

3-2017

Collision Avoidance and Navigation of UAS Using Vision-Based Proportional Navigation

Matthew J. Clark

Follow this and additional works at: <https://commons.erau.edu/edt>



Part of the [Aerospace Engineering Commons](#)

Scholarly Commons Citation

Clark, Matthew J., "Collision Avoidance and Navigation of UAS Using Vision-Based Proportional Navigation" (2017). *Dissertations and Theses*. 323.
<https://commons.erau.edu/edt/323>

This Thesis - Open Access is brought to you for free and open access by Scholarly Commons. It has been accepted for inclusion in Dissertations and Theses by an authorized administrator of Scholarly Commons. For more information, please contact commons@erau.edu.

COLLISION AVOIDANCE AND NAVIGATION OF UAS USING VISION-BASED
PROPORTIONAL NAVIGATION

A Thesis
Submitted to the Faculty
of
Embry-Riddle Aeronautical University
by
Matthew J. Clark

In Partial Fulfillment of the
Requirements for the Degree
of
Master of Science in Aerospace Engineering

March 2017
Embry-Riddle Aeronautical University
Daytona Beach, Florida

COLLISION AVOIDANCE AND NAVIGATION OF UAS USING VISION-BASED
PROPORTIONAL NAVIGATION

by

Matthew J. Clark

A Thesis prepared under the direction of the candidate's committee chairman, Dr. Richard Prazenica, Department of Aerospace Engineering, and has been approved by the members of the thesis committee. It was submitted to the School of Graduate Studies and Research and was accepted in partial fulfillment of the requirements for the degree of Master of Science in Aerospace Engineering.


THESIS COMMITTEE



Chairman, Dr. Richard Prazenica



Member, Dr. Hever Moncayo



Member, Dr. Troy Henderson



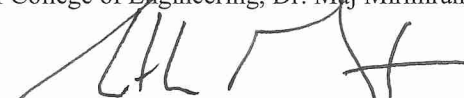
Graduate Program Coordinator, Dr. Magdy Attia

3.30.2017
Date



Dean of College of Engineering, Dr. Maj Mirmirani

3/30/2017
Date



Vice Chancellor, Academic Support, Dr. Christopher Grant

3/31/2017
Date

ACKNOWLEDGMENTS

This work could not have been completed without the inspiration and guidance from Dr. Richard Prazenica. I am grateful for all his efforts in directing me through any obstacles encountered while allowing me to follow my own path (pun intended). I would also like to thank Dr. Hever Moncayo and Dr. Troy Henderson for their support as my advisory committee.

I would like to acknowledge Dr. Heidi Steinhauer and the entire Engineering Fundamentals Department. It has been an absolute pleasure to work as a Graduate Teaching Assistant alongside a group I consider family.

I would not have been able to complete this work without the emotional support of my family and friends. They have continued to push me through one of the most difficult and rewarding times of my life, and their love and laughs have ensured my stability (yes, another pun) through all of it.

Table Of Contents

TABLE OF CONTENTS.....	iv
LIST OF TABLES	v
LIST OF FIGURES	vi
ABSTRACT.....	ix
1. Introduction	1
1.1. Overview Of UAV Sense And Avoid.....	2
1.2. Vision-Based Proportional Navigation	14
2. UAV and Camera Modeling.....	18
2.1. Medium-Scale, Propeller Uav Model	19
2.2. Small-Scale, Rc Propeller Uav Model	19
2.3. Camera Model.....	20
2.4. Simulation Environment	22
3. Pro-Nav Intercept Law	25
4. Pro-Nav Avoidance Law – Thresholding Approach	30
5. Pro-Nav Avoidance Laws – Most Imminent Threat	37
5.1. Dynamic Avoidance.....	37
5.2. Static Avoidance	48
6. Pro-Nav Avoidance Laws – Objective Weighting Function Approach	51
6.1. Dynamic Avoidance.....	53
6.2. Static Avoidance	61
7. Pro-Nav Avoidance Law – Objective Function Using $\Delta\psi$ Differencing.....	68
8. Virtual Reality Simulation.....	74
8.1. MetaVR Virtual Reality Scene Generator	74
8.2. Feature-Point Tracking	76
8.3. Single Obstacle Avoidance	78
8.4. Urban Canyon Avoidance.....	80
8.5. Urban Navigation And Avoidance.....	84
9. Conclusion.....	87

LIST OF TABLES

Table 1: Medium Scale UAV Geometric And Flight Parameters	19
Table 2: Small-Scale UAV Geometric And Flight Parameters	20
Table 3: UAV Single Avoidance, Effect Of Position, V=176 ft/s.....	35
Table 4: Single Avoidance Effect Of Velocity, V=88, 176, 264 ft/s.....	35
Table 5: Single Avoidance No Intercept, V=88, 176, 264 ft/s	36
Table 6: UAV Individual Avoidance Data - Obstacle 1 And 2 North Intercept.....	39
Table 7: UAV Multi-Avoidance Data - Obstacle 1 And 2 North Intercept.....	40
Table 8: UAV Individual Avoidance Data - Obstacle 2 Offset Intercept.....	41
Table 9: UAV Multi-Avoidance Data - Obstacle 2 Offset Intercept	42
Table 10: UAV Individual Avoidance Data - Obstacle 1 Offset Intercept.....	43
Table 11: UAV Multi-Avoidance Data - Obstacle 2 Offset Intercept	45
Table 12: UAV Multi-Avoidance Data - Obstacle 1 And 2 North Intercept.....	46
Table 13: UAV Multi-Avoidance Data - Obstacle 1 And 2 North Intercept - Close Paths	48
Table 14: Corresponding Miss Distance Between UAV And Obstacles For χ Based Cost Functions.....	55
Table 15: Corresponding Miss Distance Between UAV And Obstacles For $\dot{\psi}$ Based Cost Functions.....	57
Table 16: Corresponding Miss Distance Between UAV And Obstacles For $\Delta\psi$ Based Cost Functions	60

LIST OF FIGURES

Figure 1: Diagram Of Typical UAS (Angelov, 2012)	1
Figure 2: MQ-1 Predator, A Well-Known, SAA Capable UAV (Angelov, 2012).....	2
Figure 3: Early RC Aircraft, Germany 1936 (Mueller, 2009)	3
Figure 4: Ryan Model 147 (Mrdrone.Net, 2017).....	4
Figure 5: NASA ERAST's ER-2 In Flight (Losey, 2001)	5
Figure 6: Example Detection And Response Algorithm of a SAA UAV (Yu & Zhang, 2015)	6
Figure 7: Traffic And Resolution Advisory Zones For TCAS (Eurocontrol, N.D.).....	7
Figure 8: Examples Of Current SAA Technologies and The Common Range Limitations (Yu & Zhang, 2015).....	9
Figure 9: Example Point Cloud Scene Collected Via Lidar Measurements (Ryabinin, 2017)	10
Figure 10: Relative Zones Of Detection For Optical And PANCAS Sensing Technologies (SARA Inc., 2012)	11
Figure 11: Aerial, Infrared Image Of Wildfires, Taken By RQ-4 Global Hawk (US Navy, 2008)	12
Figure 12: Stereovision Parallax, Demonstrated By The Overlapping FOV Of Camera A And B (Lau, 2012)	13
Figure 13: Diagram Of Bank-To-Turn Control Method For UAV Simulations	18
Figure 14: Visual Representation Of Image And Focal Plane Related By Intrinsic Matrix (Openmvg, 2013)	22
Figure 15: Diagram Of Simulink Block Configuration For Point Mass Simulations	23
Figure 16: Aerodynamic Component Buildup For Six Dof Model	24
Figure 17: Diagram Of Simulink Block Configuration For Virtual Reality Simulations.	24
Figure 18: LOS Angles With Respect To The Image Plane.	25
Figure 19: Example 1 - UAV Intercept Path (Units in feet).	27
Figure 20: Example 2 - UAV Intercept Path (Units in feet).	28
Figure 21: Example 3 - UAV Intercept Path (Units in feet).	29
Figure 22: UAV Single Avoidance Path – Effect On Distance – Position = (1,1) Mile ..	32
Figure 23: UAV Single Avoidance Path – Effect On Distance – Position = (2,2) Mile ..	33

Figure 24: UAV Single Avoidance Path – Effect On Distance – Position = (0.5,0.5) Mile	34
Figure 25: UAV Individual Avoidance Paths (Ft) - Obstacle 1 And 2 North Intercept - $\dot{\chi} = .01deg/s$	38
Figure 26: UAV Multi-Avoidance Paths (Ft) - Obstacle 1 And 2 North Intercept	39
Figure 27: UAV Individual Avoidance Paths (Ft) - Obstacle 2 Offset Intercept - $\dot{\chi} =$ $.01deg/s$	40
Figure 28: UAV Multi-Avoidance Paths (Ft) - Obstacle 2 Offset Intercept.....	42
Figure 29: UAV Individual-Avoidance Paths (Ft) - Obstacle 1 Offset Intercept.....	43
Figure 30: UAV Multi-Avoidance Paths (Ft) - Obstacle 1 Offset Intercept.....	44
Figure 31: UAV Multi-Avoidance Paths (Ft) - Obstacle 1 And 2 North Intercept	46
Figure 32: UAV Multi-Avoidance Paths (Ft) - Obstacle 1 And 2 North Intercept - Close Paths.....	47
Figure 33: Most Imminent Threat Avoidance - Static Point Mass Wall	49
Figure 34: UAV Avoidance Path Comparison Using $\dot{\chi}$ Cost Function Forms - Dynamic	54
Figure 35: Comparison Of Actual UAV Heading For Corresponding $\dot{\chi}$ Cost Function Avoidance Paths – Dynamic Case	56
Figure 36: UAV Avoidance Path Comparison Using $\dot{\psi}$ Cost Function Forms – Dynamic	57
Figure 37: Comparison Of UAV Actual Heading For Corresponding $\dot{\psi}$ Cost Function Avoidance Paths – Dynamic Case	58
Figure 38: UAV Avoidance Path Comparison Using $\Delta\psi$ Cost Function Forms – Dynamic Case	59
Figure 39: Comparison Of UAV Heading For Corresponding $\Delta\psi$ Cost Function Avoidance Paths – Dynamic Case	60
Figure 40: UAV Avoidance Path Comparison Using $\dot{\chi}$ Cost Function Forms – Static Case.....	61
Figure 41: Comparison Of UAV Actual Heading For Corresponding $\dot{\chi}$ Cost Function Avoidance Paths – Static Case.....	62

Figure 42: UAV Avoidance Path Comparison Using ψ Cost Function Forms – Static Case	63
Figure 43: Comparison Of UAV Actual Heading For Corresponding ψ Cost Function Avoidance Paths – Static Case	64
Figure 44: UAV Avoidance Path Comparison Using $\Delta\psi$ Cost Function Forms – Static Case	65
Figure 45: Comparison Of UAV Actual Heading For Corresponding $\Delta\psi$ Cost Function Avoidance Paths – Static Case	66
Figure 46: UAV Avoidance Path For Static Corner Avoidance	68
Figure 47: UAV Heading History For Static Corner Avoidance	69
Figure 48: UAV Avoidance Path Of Dynamic And Close Static Obstacles	70
Figure 49: UAV Actual Heading History For Avoidance Of Dynamic And Close Static Obstacles	71
Figure 50: UAV Avoidance Path Of Dynamic And Far Static Obstacles	72
Figure 51: UAV Heading History For Avoidance Of Dynamic And Far Static Obstacles	73
Figure 52: Process Overview For Pro-Nav Guidance Virtual Reality Integration	75
Figure 53: Example Corner Detection Of Building Features In MetaVR	77
Figure 54: Camera View From UAV Of Single Building Metavr Simulation	78
Figure 55: Feature Point Overlay Of UAV Camera View For Single Building Avoidance	79
Figure 56: UAV Avoidance Path For Single Building Metavr Simulation	79
Figure 57: UAV Heading History For Single Building Metavr Simulation	80
Figure 58: Camera View From UAV Of Urban Canyon Metavr Simulation	81
Figure 59: Feature Point Overlay Of UAV Camera View Within The Urban Canyon	82
Figure 60: UAV Avoidance Path For Urban Canyon Metavr Simulation	83
Figure 61: UAV Heading History For Urban Canyon Metavr Simulation	84
Figure 62: Complex Virtual Urban Environment	84
Figure 63: UAV Path For Complex Urban Virtual Reality Scenario	85
Figure 64: UAV Complex Urban Virtual Scenario - Pre-Collision	86
Figure 65: Complex Urban Virtual Scenario – Post-Collision	86

ABSTRACT

Clark, Matthew MSAE, Embry-Riddle Aeronautical University, March 2017. Collision Avoidance and Navigation of UAS Using Vision-Based Proportional Navigation

Electro-optical devices have received considerable interest due to their light weight, low cost, and low algorithm requirements with respect to computational power. In this thesis, vision-based guidance laws are developed to provide sense and avoid capabilities for unmanned aerial vehicles (UAVs) operating in complex environments with multiple static and dynamic collision threats. These collision avoidance guidance laws are based on the principle of proportional navigation (Pro-Nav), which states that a UAV is on a collision course with another vehicle or object if the line-of-sight (LOS) angles to the object remain constant. The guidance laws are designed for use with monocular electro-optical devices, which provide information on the LOS angles to potential collision threats, but not the range. The development of these guidance laws propagates from an investigation into numerous methods of Pro-Nav based guidance, including the use of LOS rate thresholding, avoidance of the most imminent threat detected, and objective-based cost optimization. The collision avoidance guidance laws were applied to nonlinear, six degree-of-freedom UAV models in various simulation environments including a varying number of static and dynamic obstacles. A final form of the avoidance law, determined from these simulation studies, was applied to a small-scale UAV model flying through a virtual urban environment, which utilizes camera-in-the-loop simulation techniques.

The final results of these studies showed that the most effective approach was to implement a cost function-based avoidance law that includes a term based on the Pro-Nav intercept heading for a desired waypoint and avoidance terms for all obstacles in view that pose a collision threat. Obstacle avoidance headings in the cost function are based on the difference in the obstacle LOS rates from the magnitude of the minimum safe LOS rate. When applied to UAV simulations in a virtual urban environment, this guidance law provided successful avoidance for the case of a single building, maintained a safe heading through an urban canyon, and determined the safest path through a complex urban layout. For the case of the complex urban layout, a single collision during flight occurred due to a lack of visual feature points to contribute to the avoidance law calculation.

