; however, the two plots above still follow the general trends overall as the previously seen simulations. The y value from Fig. (58) again becomes larger than w faster than V_r becomes larger than 0, again showing that it is the driving factor when determining when to apply the latax to craft A.

Finally, the plots of the state variables versus simulation time in seconds are seen in the following figures, starting with the velocity of craft A in Fig. (61).

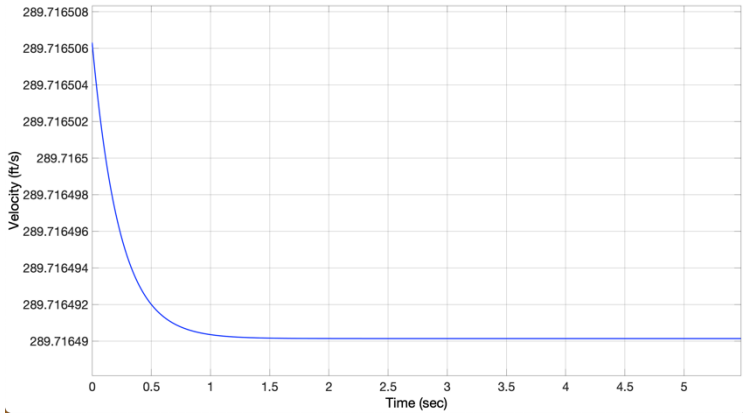

Figure 61: Lateral Simulation Aircraft A Velocity

As aircraft A does not change elevation, and does not make any large turns, it only loses a very slight amount of speed, at less than 0.0001 ft/s.

The sideslip angle β is shown in Fig. (62) below.

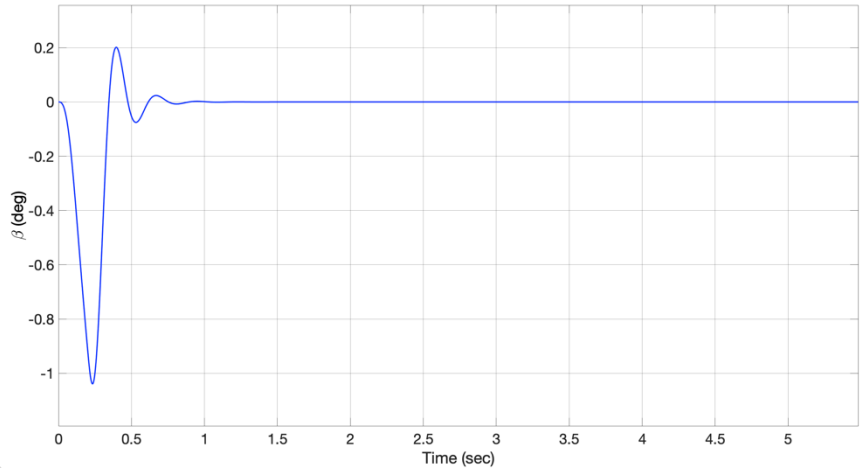

Figure 62: Sideslip Angle of Craft A

Once again showing the oscillations near the beginning of the simulation, the sideslip angle soon settles to a steady value of zero degrees. Actually, this slight oscillation behavior seems to be present in each of the state variable plots for the rest of this simulation, all with the same cause. The next state variable to be examined is the roll angle ϕ, in Fig. (63).