1. INTRODUCTION

An “unmanned aerial vehicle” (UAV) is an aircraft without an on-board pilot to control it. While being previously synonymous with “unmanned aerial systems” (UAS), there has been clarification by the Federal Aviation Administration (FAA), European Aviation Safety Agency (EASA), and the International Civil Aviation Organization (ICAO), that they are distinctly different. A UAV is considered a device used for flight that has no pilot, including all classes of airplanes, helicopters, airships, and translational balloons. The classification of a UAS is comprised of an unmanned vehicle, as well as encompassing the ground control station, communication links, and launch retrieval systems (Angelov, 2012).

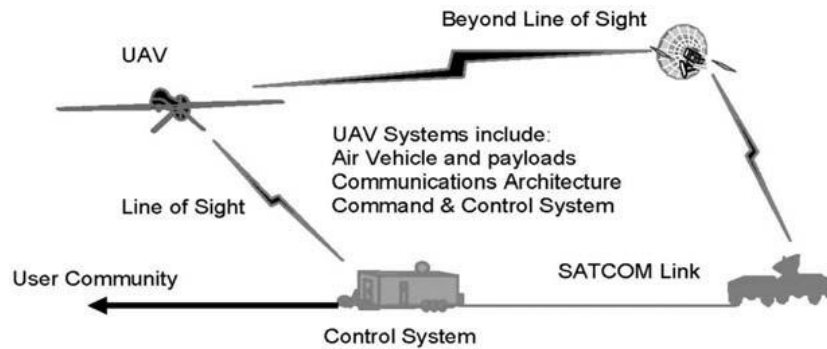


Figure 1: Diagram of typical UAS (Angelov, 2012)

This important distinction plays a critical role in the work presented in this paper. The nature of this research is to investigate a guidance and control system in relation to an unmanned aerial vehicle (UAV) alone, not an entire system; and while this scope seems limited, it is in fact a much desired and popular aspect of research in current UAV developments (Angelov, 2012). Current and prevalent developments of UAV sense and avoid (SAA) capabilities rely on technologies such as infrared sensors, lasers, acoustic emissions, and stereo cameras to map out point clouds of the environment or determine

physical distances between the UAV and objects or scenery in view.



Figure 2: MQ-1 Predator, a well-known, SAA capable UAV (Angelov, 2012)

The SAA method presented in this thesis investigates a less common method for UAV navigation and avoidance: using a single monocular camera and feature point bearings. While the immediate weight and computational benefits are known, the capabilities and limitations of a monocular vision-based controller, due to lack of range information, are investigated and presented in this thesis.

1.1. Overview of UAV Sense and Avoid

In the early 1900's, automatic stabilization, remote control, and automatic navigation were the three primary technologies said to allow for the progression of powered, manned airplanes to become unmanned (Mueller, 2009).

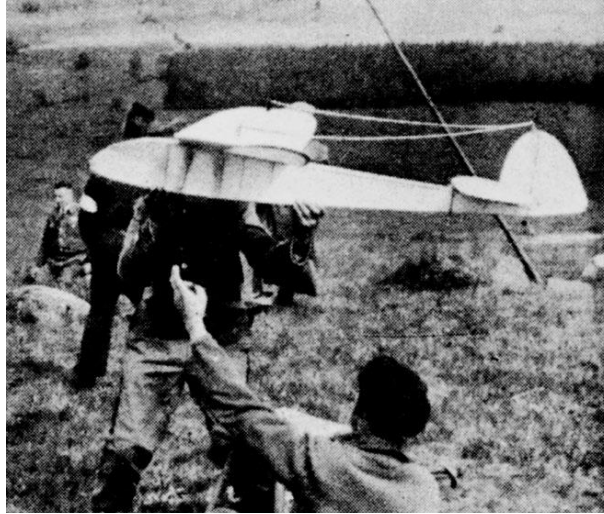


Figure 3: Early RC Aircraft, Germany 1936 (Mueller, 2009)

Prior to the emergence of the Lawrence and Sperry Aircraft company, small-scale, remote control (RC) aircraft had become well known and popular amongst a specific set of civilian interest groups, and was the closest thing to a modern definition of an unmanned aerial vehicle (Mueller, 2009). Lawrence and Sperry then provided a critical technology to full size aircraft, the gyroscope, that allowed for the possibility of attitude determination in an aircraft, in turn leading to the implementation of a stability autopilot, a mechanism or device capable of controlling an aircraft to ensure stability was maintained (Angelov, 2012). This was an opportunity that the military capitalized on, soon focusing on the development of autopilots for ordinance delivery devices and small aerial vehicles for anti-aircraft weapons training. After the Ryan Model 147 was put into service as the first reconnaissance UAV with a still camera, further developments in UAV and UAS focused primarily on airframe design and payload size to incorporate improved cameras (Angelov, 2012).



Figure 4: Ryan Model 147 (MrDrone.net, 2017)

It was not until the late 1970's, after the Vietnam war, that the production of smaller transmitters and electric sensors sparked interest by the Naval Research Laboratory (NRL) to outfit UAVs with video cameras, since advances in technology had decreased their size dramatically (Mueller, 2009). The first practical use of these newer UAV platforms became known during the Gulf War, with the AAI RQ-2 Pioneer leading reconnaissance and surveillance missions and providing a stepping-stone for the most famous UAV, the General Atomics MQ-1 Predator (Angelov, 2012).

Civil development and use of UAVs gradually and quietly developed behind the scenes during the military integration. NASA, being a forerunner in civil research and development, undertook efforts for developing high altitude and prolonged endurance UAVs in the 1980's (Angelov, 2012). When investigating the ozone depletion around Earth, NASA's go-to flight vehicles, a modified Douglas DC-8 and a Lockheed ER-2, presented a serious risk to the project when planning to test ozone over the Antarctic. If an incident occurred in which the pilots had to eject from the aircraft, survival would be difficult and a rescue would take too long. The ER-2 also had a flight ceiling of 20 km, whereas ozone measurements were to be taken at 30 km, where depletion take place.



Figure 5: NASA ERAST's ER-2 in flight (Losey, 2001)

In 1994, the Environmental Research Aircraft and Sensor Technology (ERAST) program became the research project that pushed to solve this problem and further civil development (Altman, 1998). Since then, significant research and growth have been seen with regards to civilian use of UAVs in areas of environmental application, such as pollution monitoring and weather forecasting, emergency response, including firefighting and tsunami watch, communications, particularly as relay services and cell phone transmissions, and commercial uses, the most popular of which being photography, agriculture, and mail (Angelov, 2012).

The increase in usage of UAVs for military and civil applications has raised questions on criteria that define safety regulations in terms of hardware, software, and environment, specifically in the area of sense and avoid (SAA). An SAA system can be considered an upgraded autopilot, providing assistance following flight patterns, waypoints and other mission requirements, but also reacting to situations in which hazardous outcomes must be detected, assessed, and resolved (Angelov, 2012).

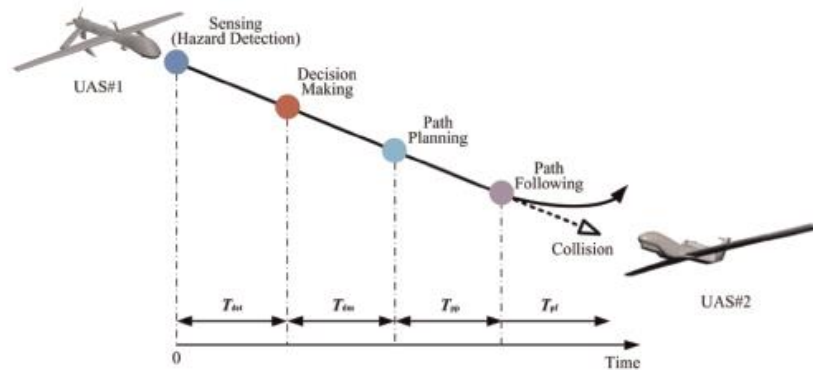


Figure 6: Example detection and response algorithm of a SAA UAV (Yu & Zhang, 2015)

Immediate and non-immediate hazards are the two primary capabilities that SAA systems must encompass. An immediate hazard will present a collision risk to a UAV with little to no time to react, requiring an immediate response from the system and perhaps deviation from the primary objective or standard flight pattern. The maneuver from this is usually significant but necessary to ensure the UAV's flight path is cleared from the hazard. A non-immediate hazard is usually detected well in advance, providing plenty of time for a slight adjustment to the UAV's flight path, resulting in little to no deviation from the original objective (Angelov, 2012). A number of sub-functions are required for an SAA system to ensure these detections and reactions are executed appropriately, typically following the steps:

1. Detecting any objects with potential for causing a hazard.
2. Tracking the motion of objects participating in any hazardous situation.
3. Prioritizing hazards into levels of importance.
4. Determining the timeline necessary for any potential maneuvers.
5. Determining a specific maneuver and path based on geometric or optical measurements.
6. Commanding and executing the maneuver.

These guidelines are not standard and will differ significantly in particular implementations of SAA platforms, but they provide the baseline for developing algorithms applicable to both airborne and ground based avoidance systems. For airborne SAA systems specifically, laying out any pseudo algorithm like the one above is a crucial stage in development, due to the inherent processing and power limitations presented with UAVS, especially smaller ones (Angelov, 2012). Any SAA algorithm will be very dependent on the hardware required and the type of system, specifically whether it is a cooperative or non-cooperative one.

Cooperative SAA methods utilize hardware that communicates between vehicles and ground stations, specifically providing data exchange of locations, trajectories, or other navigational critical information. The traffic collision avoidance system (TCAS) is currently the method of choice for manned aircraft avoidance (Billingsley, Kochenderfer, & Chryssanthacopoulos, 2011). With a maximum range of 160 km, TCAS systems create a virtual airspace map for an aircraft, alerting and providing instructions for the pilots if another aircraft intrudes on the airspace (Angelov, 2012).

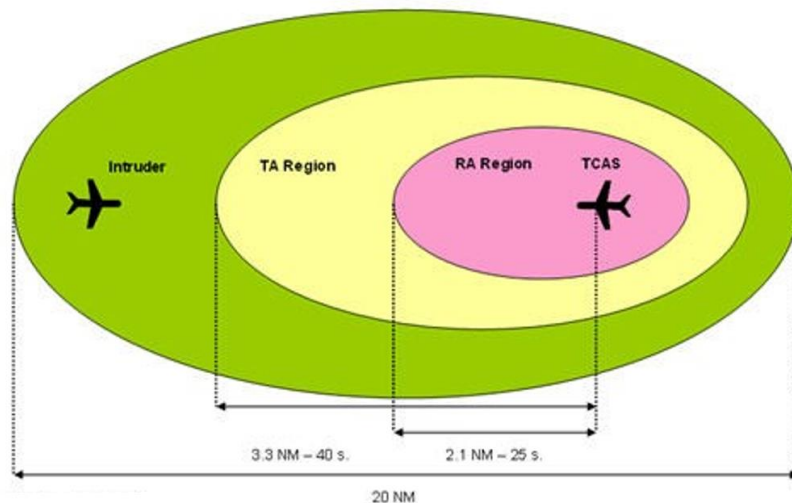


Figure 7: Traffic and Resolution Advisory Zones for TCAS (EuroControl, n.d.)

The aviation industry, specifically those interested in SAA, see TCAS as a viable option due to its current and widespread implementation, as well as its functionality for Visual Measurement Conditions (VMC) and Instrument Measurement Conditions (IMC) (Yu & Zhang, 2015). Automatic dependent surveillance – broadcast (ADS-B) is another similar, and proven, cooperative technology that provides participating receivers with aircraft global positioning system (GPS) coordinates, velocity, mission intent, and a specific identification value (Zeitlin & McLaughlin, 2007). While this is not yet a required or fully supported system, ADS-B can supply data exchange 240 km to and from ground stations, making it very favorable as a leading SAA technology. Both of these cooperative solutions, however, are limited in a few aspects in regards to use as SAA technology. TCAS is very capable with individual vehicles, but has not been proven to be able to incorporate multiple vehicles. ADS-B has the disadvantage of not working with ground or stationary objects (Yu & Zhang, 2015). The largest drawback of both these cooperative systems as SAA technologies stems from the necessity for all other vehicles to use the same technology to ensure collision trajectories are detected and avoidance maneuvers are made (Angelov, 2012).

Due to cooperative methods needing to be universally implemented, non-cooperative SAA systems have obtained a significant standing for application and research. Non-cooperative SAA methods have the advantage of not requiring communications or broadcasts between vehicles or ground stations. With use of technologies like synthetic aperture radar (SAR), Lidar, acoustic sensors, and electro-optical cameras, UAVs and other vehicles can independently detect and avoid other vehicles, buildings, and ground targets without any interaction with outside communications (Yu & Zhang, 2015).

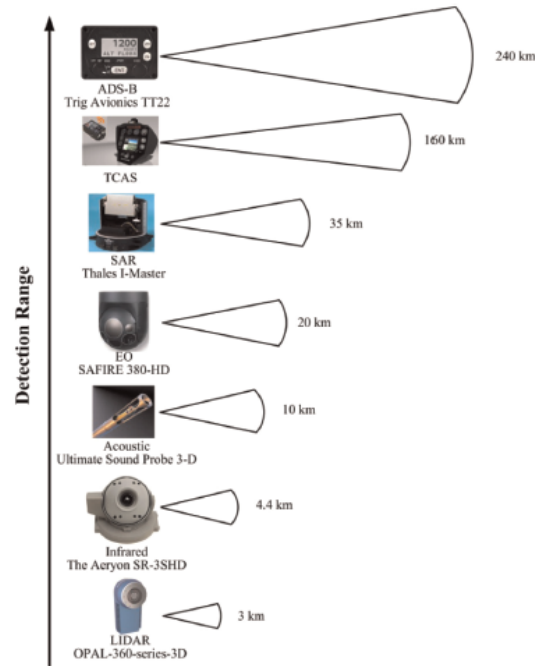


Figure 8: Examples of current SAA technologies and the common range limitations (Yu & Zhang, 2015)

These technologies, having been well established, present advantages and disadvantages, specifically in the area of physically sensing environments and obstacles surrounding them. SAR, for instance, has the capability to collect location, velocity, and size information about its environment or other vehicles using multiple radar pulses (Yu & Zhang, 2015). In particular, the NASA Jet Propulsion Lab (JPL) has proven the effectiveness of SAR for detection ranges of 16 km at a resolution capable of sensing miniature UAVs, and it provides General Atomic's Predator B with 220° azimuth and 30° elevation field of view (FOV) (Moses, Rutherford, & Valavanis, 2011). While it has a significant advantage of being SAA capable in all weather conditions, one fallback of SAR technology is the limitation of sending and receiving data at the speed of sound, making it difficult for real time implementation. In SAA systems, time becomes a mission, not a variable, meaning any critical detection and required avoidance maneuvers cannot be fully dependent on a time-consuming method (Yu & Zhang, 2015).

Lidar can also provide environment and obstacle location and distance information relative to the UAV, but can also be used to maintain a collection of point cloud data for virtual reconstruction of the environment. This powerful technology has been a tool for Carnegie Mellon University's Robotics Institute to reduce false positives from other SAA methods, and has the potential to be utilized as a way to "remember" the environment, a concept that can allow for more than fly-through navigation, but also loitering and reconnaissance missions (Geyer, Dey, & Singh, 2009).

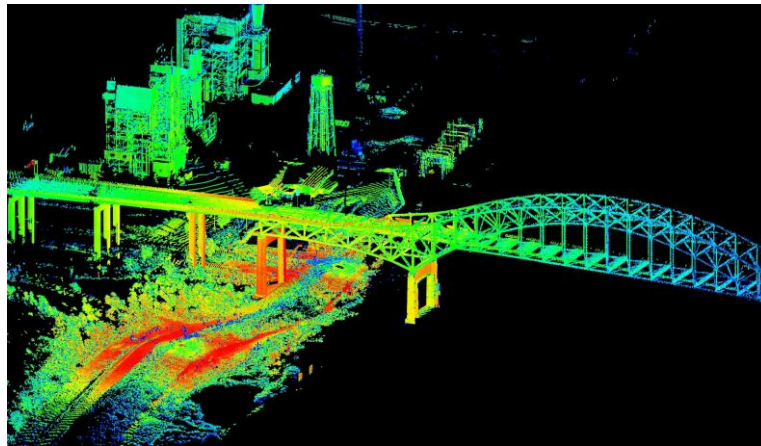


Figure 9: Example point cloud scene collected via Lidar measurements (Ryabinin, 2017)

With detection ranges typically between 200 m and 3 km and resolution capabilities of 5 mm, Lidar proves to be an excellent solution for SAA methods. However, a limiting factor for all of this potential is the limited FOV and large amount of data collected. Lidar typically have a relatively limited FOV in either azimuth or elevation, making its usefulness a function of specific scenarios. Collision avoidance scenarios requiring detection outside of the Lidar FOV places the UAV in an undesirable state. Also, the large amount of data that Lidar can collect poses a problem with data handling and processing to ensure the required computing and electrical power is available to allow for a real-time SAA system (Yu & Zhang, 2015).

Acoustic detection systems are not a particularly new technology, but their application for SAA methods is. The passive acoustic non-cooperative collision alert system (PANCAS), developed by Scientific Applications and Research Associates (SARA) Incorporated, is a leading project in the development of acoustic SAA detection systems (Yu & Zhang, 2015).

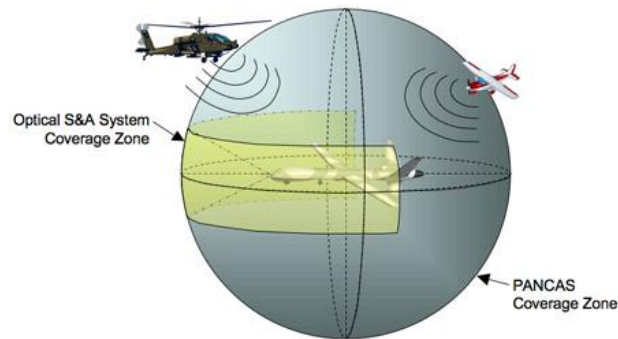


Figure 10: Relative zones of detection for optical and PANCAS sensing technologies (SARA Inc., 2012)

Using the noise generated by aircraft engines, propellers, and rotors, acoustic microphones can accurately determine intruding vehicle locations and velocities over a large range of frequencies within a sphere of detection, as seen in Figure 10. Because of its recent development, acoustic detection is limited in a number of ways, particularly with respect to its lack of range, poor performance in weather, and non-real time functionality.

Electro-optical (EO) devices use visible light to create an image of the surrounding environment or obstacles in view. The extent of this technology ranges from common, fixed, monocular vision cameras systems, like those used by recreational and professional photographers, to gimballed, panoramic cameras. The popularity of use of EO sensors as a primary or auxiliary SAA technology stems from its advantages of typically being low cost, lightweight, and requiring low power consumption.



Figure 11: Aerial, infrared image of wildfires, taken by RQ-4 Global Hawk (US Navy, 2008)

On a large scale, a notable implementation of EO technology for SAA is on Northrop Grumman's RQ-4 Global Hawk, cooperatively developed by the Air Force Research Laboratory and Defense Research Associates. This system utilizes three optical cameras to produce an effective $\pm 100^\circ$ azimuth and $\pm 15^\circ$ elevation FOV (Griffith, Kochenderfer, & Kuchar, 2008). Figure 11 shows an example image, provided by the United States Navy, of Californian wild fires captured by the primary center optical camera. On a smaller scale, optical flow sensors provide a promising and compact method for position determination and tracking at specific altitudes, similar to that in Gageik et. al., where waypoint navigation, position holding, and landing are all autonomously performed on a small quadrotor (Gageik, Strohmeier, & Montenegro, 2013).

EO cameras can, however, be greatly affected by weather conditions and may require an array of devices and sensor suites to be useful (Yu & Zhang, 2015). And while EO devices allow for a range of chosen detection algorithms, the most common of these require a comparatively large amount of data processing for vision-based tracking methods, and in the end still lack a critical piece of information: range. However, many

approaches can be implemented in order to overcome this lack of information with EO cameras.

Stereo vision utilizes two monocular EO cameras separated by a small horizontal distance from each other. This distance creates a horizontal offset, known as parallax, between corresponding features within the two images, which can in turn be used to estimate the range to the feature or object of interest. An example of this can be seen in the diagram presented in Figure 12. The capable range of stereo vision cameras relies on a number of factors including disparity distance, focal length of the lenses, and environment conditions, but their overall capabilities are limited to smaller distance in comparison to other SAA devices (Yu & Zhang, 2015).

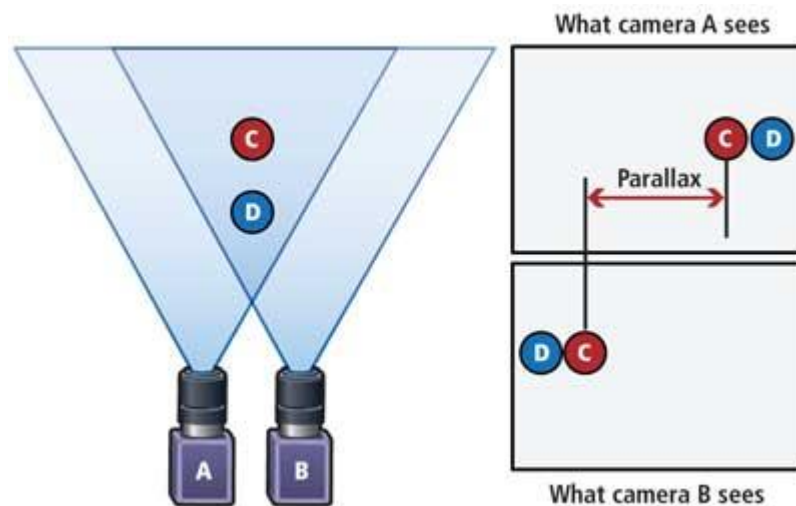


Figure 12: Stereovision parallax, demonstrated by the overlapping FOV of camera A and B (Lau, 2012)

A multitude of EO types can be used in this application, including infrared, demonstrated by Chen et al., in which infrared technology bridged the difficulty in detecting and tracking objects in subpar or low quality environments, something that proves difficult when using true-color EO devices (Chen, Cao, Wu, & Huang, January 2014).

Homography vision measurements provide an alternative solution that utilizes EO monocular cameras to derive camera translation, rotation, skewing, and scaling. When fixed to a vehicle body reference frame, these transformations can be directly applied to the vehicle, providing useful information for application of vision-based navigation, or position estimation (Zhao, 2012) An important limitation of this approach is that the homography relationship only holds for tracked features that lie in a common 3-D plane.

1.2. Vision-Based Proportional Navigation

Proportional navigation, commonly referred to as constant bearing decreasing range, is a method of utilizing line-of-sight (LOS) angles for the determination of imminent collisions. This technique has been used more commonly in applications of guidance laws for modern guided munitions, known as intercept guidance laws, in which the path to be taken by a primary vehicle is the one in which its distance with an ultimate target approaches zero. Intercept guidance laws that are based on monocular image plane line-of-sight angles use the change in line-of-sight to ensure that an intercept occurs. This proportional navigation (Pro-Nav) concept was originally discovered at sea, with ship navigators determining course corrections required for avoidance of other oncoming watercraft. This robust method of collision detection evolved rapidly for application in guided munitions, particularly with advancements in sensor and computing technology. Unlike stereo vision and other monocular vision techniques, at a basic level, vision-based proportional navigation requires much less computational power by utilizing optical flow, providing changes in positions along the image plane and in turn providing the information necessary to determine the LOS angles. For the objective of achieving intercept trajectories, the rate of change in the LOS angles provides the information

necessary to determine if a collision is imminent. By driving the LOS rate to zero and keeping it at zero, an interception or collision will occur, regardless of initial range or heading. Unfortunately, monocular vision-based guidance is unable to provide a critical piece of information that other sensors can: the range to targets. However, this shortcoming also plays a role in the advantage of Pro-Nav. The computational cost required for vision-based Pro-Nav is noticeably less than the cost required for most other guidance methods that determine and utilize range or distance information. This is also the point where this research seeks to utilize and apply Pro-Nav to its full potential. With an underlying low computational cost, this method can be applied to a very small vehicle with a large number of obstacles, targets, or features, providing a robust solution for incorporating dynamic and static environments into a resultant navigational path.

The investigation into intercept guidance has been present for a long time; however, it was not well defined until 1966, when Cornell University's Aeronautical Laboratory Inc. work that placed a classification to Pro-Nav. In its basic form, the paper published was related to missile/seeker interception with a target, using satellites as an example. Characteristics and limitations of Pro-Nav were presented and included optimal values for the Pro-Nav constant, N_H , best performance with and without continuous modulation of thrust, and inclusion of dead space with small LOS rate values (Murtaugh & Criel, 1966). Since then, more advanced methods of Pro-Nav intercept have been examined, notably in work by Cornell University and the United States Army. In this work, the attitude angle at waypoint interception for a guided reentry vehicle is controlled using a time-varying gain, rather than the constant gains defined by Murtaugh and Criel (Kim & Grider, 1973). With the implementation of a linear quadratic controller, Kim and Grider's

work has been distinguished as pioneering, and a notable stepping stone upon which others have expanded, including work by Kim, Lee, and Han, which was a particularly noteworthy implementation of Pro-Nav in which a bias was used to ensure a zero miss distance for a nonlinear analytical model of an intercepting vehicle (Kim, Lee, & Han, 1998).

In relation to this thesis, vision-based Pro-Nav using EO pixel information has been investigated in the context of developing intercept guidance laws for micro aerial vehicles (MAVs) (Beard, Curtis, Eilders, Evers, & Cloutier, 2007). In this paper, Beard et al. derive Pro-Nav guidance intercept commands based solely on pixel information. This contrasts with the work by Han, Bang, and Yoo, which manipulates Murtaugh & Criel's original Pro-Nav definition to fulfill avoidance criteria. Their proposed method utilizes a predicted zone of influence, which the primary vehicle must maneuver away from (Han, Bang, & Yoo, 2009). Sharma et al. and Trinh et al. also implement modified methods of Han, Bang, and Yoo's work in applications to path planning and planar navigation. While these works do not completely describe the amount of research and development that has been put towards Pro-Nav guidance and navigation, they are among the most notable. However, the one contrasting aspect that these works all have in common is the assumption of known range to targets or obstacle. These distances play a part in determining the time-to-target and using that information to develop successful intercept or avoidance commands.

This thesis extends previous work in regards to using Pro-Nav guidance law for intercept and avoidance guidance to a set of guidance laws that will enable a UAV to avoid multiple dynamic and static obstacles. These algorithms are studied within a

simulation environment, utilizing a six degree of freedom (DoF) dynamic model, without the use of target and obstacle range. The results of these simulations are quantified and analyzed for each set of guidance laws in order to determine characteristics and traits, allowing for successive variations and improvements in methodology and algorithmic process. The performance of these algorithms are quantified in terms of the minimization of collision probability.

The path of this thesis takes a developmental approach towards a final Pro-Nav avoidance law capable of avoiding dynamic and static obstacles using Pro-Nav concepts, without the use of range information. The approach systematically characterizes guidance law traits that contribute both positively and negatively, such that these can be taken into consideration for the next iteration of avoidance law development. The final form of the Pro-Nav guidance law is then implemented into a “camera-in-the-loop” simulation environment, where point-of-view images are captured in a realistic virtual environment, processed through an image feature detection algorithm, and applied to the design and implementation of Pro-Nav guidance laws to update the six DoF UAV model’s dynamic response.