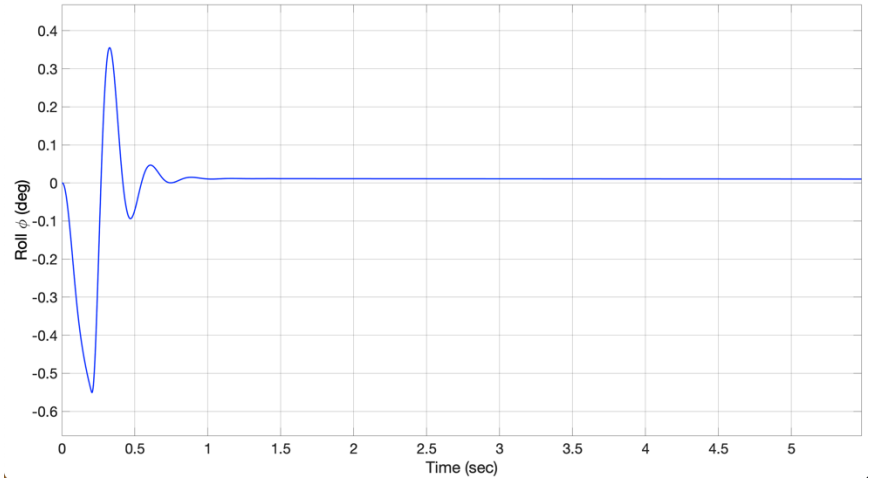

Figure 63: Roll Angle of Craft A

Fig. (64) is the yaw angle ψ for aircraft A.

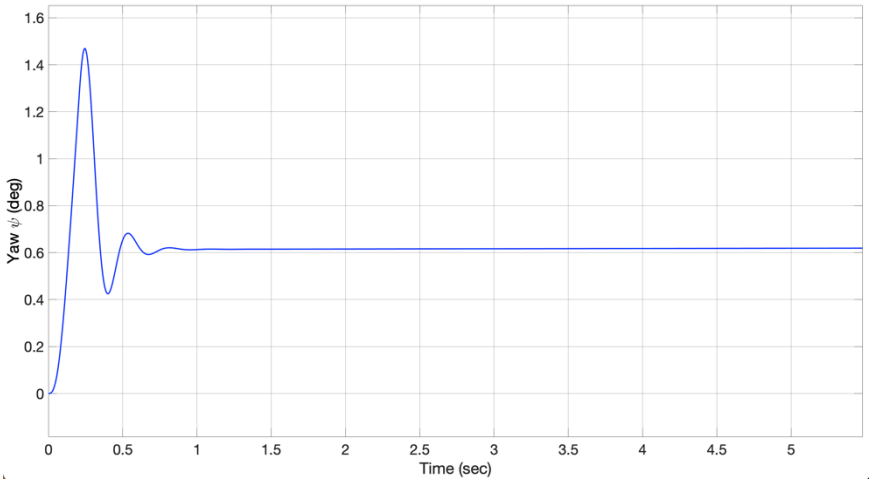

Figure 64: Aircraft A Yaw Angle

The final two plots to be seen for the lateral simulation are the roll rate P and yaw rate R, which are Fig. (65) and Fig. (66), respectively.