2. UAV AND CAMERA MODELING

This thesis presents the development of collision-avoidance guidance laws based on monocular vision and the principle of proportion navigation. The approach uses two different mathematical UAV simulation models: one during the beginning and middle stages of guidance law design and testing, and one for final realization and finalization of the chosen law. Both models utilize the 12 aircraft equations of motion to allow for nonlinear, six degree of freedom (DoF) UAV dynamics to be computed in a simulation environment. A waypoint guidance law provides an underlying goal for the UAV to achieve during all simulations (with the exception of intercepting guidance cases). Altitude and velocity are maintained during maneuvering using PID controllers that maintain pitch and velocity by changing elevator and thrust values. The primary method of lateral maneuvering is uses a bank-to-turn heading controller, shown in Figure 13, which converts a commanded change of a selected heading parameter to a commanded bank angle, ϕ_{com} , with the proportional gain $K_{\phi_{com}}$. The controller then converts the commanded change in bank angle to an aileron command, δa_{com} , with a PID controller. It is important to note that $\Delta\psi_{com}$ is a resultant change in heading derived from the implementation of a Pro-Nav based guidance law.

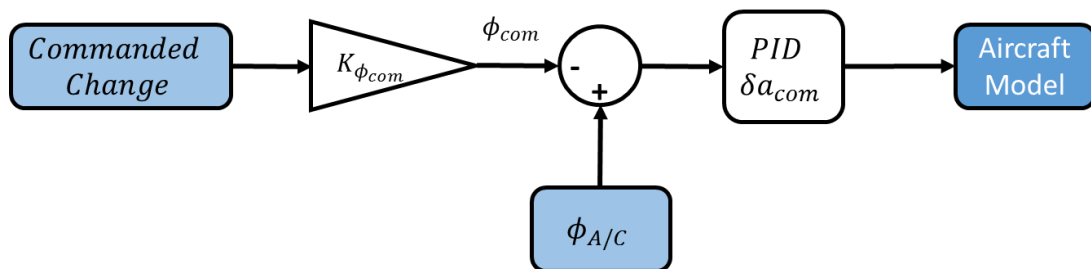


Figure 13: Diagram of Bank-to-turn control method for UAV simulations

While the modeling and control are similar for both UAVs, the initial conditions and geometric parameters differ greatly, which are discussed further below.

2.1. Medium-Scale, Propeller UAV Model

The initial aircraft model used for development of the Pro-Nav guidance laws represents a medium scale, propeller driven UAV. Some important aircraft geometric properties and flight conditions are shown in Table 1.

Table 1: Medium scale UAV geometric and flight parameters

Parameter	Value	Units
b_w	33.4	<i>ft</i>
A_w	6.06	
V_o	176.4	<i>ft/s</i>
h_o	1000	<i>ft</i>

In Table 1, b_w is the UAV wing span from tip to tip and A_w is the wing aspect ratio, which relates the wing span to the wing surface area S_w using the equation $A_w = \frac{b_w^2}{S_w}$. V_o and h_o are the UAV trim velocity and altitude, respectively. This model was provided by students in the Embry-Riddle Flight Dynamics and Control Research Laboratory (FDCRL) that includes PID tracking controllers for velocity, altitude and roll angle. Control tests were conducted initially using waypoint navigation to ensure the validity of the controller gains.

2.2. Small-Scale, RC Propeller UAV Model

The Condor Skywalker 1880 was used as a representative model for a small-scale, RC propeller driven UAV, and modified accordingly to achieve greater lateral dynamic stability and response. The geometric properties and trim flight conditions are shown in Table 2 for comparison with the medium-scale UAV model.

Table 2: Small-scale UAV geometric and flight parameters

Parameter	Value	Units
\mathbf{b}_w	5.43	<i>ft</i>
\mathbf{A}_w	5.06	
\mathbf{V}_o	45.3	<i>ft/s</i>
\mathbf{h}_o	23.0	<i>ft</i>

This aircraft model was also provided by FDCRL students; however, the control laws were designed specifically for this research. Like the medium-scale UAV model, the control laws utilize PID controllers for altitude, velocity, and bank control, designed via the Ziegler-Nichols PID tuning method (Nelson, 1998).

2.3. Camera Model

A projective, pinhole camera model is the projection of three-dimensional geometry of the scene via a pinhole camera model, and can be represented by using three primary coordinate systems: the camera frame C (in some instances also denoted by M), body-fixed frame B , and navigational frame E . The use of a camera model in this thesis can be broken up into two primary methods. The first model, implemented for point-mass simulations, uses given point-mass locations in the inertial reference frame, also known as the north-east-down (NED) frame. The reference frame is a local coordinate system in which x and y are aligned with the magnetic north and east poles of earth, with z oriented positively in the downwards direction. This camera modeling method computes the vector projection normalized about a focal-axis. For this case, the camera-fixed focal-axis is aligned with the NED x -axis, resulting in the following normalized vector:

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix}_{p/c}^E = \underline{r}_p^E - \underline{r}_c^E \rightarrow \begin{Bmatrix} 1 \\ \left(\frac{y_{p/c}^E}{x_{p/c}^E}\right) \\ \left(\frac{z_{p/c}^E}{x_{p/c}^E}\right) \end{Bmatrix}^E \quad (1)$$

where \underline{r}_p^E and \underline{r}_c^E denote the inertial position of the target feature point and the UAV center of mass, respectively. It is important for the point-masses (or feature points) to be expressed with respect to the inertial frame when implementing Pro-Nav guidance laws. This means a few steps are not required when utilizing this camera model method, since the point locations are already provided in the inertial reference frame. However, multiple additional processing steps are required when using a real camera/EO device or a virtual reality environment.

In real-world or virtual scenarios and simulation, the latter of which will be presented later in this thesis, the locations of feature points are provided in the two-dimensional camera frame, also known as the pixel frame. These pixel coordinates are converted to the inertial reference frame via the use of a camera calibration matrix, a camera-to-body frame rotation matrix, and a body-to-inertial frame rotation matrix, commonly known as the direction cosine matrix (DCM) with respect to the aircraft. This calculation can be represented as:

$$x_{p/c}^E = R_c^E K x_{p/c}^c \quad (2)$$

In Equation (2), x^c and x^E are the feature point relative to the camera coordinates with respect to the camera-pixel frame and the inertial frame, respectively. K is the intrinsic calibration matrix that transforms the pixel coordinates into real-world camera coordinates. These real-world coordinates are values with physical units rather than pixel

dimensions. A physical representation of this conversion can be seen in Figure 14.

$$K = \begin{bmatrix} \alpha_x & \gamma & u_o \\ 0 & \alpha_y & v_o \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

Equation (3) shows the specific form of the intrinsic calibration matrix, in which α_x and α_y represent the focal length in terms of image pixels, and γ provides a coefficient to compensate for any skew angle between the x and y axis of the focal plane. u_o and v_o are the principle point, which is the center of the image with respect to the pixel coordinates.

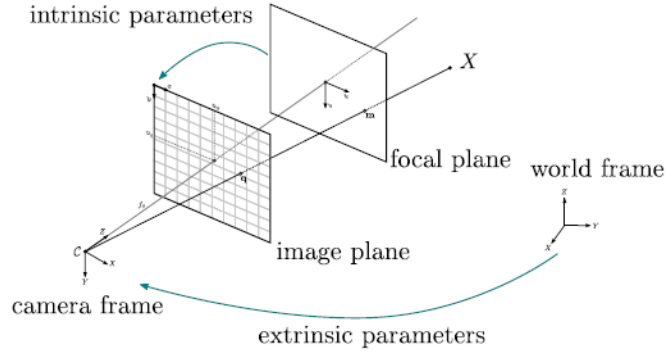


Figure 14: Visual representation of image and focal plane related by intrinsic matrix (OpenMVG, 2013)

The calibrated coordinates are then transformed by the previously mentioned DCM and camera-to-body rotation, represented by R_C^E , which will be defined further in the next section of this thesis.

2.4. Simulation Environment

MATLAB/Simulink is the primary simulation environment for the work presented in this thesis. MATLAB coding is used alongside Simulink block diagrams to simulate the guidance and navigation of a six degree-of-freedom aircraft model. The diagram in Figure 15 shows the point-mass simulation process, in which the aircraft model sends its position and orientation information to a vision measurement block so that the vision

based measurements of the obstacles relative to the UAV can be derived. The guidance commands produced by this block then send information for the tracking PID controllers to produce a commanded control input that the UAV model can interpret.

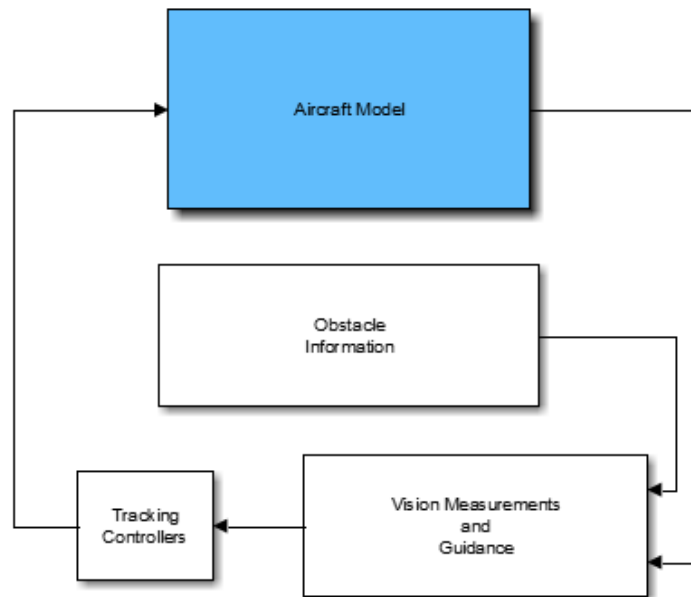


Figure 15: Diagram of Simulink block configuration for point mass simulations

The aerodynamics of the UAV model are shown in Figure 16, where a component buildup method using aerodynamic coefficients is implemented to determine the resultant aerodynamic forces and moments. Although there is more complexity to the UAV model, the application of these aerodynamic coefficients contributes largely to the dynamics exhibited by the aircraft. Each coefficient has a subset of contributing derivatives, such as from the wing, elevator, or aileron.

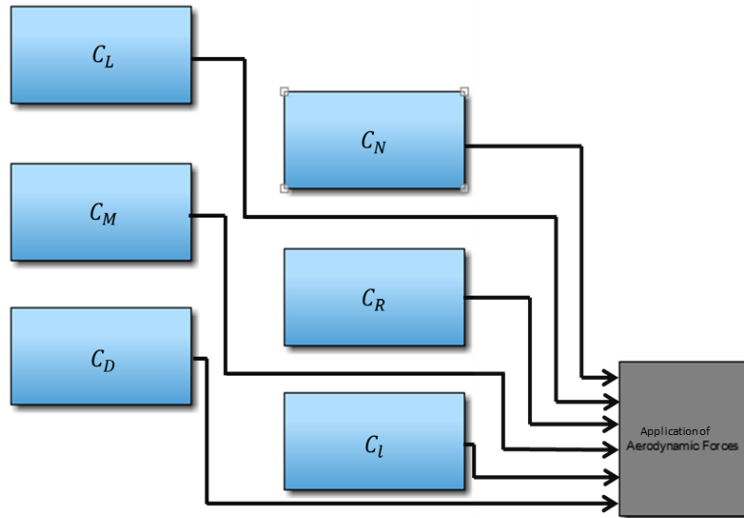


Figure 16: Aerodynamic component buildup for six DoF model

Figure 17 shows a simulation diagram similar to that seen in Figure 15, but applicable to the simulation supporting virtual environment simulations. In this diagram, communication must be made to an application outside of the simulation environment. Information must then be received from the application before the simulation can continue. This information is then used to produce a guidance command and in turn, control the aircraft. This simulation configuration will be discussed further in depth in Chapter 8.

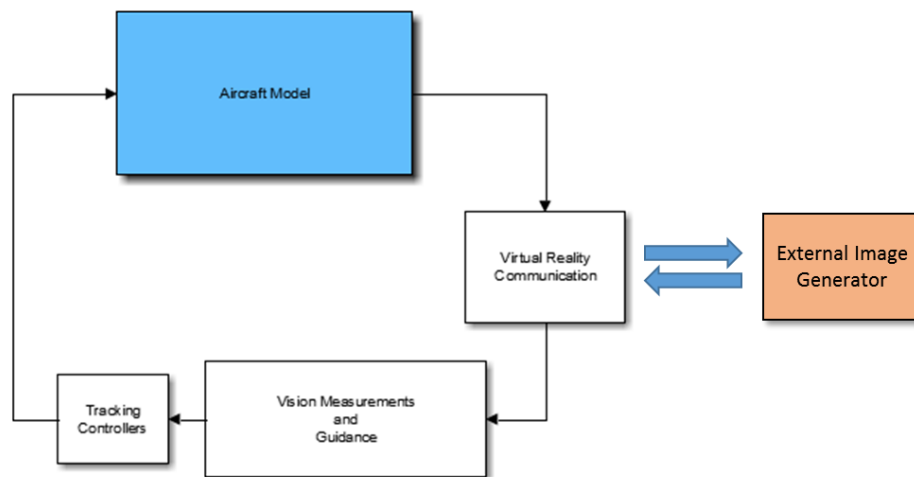


Figure 17: Diagram of Simulink block configuration for virtual reality simulations

3. PRO-NAV INTERCEPT LAW

The Pro-Nav intercept laws provide a mathematically derived heading and flight angle command to the UAV, which in turn initiates aileron and elevator control commands to avoid the obstacle. The commanded heading and flight path angles are determined using the information provided in the monocular image plane, specifically the horizontal and vertical location of the target(s) relative to the center of the image plane.