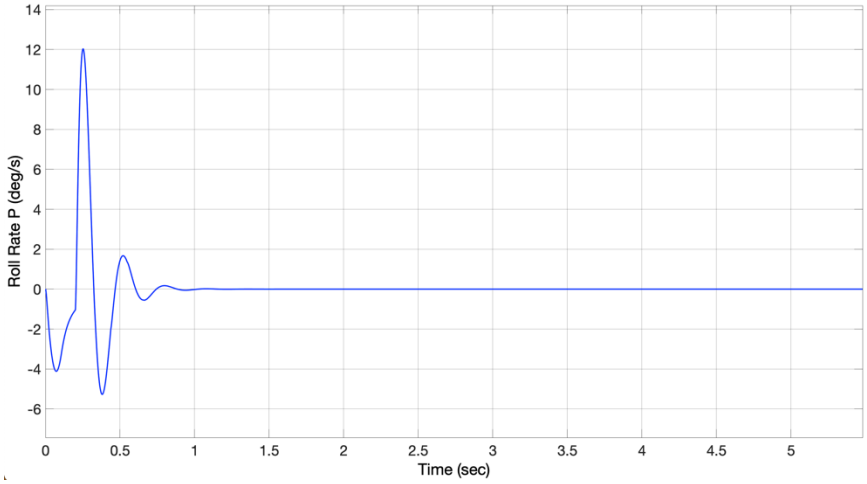

Figure 65: Aircraft A Roll Rate

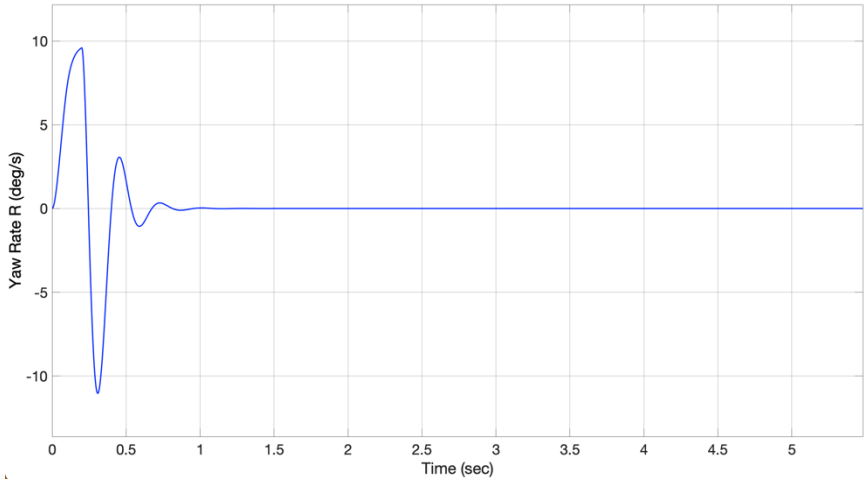

Figure 66: Aircraft A Yaw Rate

Determination of Ideal k_{AR}

As mentioned earlier, the k_{AR} value used for the above lateral simulations is $k_{AR} = 0.4235$. This value was found via trial and error, by running the simulation multiple times using various k_{AR} values, and comparing the results. The value of k_{AR} determines the ratio between the commanded aileron deflection δ_{Ac} and commanded rudder deflection δ_{Rc}, and since the actual deflections are designed to track the commanded deflections, k_{AR} also determines the ratio between the actual aileron deflection δ_A and the actual rudder deflection δ_R. From Eqs. (25), it can be seen that δ_A and δ_R mostly influence the roll rate and yaw rate values, and therefore also the roll and yaw values. And so, roll

angle plots were generated when the simulation was run with 5 different k_{AR} values, and that plot is seen below in Fig. (67).