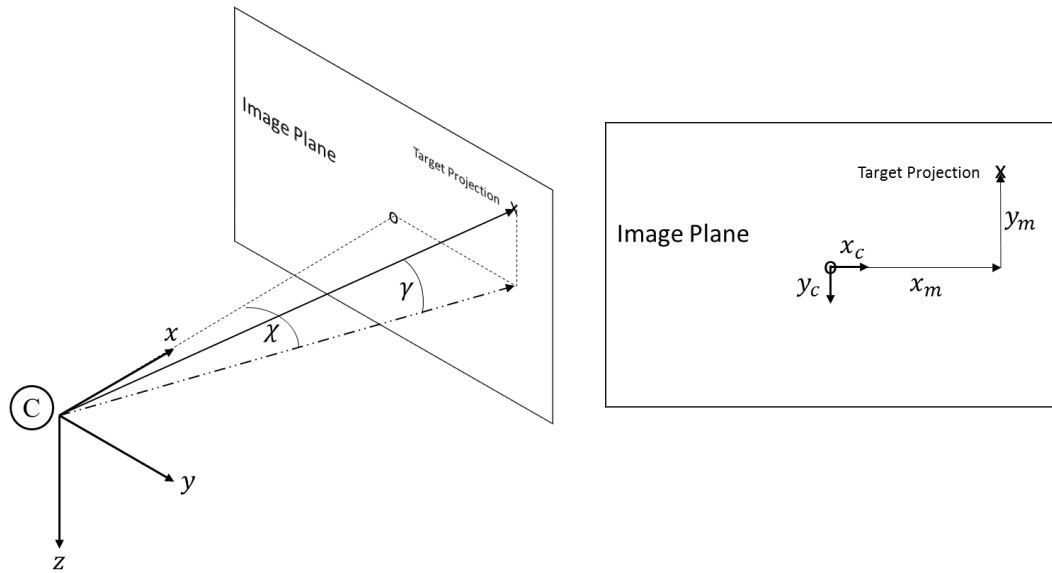


Figure 18: LOS Angles with respect to the Image Plane.

On the left of Figure 18, the three-dimensional camera frame, with respect to the image plane, shows the horizontal and vertical line of sight angles, χ and γ , derived from the projected target locations, x_m and y_m on the image plane in the camera reference frame.

$$\hat{\underline{p}}_{p/c}^E = \begin{Bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{Bmatrix}^E = R_c^E K \hat{\underline{p}}_{p/c}^C = R_c^E K \begin{Bmatrix} x_m \\ y_m \\ 1 \end{Bmatrix}^C \quad (4)$$

$$R_C^E = R_B^E R_C^B = (DCM)^T \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \begin{matrix} \text{Fixed} \\ \text{Camera} \end{matrix} \quad (5)$$

$$R_B^E = (DCM)^T = \begin{bmatrix} \cos \theta \cos \psi & \sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi & \cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi \\ \cos \theta \sin \psi & \sin \phi \sin \theta \sin \psi + \cos \phi \cos \psi & \cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi \\ -\sin \theta & \sin \phi \sin \theta & \cos \phi \cos \theta \end{bmatrix} \quad (6)$$

This derivation, shown in Equations (4) through (6), is obtained through the rotation matrix R_C^E and assumes the camera is fixed relative to the UAV. For the cases studied in this thesis, the camera is assumed to be pointed in the body-x direction; therefore, the R_C^B defined in Equation (5) simply reorders the camera axes to coincide with the body-fixed axes. The resulting LOS angles, along with their respective LOS angular rates, can then be derived as follows:

$$\chi_{LOS} = \tan^{-1} \left(\frac{\hat{y}_{p/c}^E}{\hat{x}_{p/c}^E} \right) \rightarrow \dot{\chi}_{LOS}(t_k) = \frac{\chi_{LOS}(t_k) - \chi_{LOS}(t_{k-1})}{\Delta t} \quad (7)$$

$$\gamma_{LOS} = \sin^{-1} \left(\frac{\hat{z}_{p/c}^E}{\sqrt{\hat{x}_{p/c}^E{}^2 + \hat{y}_{p/c}^E{}^2}} \right) \rightarrow \dot{\gamma}_{LOS}(t_k) = \frac{\gamma_{LOS}(t_k) - \gamma_{LOS}(t_{k-1})}{\Delta t} \quad (8)$$

Equations (7) and (8) form the basis for the laws implemented in an intercept based simulation, in which a medium-scale UAV seeks to intercept and “collide” with a target moving at a constant velocity and heading. For an intercept to be made, the simulated UAV uses Equation (9) and (10) to initiate commanded heading and flight path angles. These longitudinal and lateral commands can vary, based on the method of implementation, but are presented below for a bank-to-turn method of lateral navigation.

$$\dot{\psi}_{com} = N_H \dot{\chi}_{LOS} \rightarrow \begin{cases} \Delta\psi_{com} = N_H \dot{\chi}_{LOS} \Delta t_T \\ \Delta\phi_{com} = k_\phi \Delta\psi_{com} \end{cases} \quad (9)$$

$$\dot{\gamma}_{com} = N_v \dot{\gamma}_{LOS} \rightarrow \Delta\gamma_{com} = N_v \dot{\gamma}_{LOS} \Delta t_T \quad (10)$$

Equations (9) and (10) are the proportional guidance commands that use proportional gains N_H and N_v to provide a commanded rate of change of yaw and vertical flight path angles. These constant are the proportional navigation constants that enable the concept of Pro-Nav to be successful. The optimal values have been previously been determined to be optimized at three (Murtaugh & Criel, 1966). Calculating the rate of change in LOS angles using finite differences allows the flight commands to be referenced to where the target will be in the next iteration. Results of the implementation of this intercept guidance controller can be seen in Figure 19 - Figure 21, in which a few representative examples of the numerous scenarios simulated are presented with an active intercept controller for the UAV and different headings for the target.

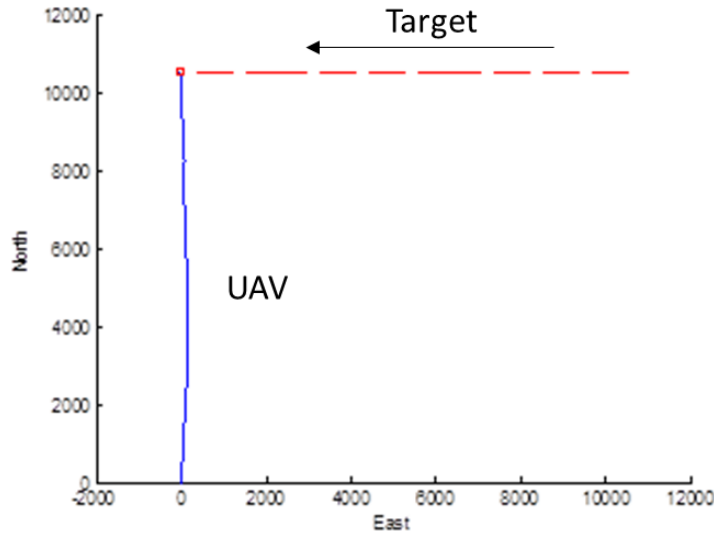


Figure 19: Example 1 - UAV Intercept path (units in feet).

Figure 19 displays an engagement scenario in which the target has an initial heading

of -90° , starting from one mile north and one mile east of the UAV. Both the UAV and target obstacle are traveling at the same initial velocity, which is the case for the rest of the scenarios as well. The similar velocities for the scenario in Figure 19 would mean the optimal UAV intercept path would be to continue on a constant north heading. The deviation seen in the path shows the intercept guidance law at work, generating control inputs based on the LOS angular rates rather than relative distances and velocities, which are assumed to be unknown.

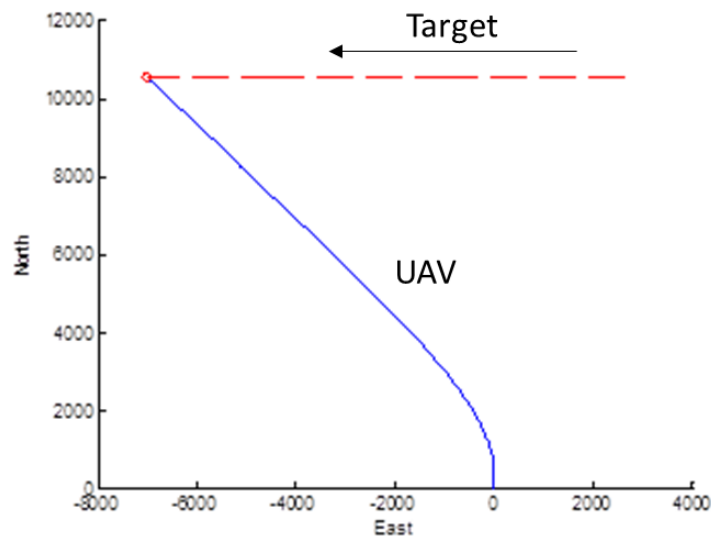


Figure 20: Example 2 - UAV Intercept path (units in feet).

Figure 20 shows another engagement using the intercept controller, similar to the one presented in Figure 19, except that the target starting location is one mile north and half a mile east relative to the UAV. The controller's effectiveness is seen almost immediately by the UAV's deviation westward to intercept the target.

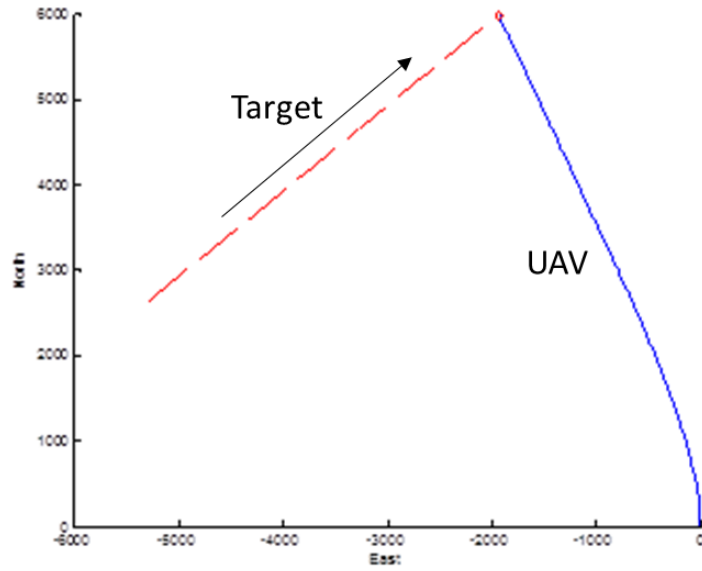


Figure 21: Example 3 - UAV Intercept path (units in feet).

The scenario shown in Figure 21 differs from the previous two in that the obstacle has an initial starting point a half mile north and one mile west of the UAV, with a heading of 26.6° . This scenario aids in verifying the functionality of the controller by having a target traveling from the opposite side as the previous cases. This shows that the sign of the calculated $\dot{\chi}$ is accounted for and interpreted correctly, regardless of which direction the represented target is moving across the image plane.

4. PRO-NAV AVOIDANCE LAW – THRESHOLDING APPROACH

Extending the intercept guidance law to avoidance based guidance laws requires a modified set of control laws. Specifically, the modified guidance laws must first determine if a collision avoidance maneuver is required and then compute the magnitude and direction of the avoidance maneuver that should be made. These two criteria are accounted for by using a threshold activation controller and a piecewise directional controller.

The threshold activation controller compares the absolute value of the LOS angular rates with a selected threshold value. A zero LOS angular rate implies an imminent collision between the UAV and target, so that setting a threshold value to zero would only ensure any avoidance distance, no matter how small, would be acceptable. Therefore, a small threshold value is implemented to command an avoidance maneuver for LOS angular rates close to zero. When considering the size of a medium size UAV, an avoidance larger than 20 *ft* could be considered an acceptable distance, although this would depend on the size and speed of the other vehicle. The distance to the target is assumed to be unknown, but the threshold parameter determines the degree of resulting avoidance distance and sensitivity of the avoidance law. This is an area that was first investigated, with the results summarized in Table 3 - Table 5.

In regards to commercial flight, when an incident occurs in which an avoidance maneuver is required, the FAA has regulated lateral maneuvering to be the primary course of action. The direction of this lateral maneuver, however, is not defined. The avoidance law implemented in this thesis uses the following logic, in which $\dot{\psi}_{com}$ is selected to be a constant value:

$$\psi_{\text{com}} = \begin{cases} \dot{\psi}_{\text{com}}\Delta t_c, & \dot{\chi} \leq 0 \\ -\dot{\psi}_{\text{com}}\Delta t_c, & \dot{\chi} > 0 \end{cases}$$

This piecewise directional controller uses the logic of traveling behind the intercepting target, relative to its velocity and heading. Therefore, a negative LOS angular rate results in a positive commanded heading, while a positive LOS angular rate produces a negative commanded heading. Zero was included in the first definition of the piecewise function to favor deviating behind the collision threat. Any zero LOS rate requires some form of action to be taken, and without a direction of the intercepting obstacle, any reaction in a direction provides a better choice than not maneuvering at all.

Several dynamic avoidance scenarios were used to demonstrate and further develop the avoidance controller. These scenarios are similar to those used for the intercept simulations. The initial conditions for the dynamic obstacles are varied in terms of obstacle heading, starting location, and velocity. A major difference between these simulations and the intercept simulations is that these initial conditions are modified to ensure an imminent collision if an avoidance maneuver is not made. Figure 22 - Figure 24 show the first configuration set of three simulations. In this set, the obstacle velocity and heading are held constant at 176 ft/s and -90° , respectively, while the starting location of the obstacle is varied proportionally to ensure that an imminent collision along the north axis will occur if no avoidance maneuver is performed. By varying the range alone for this set of simulations, insight can be gained on the effectiveness of range on avoidance distance over a variety of threshold $\dot{\chi}$ values. Figure 22 shows the first of the three simulations varying range and holding all other variables constant. For this simulation, the initial position of the obstacle is one mile north and one mile east of the

UAV. The avoidance controller produces a deviation in the UAV path that becomes noticeable at a range of 4,000 *ft* for each $\dot{\chi}$ threshold value.

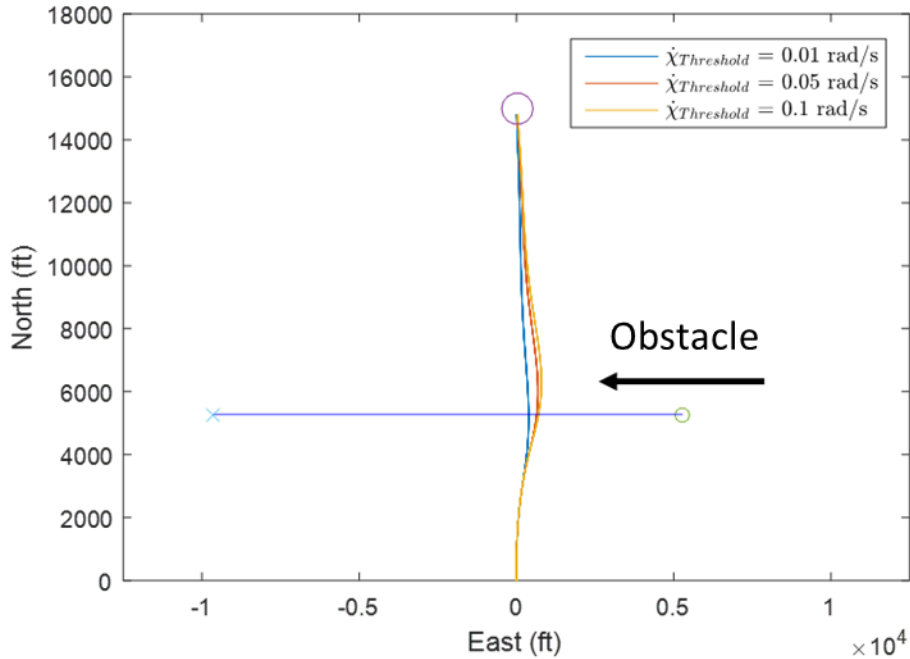


Figure 22: UAV Single Avoidance Path – Effect on Distance – Position = (1,1) mile

Figure 23 presents a similar scenario for investigating the effect of distance on avoidance distance using the Pro-Nav avoidance controller. The obstacle location, however, is set to two miles north and two miles east, ensuring that an intercept will occur two miles in front of the UAV along the north axis with no avoidance controller active. The simulations were computed for threshold values of 0.005, 0.01, and 0.02 *rad/s* and show avoidance maneuvers implemented.

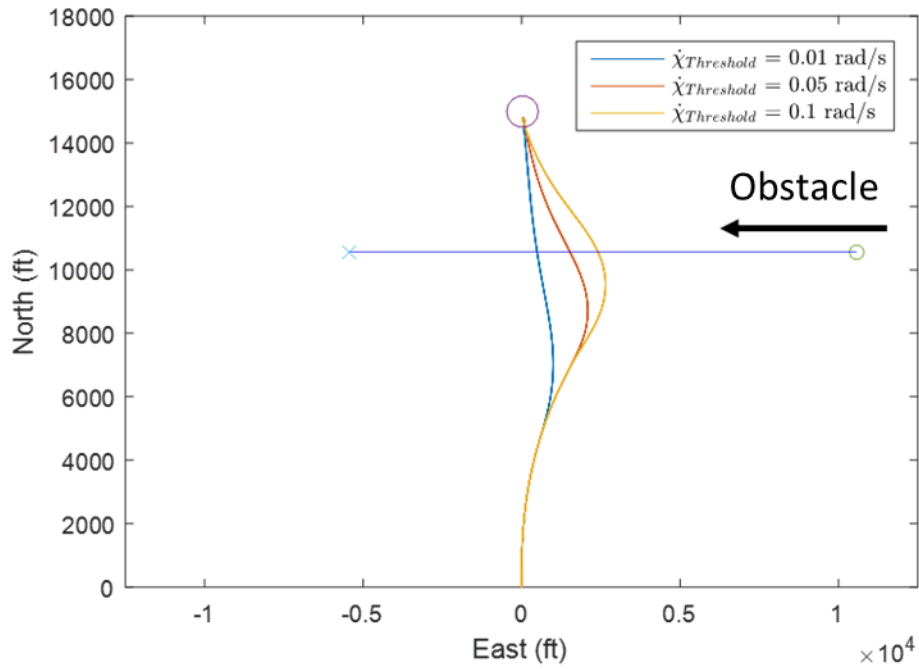


Figure 23: UAV Single Avoidance Path – Effect on Distance – Position = (2,2) mile

In Figure 24, the obstacle is set with an initial heading of 90° , similarly to the simulations in Figure 22 and Figure 23, but with a starting position of half a mile north and half a mile east. This is the third simulation in the set of three to investigate the effect of distance on the avoidance controller's capability of ensuring an imminent collision is prevented. This path does not lead to a collision without active avoidance, but does not follow the trend of the previous scenarios, initiating an avoidance maneuver at around $2,000 \text{ ft}$ for each simulation of $\dot{\chi}$ threshold values of $0.005, 0.01, \text{ and } 0.02 \text{ rad/s}$.

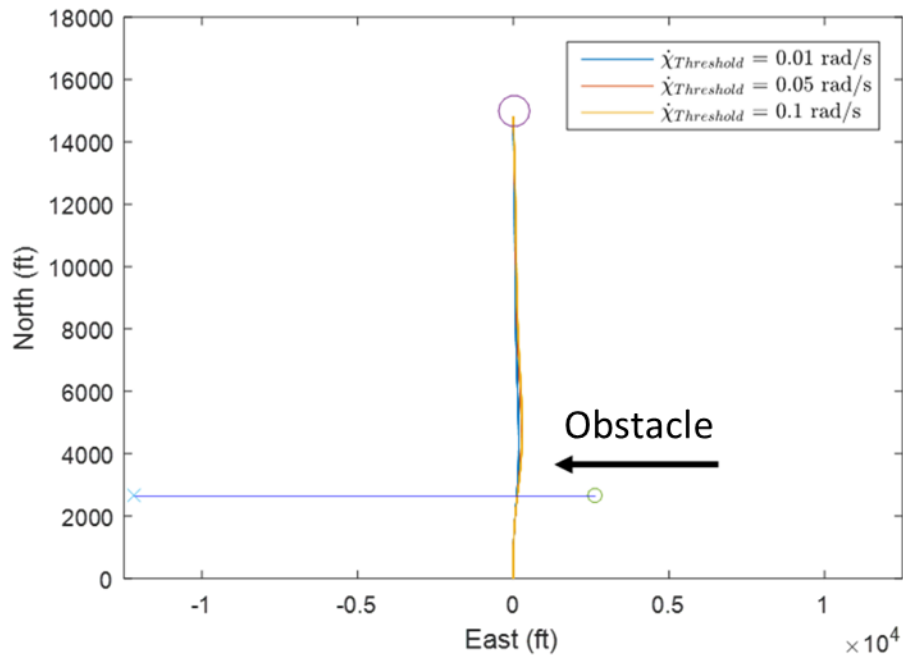


Figure 24: UAV Single Avoidance Path – Effect on Distance – Position = (0.5,0.5) mile

Table 3 summarizes the results for the effect of obstacle range on avoidance distance from the obstacle. One noticeable trend for all avoidance thresholds simulated is an increase in avoidance distance with an increase in obstacle starting position and intercept distance. However, the increase in avoidance distance with the increase in $\dot{\chi}$ threshold is not significant at shorter obstacle distances but does become effective at larger distances. Overall, it can be said that the guidance controller is more effective at avoiding obstacles further away than obstacles that are closer, with the value of $\dot{\chi}$ threshold being more significant at further distances.

Table 3: UAV Single Avoidance, Effect of Position, $V=176$ ft/s

	$\dot{\chi}_{Threshold}$ (rad/sec)			Miss Distance (No Guidance)
	0.01	0.05	0.1	
Position (ft, ft)	Miss Distance (Guidance On)			
(2,640, 2,640)	78 ft	90 ft	91 ft	1.2 ft
(5,280, 5,280)	289 ft	457 ft	487 ft	1.2 ft
(10,560, 10,560)	482 ft	1,661 ft	2,175 ft	1.6 ft

The second set of simulations, the results of which are shown in Table 4, explore the effect of velocity on the UAV’s avoidance distance from the obstacle. For the simulations, the obstacle is set with an initial heading of -90° (due west), and has a varying velocity and initial east position. The position is increased proportionally with the speed to ensure a collision occurs along the north axis if an avoidance maneuver is not performed. An overarching interpretation of the results in Table 4 is that higher obstacle velocity, relative to the UAV velocity, leads to larger avoidance distances between the obstacle and UAV.

Table 4: Single Avoidance Effect of Velocity, $V=88, 176, 264$ ft/s

	$\dot{\chi}_{Threshold}$ (rad/sec)			Miss Distance (No Guidance)
	0.01	0.05	0.1	
Position (ft, ft)	Miss Distance (Guidance On)			
(5,280, 2,640)	286 ft	530 ft	587 ft	1.2 ft
(5,280, 5,280)	289 ft	457 ft	487 ft	1.2 ft
(5,280, 10,560)	282 ft	395ft	411ft	1.3 ft

The scenario results presented in Table 5 are based on obstacle engagements where the UAV is not initially placed on a collision path. This means that the miss distances between the obstacle and UAV with no active avoidance in Table 5 are already sufficient, such that any added control input can be considered unnecessary, unlike the scenarios in Table 3, which required avoidance maneuvers.

Table 5: Single Avoidance No Intercept, $V=88, 176, 264$ ft/s

	$\dot{\chi}_{Threshold}$ (rad/sec)			Miss Distance (No Guidance)
	0.01	0.05	0.1	
Position (ft, ft)	Miss Distance (Guidance On)			
(2,640, 2,640)	1,867 ft	1,994 ft	2,075 ft	1,870 ft
(5,280, 5,280)	2,368 ft	2,565 ft	2,562 ft	2,361 ft
(10,560, 10,560)	1,469 ft	1,605 ft	1,654 ft	1,464 ft

The simulation sets presented in Table 3 - Table 5 show that the application of the Pro-Nav guidance law was successful in providing acceptable avoidance distance between the UAV and obstacles when initially set for intercepting paths. While some unnecessary maneuvering does occur, specifically when no collision is present, the amount of overcompensation is not considered to be excessive.

5. PRO-NAV AVOIDANCE LAWS – MOST IMMINENT THREAT

With a proven method for single obstacle avoidance using LOS rate thresholds, the next step was to apply this controller to a more complicated scenario. Introducing a second intercepting target adds an element of risk when executing an avoidance maneuver. Successful avoidance of one moving obstacle may lead to an intercepting path with the other obstacle. In this section, the avoidance guidance laws are applied in scenarios involving multiple collision threats.

To investigate these scenarios, the simulation environment first introduces two constant velocity obstacles at independent starting locations and headings. Due to the second obstacle being present in the simulation, the projection method for simulating a camera view exhibits two feature points, meaning the implemented guidance law has to be updated to process LOS angular rates for both obstacles. The modified law's primary method of deciding on an avoidance maneuver is prioritization of the smallest LOS rate value at time t_k between both targets. This most imminent threat strategy entails maneuvering to avoid the target with the smallest LOS rate, which is indicative of the most serious collision threat, while not reacting to the other obstacles in the scene. This strategy is the applied to a scene with multiple static obstacles.

5.1. Dynamic Avoidance

Several multiple engagement scenarios were simulated in which constant velocity obstacles were placed at starting positions and headings that would ensure an intercept is inevitable with at least one of them by the UAV. In some cases, one obstacle is positioned to intercept or have a near-intercept situation with the UAV as it attempts to avoid the other obstacle. For each scenario, the controller was set to actively avoid each

target individually for two individual simulations, followed by a third simulation with multi-avoidance active.

In Figure 25, both obstacles have an initial heading of -90° and starting positions such that they result in a collision with the UAV along the North axis. On the left side of the figure, a single avoidance simulation was performed in order to determine the miss distance between both obstacles and the UAV while attempting to avoid the closer of the obstacles, Obstacle 1.

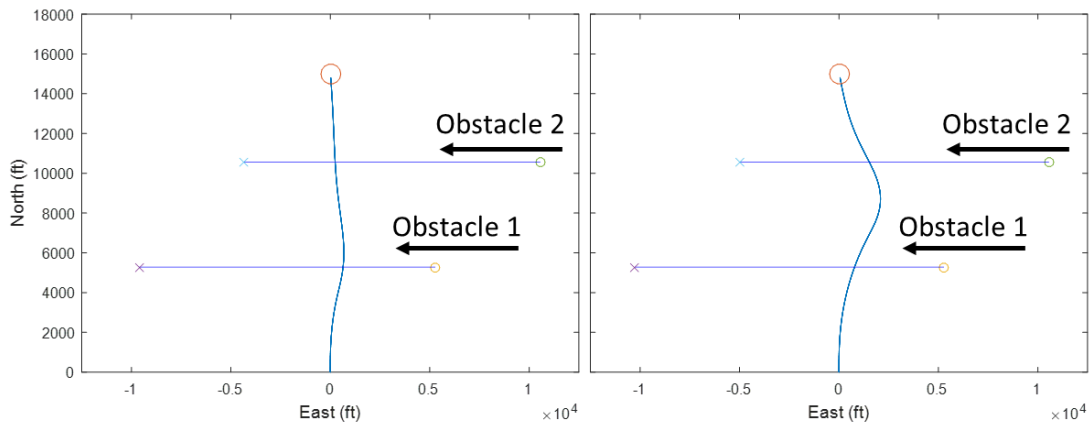


Figure 25: UAV Individual Avoidance Paths (ft) - Obstacle 1 and 2 North Intercept - $\dot{\chi} = .01 \frac{deg}{s}$

Table 6 shows the obstacle offsets and miss distances from the simulations in Figure 25. For the first simulation, in which Obstacle 1 is being avoided without regards to Obstacle 2, there is an avoidance distance of 457 ft between Obstacle 1 and the UAV. Obstacle 2 is avoided by 237 ft , but it is important to note that this avoidance is only inherited by the intent to avoid Obstacle 1. For the second simulation, on the right of Figure 25, active avoidance of Obstacle 2 only resulted in an avoidance distance of 506 ft for Obstacle 2 and $1,661 \text{ ft}$ for Obstacle 1, noting again the miss distance between Obstacle 1 and the UAV is inherited.