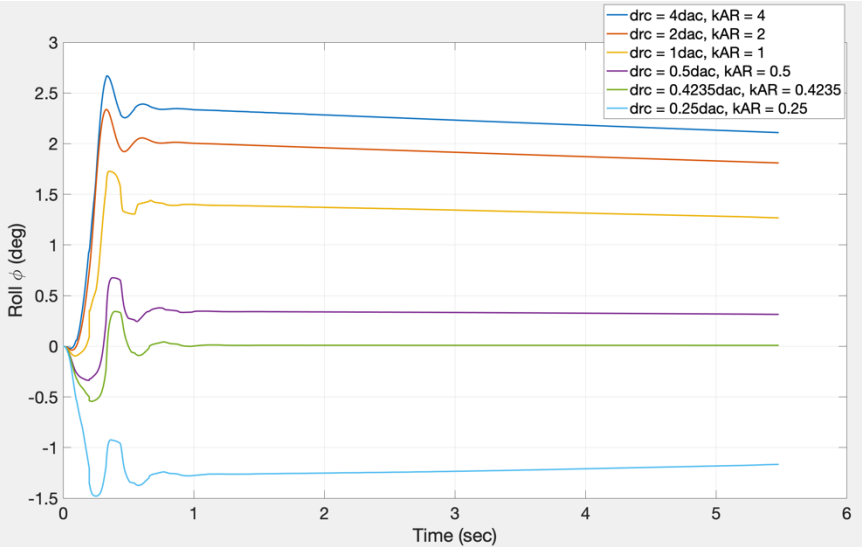

Figure 67: Roll Plots for Different k_{AR} Values

Originally, the above plot was generated only with the k_{AR} values of 4, 2, 1, 0.5, and 0.25. The general pattern was noticed where some k_{AR} value between 0.5 and 0.25 would cause the steady-state portion of the roll plot to settle on a roll angle of zero, which was determined to be the ideal end-state roll angle. After some guessing and checking, a k_{AR} value of 0.4235 was settled on as being very near the ideal value, and is seen in Fig. (67) as the green line, which is the same as the roll angle plot in Fig. (63). It is possible that there is a more analytical method besides trial and error to determine a more exact ideal value of k_{AR}, but that is outside the purview of this thesis, and the above method was deemed to be sufficient.

5. CONCLUSIONS and FUTURE WORK

Conclusions

In this work, collision between two fixed-wing aircraft was examined along the longitudinal axis and the lateral-directional axis separately. Guidance laws governing the commanded acceleration were derived using dynamic inversion, and were designed to be applied to steer aircraft A's trajectory out of the collision cone with aircraft B. Control laws governing the commanded elevator and aileron/rudder deflections were also derived via dynamic inversion, and both the guidance and control laws were integrated in MATLAB/Simulink programs, along with the aircraft nonlinear dynamics and actuator dynamics. Results for various simulation conditions were presented and discussed.

Future Work

While a Simulink framework including guidance, control, and nonlinear dynamics is a relatively complete aircraft system, it is clear that a more holistic aircraft system is possible. Future work could append these models with noisy sensors, actuators with nonlinearities, and observers, to better model a realistic aircraft. Aircraft engines and propellers can also be factored in, as can environmental conditions like wind gusts and turbulence, etc. Other collision scenarios can also be explored, as can examples with a maneuvering aircraft B. There can also be more than just two aircraft on collision courses. Additionally, 3-D scenarios can also be examined.

REFERENCES

[1] "3-View Drawing of a Edge 540t." *RCU Forums*, 13 Mar. 2005, https://www.rcuniverse.com/forum/giant-scale-aircraft-3d-aerobatic-110/2754267-3-view-drawing-edge-540t.html.

[2] "Airborne Collision Avoidance System (ACAS)." *SKYbrary Aviation Safety*, 6 Apr. 2022, https://skybrary.aero/articles/airborne-collision-avoidance-system-acas.

[3] Carbone, C., Ciniglio, U., Corraro, F., & Luongo, S. (2006). A novel 3-D geometric algorithm for aircraft autonomous collision avoidance. In *Proc. of the 45th IEEE conference on decision and control* (pp. 1580–1585). San Diego, CA, USA, December 2006.

[4] Chakravarthy, A., & Ghose, D. (1998). Obstacle avoidance in a dynamic environment: a collision cone approach. *IEEE Transactions on Systems, Man and Cybernetics. Part A. Systems and Humans*, *28*(5), 562–574.

[5] Childs, Jeremy. "NTSB Report Details Fatal Plane Crash near Camarillo Airport." *Ventura County Star*, Ventura County Star, 7 July 2022, https://www.vcstar.com/story/news/local/communities/camarillo/2022/07/07/ntsb-report-details-fatal-plane-crash-near-camarillo-airport/7830493001/.

[6] Day, Richard E. "Coupling Dynamics in Aircraft: A Historical Perspective." *NASA Special Publication*, vol. 532, 1997.

[7] Duncan, Ariann M. "Stability and Control Analysis of the Zivko Edge 540T." *Georgia Institute of Technology*.

[8] Ferrara, A., & Paderno, J. (2006). Application of switching control for automatic pre-crash collision avoidance in cars. *Nonlinear Dynamics*, *46*, 307–321.

[9] Ferrara, A., & Vecchio, C. (2007). Collision avoidance strategies and coordinated control of passenger vehicles. *Nonlinear Dynamics*, *49*, 475–492.