Table 6: UAV Individual Avoidance Data - Obstacle 1 and 2 North Intercept

	Obstacle 1 Miss Distance (ft)	Obstacle 2 Miss Distance (ft)
Obstacle 1 Avoidance	457	237
Obstacle 2 Avoidance	506	1,661

The simulations for which the multi-avoidance controller was enabled can be seen in Figure 26. These simulations are represented by the UAV paths for a chosen set of $\dot{\chi}$ threshold values, including 0.01, 0.05, and 0.1 $\frac{deg}{s}$. The resulting path resembles that of a superposition of the simulations in Figure 25, with an increase in path deviation because of the increased $\dot{\chi}$ threshold.

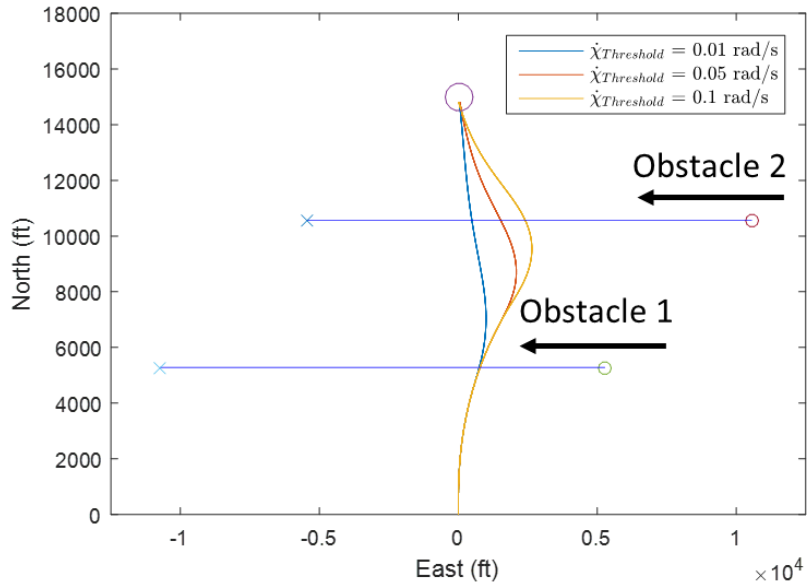


Figure 26: UAV Multi-Avoidance Paths (ft) - Obstacle 1 and 2 North Intercept

Justification for the increasing path deviations in Figure 26 can be seen in Table 7, where the avoidance distance for both Obstacles 1 and 2 are larger the 400 ft. This avoidance distance differs slightly from the original $\dot{\chi}$ threshold avoidance controller for

single obstacle avoidance; however, the results support the previous trend of increasing $\dot{\chi}$ threshold values from the single avoidance cases.

Table 7: UAV Multi-Avoidance Data - Obstacle 1 and 2 North Intercept

$\dot{\chi}$ Threshold (rad/s)	Obstacle 1 Miss Distance (ft)	Obstacle 2 Miss Distance (ft)
0.01	502	482
0.05	506	1661
0.10	506	2175

For the single obstacle avoidance controller, the sensitivity of the controller was investigated with a scenario in which the obstacle was not set on an intercepting path. Figure 27 shows the avoidance path for the first of two similar multi-avoidance scenarios. Obstacle 2 is set to not intercept the UAV along the north axis, but instead along the path taken to avoid Obstacle 1 only. By doing this, the avoidance sensitivity can be assessed along with the reactivity of the controller. Because avoiding Obstacle 1 will place Obstacle 2 on a collision course, it is critical to see what kind of avoidance maneuver will be taken, if at all, and whether it is enough to avoid both obstacles with enough distance without overcompensating.

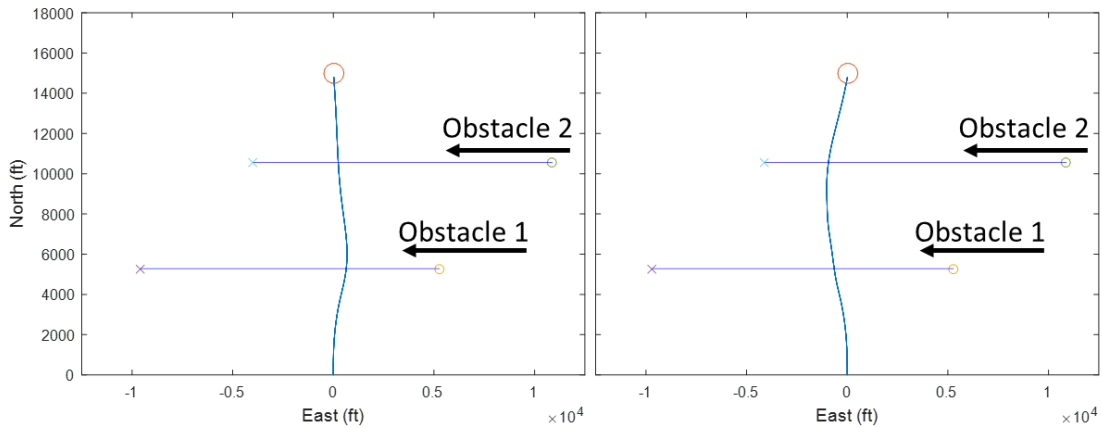


Figure 27: UAV Individual Avoidance Paths (ft) - Obstacle 2 Offset Intercept - $\dot{\chi} = .01 \frac{deg}{s}$

In Table 8, the 457 *ft* miss distance of Obstacle 1 and the 1.3 *ft* miss distance of Obstacle 2 correlates to the simulation on the left in Figure 27, where the avoidance controller is active for Obstacle 1. The small avoidance distances between Obstacle 2 and the UAV can be interpreted as an intercept and the focus of this simulation. On the right of Figure 27, the avoidance of Obstacle 2 leads to a similar avoidance of Obstacle 1 and 2, the distance being 443 *ft* and 759 *ft*, respectively. The avoidance distance between Obstacle 1 and the UAV is only an inherited aspect, due to the avoidance controller actively avoiding only Obstacle 2.

Table 8: UAV Individual Avoidance Data - Obstacle 2 Offset Intercept

	Obstacle 1 Miss Distance (ft)	Obstacle 2 Miss Distance (ft)
Obstacle 1 Avoidance	457	1.3
Obstacle 2 Avoidance	443	759

Figure 28 shows the multi-avoidance path of the UAV for a North intercept of Object 1, with a starting position East of the UAV, and an East offset intercept of Object 2, initially starting East as well. The paths deviate similarly to those in Figure 26, having a larger deviation with a larger χ threshold value, and resembling the superposition of the individual avoidance maneuvers in Figure 27.

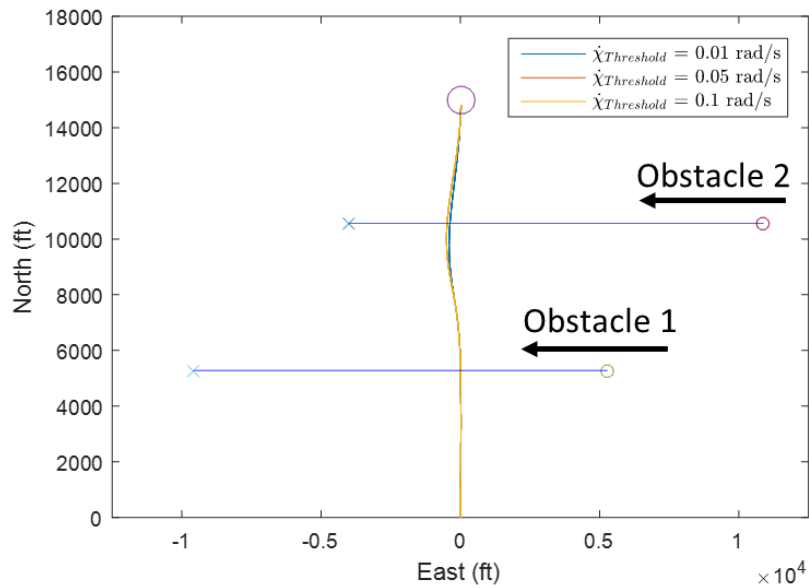


Figure 28: UAV Multi-Avoidance Paths (ft) - Obstacle 2 Offset Intercept

The miss distances shown in Table 9 correspond to the avoidance paths shown in Figure 28. An increase in $\dot{\chi}$ threshold, like previous simulations, increases the avoidance distance, but only slightly with regards to Obstacle 2, with a maximum closest distance of 515 ft between the Obstacle 2 and UAV for a $\dot{\chi}$ threshold of $0.05 \frac{deg}{s}$.

Table 9: UAV Multi-Avoidance Data - Obstacle 2 Offset Intercept

$\dot{\chi}$ Threshold (rad/s)	Obstacle 1 Miss Distance (ft)	Obstacle 2 Miss Distance (ft)
0.01	6.7	454
0.05	6.7	515
0.10	6.7	515

Figure 29 shows the multi-avoidance path of the UAV for a north intercept of Object 2, with a starting position East of the UAV, and an East offset intercept of Object 1, initially starting East as well, such that no intercept is to occur if no maneuver is made. The path in avoidance of Obstacle 1 deviates opposite to that seen in Figure 27, resulting

in an avoidance maneuver left of Obstacle 1. This is due to the obstacle being offset east, leading the avoidance law to travel the safest path, being in front of the obstacle.

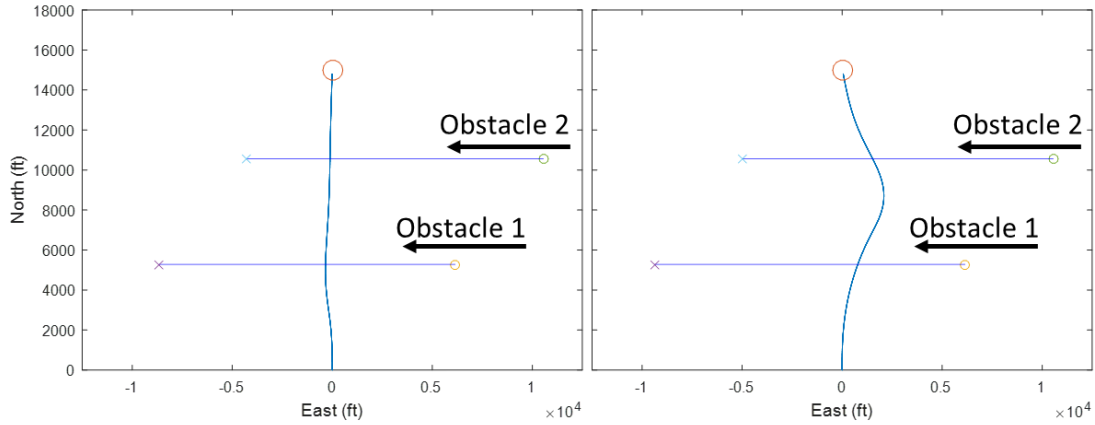


Figure 29: UAV Individual-Avoidance Paths (ft) - Obstacle 1 Offset Intercept

Table 10 shows an 830 *ft* miss distance of Obstacle 1 and the 63 *ft* miss distance of Obstacle 2, correlating to the simulation on the left in Figure 29, where the avoidance controller is active for Obstacle 1. The avoidance distances between Obstacle 2 and the UAV is large enough to be interpreted as a safe engagement. On the right of Figure 29, the avoidance of Obstacle 2 leads to an avoidance of Obstacle 1 that is less than Obstacle 2, the distances being 1.8 *ft* and 1,661 *ft*, respectively. This avoidance distance between Obstacle 1 and the UAV can clearly be defined as a collision due to the inherited aspect of the avoidance controller actively avoiding Obstacle 2 only.

Table 10: UAV Individual Avoidance Data - Obstacle 1 Offset Intercept

	Obstacle 1 Miss Distance (ft)	Obstacle 2 Miss Distance (ft)
Obstacle 1 Avoidance	830	63
Obstacle 2 Avoidance	1.8	1,661

The scenarios in Figure 29 are again presented in simulations for which the multi-avoidance controller was enabled, seen in Figure 30. These simulations are represented by the UAV paths for a chosen set of $\dot{\chi}$ threshold values, including 0.01, 0.05, and $0.1 \frac{deg}{s}$. The resulting path resembles that seen in the right-hand side of Figure 29, where Obstacle 2 was the only obstacle actively avoided. The same trend previously seen in multi-avoidance simulations is present in this case as well, with an increase in path deviation due to an increased $\dot{\chi}$ threshold.

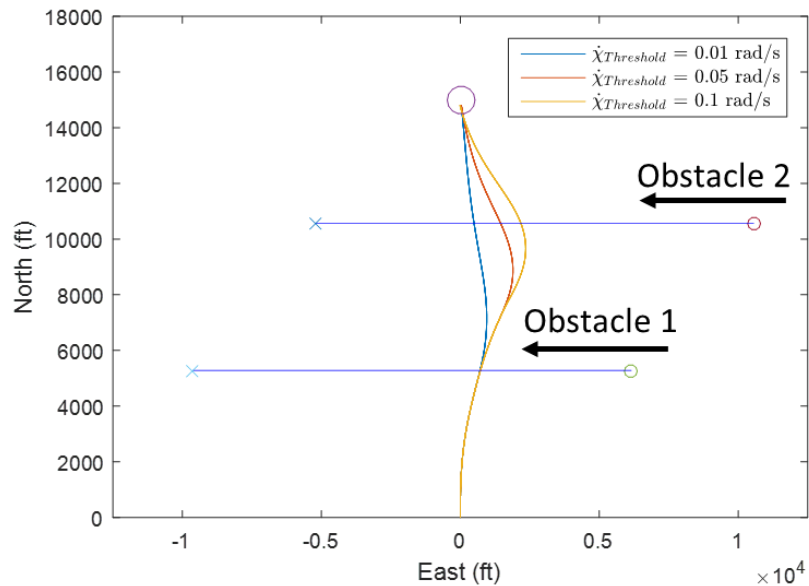


Figure 30: UAV Multi-Avoidance Paths (ft) - Obstacle 1 Offset Intercept

The miss distances shown in Table 11 relate to the avoidance paths shown in Figure 30. An increase in $\dot{\chi}$ threshold, like previous simulations, increases the avoidance distance, but only with regards to Obstacle 2, with a maximum closest distance of 1,481 ft between Obstacle 2 and the UAV for a $\dot{\chi}$ threshold of $0.05 \frac{deg}{s}$. However, the avoidance of Obstacle 1 is not only small, but decreases as the avoidance threshold

increases, due to the UAV choosing to avoid Obstacle 2 for a longer period of time, placing the UAV on a more imminent collision path with Obstacle 1.