[10] Ferrara, A., & Vecchio, C. (2009). Second order sliding mode control of vehicles with distributed collision avoidance capabilities. *Mechatronics*, *19*, 471–477.

[11] Gross, J., Rajvanshi, R., & Subbarao, K. (2004). Aircraft collision detection and resolution using mixed geometric and collision cone approaches. In *Proc. of AIAA guidance, navigation and control conference*, Rhode Island, Providence, USA. Paper No. AIAA 2004-4879.

[12] Leone, Dario. "USMC KC-130J Pilot Who Landed His Plane Safely in a Farmer's Field in California after a Mid-Air Collision with an F-35B Received the Distinguished Flying Cross." *The Aviation Geek Club*, 30 May 2022, https://theaviationgeekclub.com/usmc-kc-130j-pilot-who-landed-his-plane-safely-in-a-farmers-field-in-california-after-a-mid-air-collision-with-an-f-35b-received-the-distinguished-flying-cross/.

[13] Lin, C. (1991). *Modern navigation, guidance and control processing*. Englewood Cliffs: Prentice Hall.

[14] Patel, Vimal, and Johnny Diaz. "Four Killed in Midair Plane Collision in Nevada." *The New York Times*, The New York Times, 18 July 2022, https://www.nytimes.com/2022/07/18/us/las-vegas-plane-crash.html.

[15] Roskam, J. (1979). *Airplane Flight Dynamics and Automatic Flight Controls*. Roskam Aviation and Engineering Corporation.

[16] "Separation Standards." *SKYbrary Aviation Safety*, 16 May 2021, https://skybrary.aero/articles/separation-standards.

[17] Stevens, B. L., Lewis, F. L., & Johnson, E. N. (2015). *Aircraft Control and Simulation: Dynamics, Controls Design, and Autonomous Systems*. John Wiley & Sons.

[18] Stoliker, P. C., et al. “Linearized Poststall Aerodynamic and Control Law Models of the X-31A Aircraft and Comparison With Flight Data.” *NASA/TM-97-206318*, Dec. 1997.

[19] “Terrain Avoidance and Warning System (TAWS).” *SKYbrary Aviation Safety*, 29 May 2022, https://skybrary.aero/articles/terrain-avoidance-and-warning-system-taws.

[20] Watanabe, Y., Calise, A. J., Johnson, E. N., & Evers, J. H. (2006a). Minimum-effort guidance for vision-based collision avoidance. In *Proc. of AIAA atmospheric flight mechanics conference*, Keystone, Colorado, USA, August 2006. Paper No. AIAA 2006- 6641.

[21] Watanabe, Y., Johnson, E. N., & Calise, A. J. (2006b). Vision-based guidance design from sensor trajectory optimization. In *Proc. of AIAA atmospheric flight mechanics conference*, Keystone, Colorado, USA. August 2006. Paper No. AIAA 2006-6607.

[22] Watanabe, Y., Calise, A. J., & Johnson, E. N. (2007a). Vision-based obstacle avoidance for UAVs. In *Proc. of AIAA guidance, navigation and control*, Hilton Head, South Carolina, USA, August 2007. AIAA 2007-6829.

[23] Watanabe, Y., Johnson, E. N., & Calise, A. J. (2007). Stochastically optimized monocular vision-based guidance design. In *Proc. of AIAA atmospheric flight mechanics conference,* Hilton Head, South Carolina, USA, August 2007. Paper No. AIAA 2007-6865.

[24] Weingarten, Norman C. “An In-Flight Investigation of a Twin Fuselage Configuration in Approach and Landing.” *NASA Contractor Report 172366*, Aug. 1984.

[25] “Zivko Aeronautics Inc. Edge Aircraft.” *Zivko Aeronautics Inc*, © 1987 - 2022 - Zivko Aeronautics Inc. - Website Designed by Needle & Hay Creative, http://www.zivko.com/edge/.