Table 11: UAV Multi-Avoidance Data - Obstacle 2 Offset Intercept

$\dot{\chi}$ Threshold (rad/s)	Obstacle 1 Miss Distance (ft)	Obstacle 2 Miss Distance (ft)
0.01	59	472
0.05	48	1481
0.10	48	1901

Figure 31 presents an avoidance scenario with both obstacles having opposite trajectories. Obstacle 1 is positioned east of the UAV with a westward heading and velocity to ensure an intercept along the North axis. Obstacle 2 is set to start its trajectory west of the UAV with an Eastward heading, intercepting the UAV along the north axis as well. This scenario provides validation of the original avoidance laws applied to a multi-avoidance case with a most imminent threat strategy. In this case, the UAV will select an avoidance path behind the obstacles. As Figure 31 shows, the UAV guidance law does take the path least threatening by going behind both obstacles, even though it requires the UAV to divert in the opposite direction as the previous scenarios in Figure 28 and Figure 30.

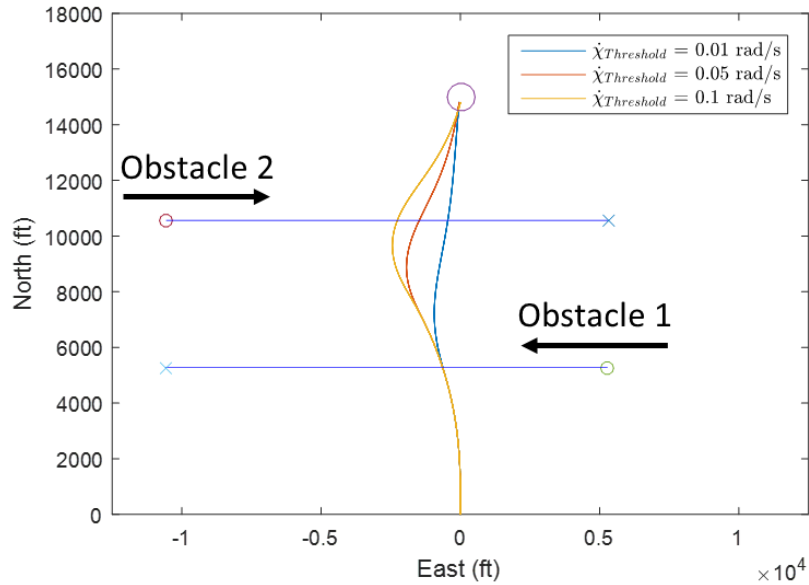


Figure 31: UAV Multi-Avoidance Paths (ft) - Obstacle 1 and 2 North Intercept

The avoidance distances in Table 12 provide insight on the avoidance law’s ability to ensure a safe distance is maintained for obstacles with opposite trajectories. The successful avoidance can once again be seen even as the smallest miss distances are 450 ft and 444 ft for a LOS rate threshold of $0.01 \frac{deg}{s}$.

Table 12: UAV Multi-Avoidance Data - Obstacle 1 and 2 North Intercept

Chi Dot Avoid (rad/s)	Obstacle 1 Miss Distance (ft)	Obstacle 2 Miss Distance (ft)
0.01	450	444
0.05	465	1541
0.10	465	1991

The final scenario in Figure 32 uses similar obstacle trajectories, with Obstacle 1 starting from the East and Obstacle 2 starting from the West of the UAV. However, Obstacle 2 has an initial North starting point closer to the UAV, making it so an intercept will not occur along the North axis. The resulting path shows a deviation of the UAV for

the avoidance of Obstacle 1, then a short maneuver behind Obstacle 2, placing the UAV trajectory close to its original North axis position. This trajectory is not to be confused with the avoidance controller directing the UAV back to the North axis. This is a coincidence of the scenario and corresponds solely to a visual based avoidance maneuver.

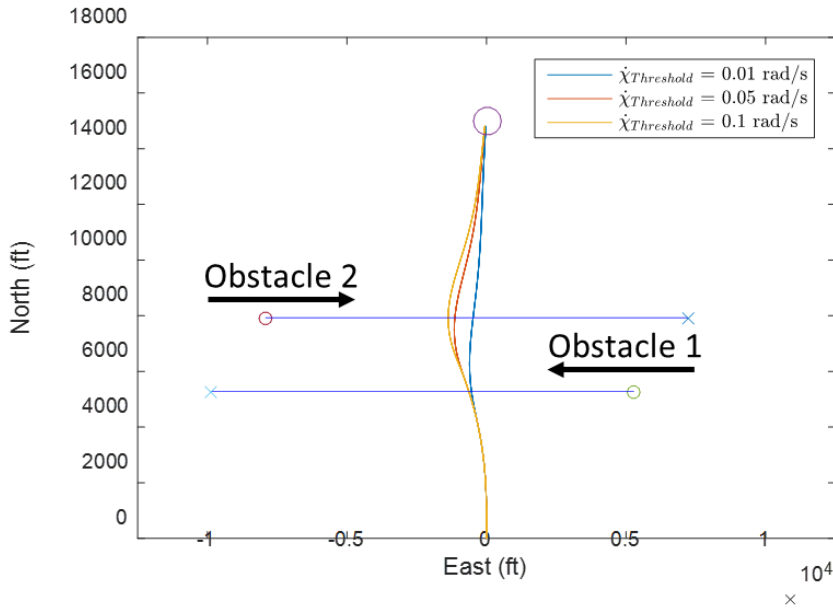


Figure 32: UAV Multi-Avoidance Paths (ft) - Obstacle 1 and 2 North Intercept - Close Paths

The distances in Table 13 correlate with the scenario in Figure 31, but also match the miss distances of the previous scenario, presented in Table 12. The reason for this is that decreasing the North starting position of Obstacle 2 still places it within a range of the UAV such that its $\dot{\chi}$ is below the threshold for avoidance. Since the trajectory direction is the same, the distance avoided by the UAV turns out the same, due to the limitation of the threshold value. This can be said for all three simulated LOS rate threshold values.

Table 13: UAV Multi-Avoidance Data - Obstacle 1 and 2 North Intercept - Close Paths

$\dot{\chi}$ Threshold (rad/s)	Obstacle 1 Miss Distance (ft)	Obstacle 2 Miss Distance (ft)
0.01	370	399
0.05	461	909
0.10	461	1037

The previous scenarios were simulated to provide validation of a functional, minimum $\dot{\chi}$ threshold avoidance controller for multi-obstacle avoidance. The different scenarios also helped to investigate the characteristics and limitations of the controller, discovering if adequate avoidance maneuvers are made without overcompensation. The results have shown that the validated single avoidance controller was successfully converted and implemented for a multi-avoidance situation using most imminent threat strategy. The variation in avoidance distance was significant between subsequent $\dot{\chi}$ thresholds, at some points increasing the distance by about 1,200 *ft*. While previous results for single avoidance cases show that velocity and initial distance plays a role in avoidance distance, a $\dot{\chi}$ threshold of $0.1 \frac{deg}{s}$ leads to an avoidance maneuver that can be considered to result in overcompensation for all multi-avoidance scenarios, regardless of initial velocity or position.

5.2. Static Avoidance

The successful initial implementation of the most imminent threat avoidance law using dynamic objects provided motivation to develop a scenario in which the UAV would attempt to avoid multiple static objects as well. Static objects represent a different challenge in avoidance with regards to the need to consider a higher quantity of points. Dynamic objects tend to be small and moving quickly with the possibility to have a

difficult color or contrast to identify. This means the number of points representing the object could be challenging to predict or small enough to isolate into one single centroid point. Large, static objects, such as buildings, trees, or boulders, would be associated with many points, introducing potentially a large number of objects to be avoided. For this reason, multi-object avoidance has been extended to a scenario in which the previously developed avoidance law is tested to navigate the UAV to a waypoint.

Figure 33 shows the scenario in which the effectiveness of the most imminent threat avoidance strategy is investigated on static objects. This environment consists of a single wall of points directly ahead of the UAV starting position. Keeping a simplistic environment for static point mass avoidance was important due to the lack of depth buffer, meaning at all times during the simulation, any obstacles behind other sets are still taken into consideration. This is not entirely representative of a real-world scenario, for which any buildings, rocks, or other objects behind one another are not taken into account due to their lack of visibility to the UAV. The first application of this simulation using the most imminent threat avoidance law can be seen in Figure 33.

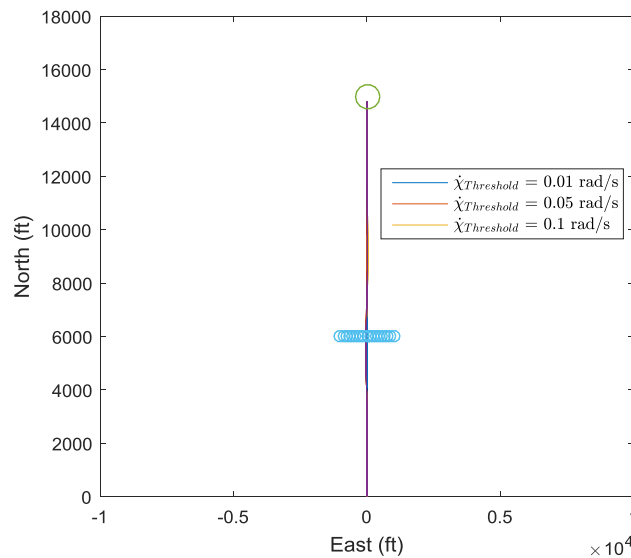


Figure 33: Most Imminent Threat Avoidance - Static Point Mass Wall

In Figure 33, the most imminent threat avoidance method is simulated on the static scene described above. In this scenario, and others in which the wall is enlarged or the waypoint is shifted laterally, the most imminent threat avoidance method fails to determine a successful path. The reason for this is that when an avoidance maneuver is initiated using the most imminent threat, the next most imminent threat becomes the primary threat. For the case of a wall of obstacles, these two threats are directly next to each other, causing the avoidance law to switch back and forth between the two obstacles with the smallest LOS rate, producing no avoidance maneuver in relation to the other obstacles in view.

6. PRO-NAV AVOIDANCE LAWS – OBJECTIVE WEIGHTING FUNCTION APPROACH

While the minimum LOS rate avoidance method with a most imminent threat strategy is effective for cases with a few, dynamic targets, its ability to effectively avoid numerous static and dynamic targets is limited. This shortcoming motivated the development of more robust guidance laws that implement a cost function to account for the UAV reaching a goal waypoint while determining effective avoidance maneuvers that incorporate all obstacles in view, not just the single most imminent collision threat. The progression for this development began with a broad approach, developing multiple cost functions based on three different Pro-Nav variables, $\dot{\chi}$, $\dot{\psi}$, and $\Delta\psi$. Recalling the earlier discussion of Pro-Nav guidance laws, these three variables are related as shown in Equation (11):

$$\dot{\psi} = N_H \dot{\chi} \rightarrow \Delta\psi = N_H \dot{\chi} \Delta t \quad (11)$$

To investigate the use of these variables in a guidance law, three types of cost functions were developed and implemented. These three methods include a negation, inversion, and threshold differencing terms in the cost function. Equations (12) - (13) show the implementation of the LOS rate $\dot{\chi}$ into the chosen cost functions.

$$J(\dot{\chi}_{com}) = W_{WP}(\dot{\chi}_{com} - \dot{\chi}_{WP})^2 + \sum_{i=1}^N W_i(\dot{\chi}_{com} - \dot{\chi}_{Neg_i})^2 \quad (12)$$

$$J(\dot{\chi}_{com}) = W_{WP}(\dot{\chi}_{com} - \dot{\chi}_{WP})^2 + \sum_{i=1}^N \frac{W_i}{(\dot{\chi}_{com} - \dot{\chi}_i)^2} \quad (13)$$

$$J(\dot{\chi}_{com}) = W_{WP}(\dot{\chi}_{com} - \dot{\chi}_{WP})^2 + \sum_{i=1}^N W_i(\dot{\chi}_{com} - \dot{\chi}_{Diff_i})^2 \quad (14)$$

As previously shown, $\dot{\chi}$ provides an overall objective in target vision based Pro-Nav intercept and avoidance, by driving the horizontal LOS rate $\dot{\chi}$ to or away from zero, respectively. Its application in Equations (12) - (14) is sought to have the same effect using different mathematical approaches. These cost functions, along with the others to follow, take the form of two components: the objective of the first term [$W_{WP}(\dot{\chi}_{com} - \dot{\chi}_{WP})^2$] is to intercept the waypoint, and the second term incorporates avoiding any obstacles that pose a threat, present in Equation (12) as $W_i(\dot{\chi}_{com} - \dot{\chi}_{Neg_i})^2$. This cost function approach differs from the most imminent threat method of avoidance in that there exists an intercept and avoidance component, as well as considering multiple obstacles rather than just one. The cost aspect of these terms are guided by the weights W_{WP} and W_i , which control how much the waypoint intercept or obstacle avoidance influences the solution. When $W_{WP} = 1$ and $W_i = 0$, the solution of this cost function simplifies to the LOS rate required for waypoint intercept. This trivial solution also occurs when no obstacles are in view, or no obstacle LOS rates are below the specified $\dot{\chi}$ threshold.

Equation (12) is the first implementation of a cost function that is minimized for a commanded LOS rate, $\dot{\chi}_{com}$. The computed value for the waypoint horizontal LOS rate, $\dot{\chi}_{WP}$, is the Pro-Nav LOS rate required to intercept the desired waypoint, while the horizontal LOS rate required to intercept all obstacles below a specified threshold is negated, such that $\dot{\chi}_{Neg_i} = -\dot{\chi}_i$. This logic can be considered a basic, but practical implementation of Pro-Nav avoidance. By negating the obstacle $\dot{\chi}$, the cost function is minimized for a solution that directs the UAV in the opposite direction of the Pro-Nav intercept. The solution to the cost function, $\dot{\chi}_{com}$, is then used in to command a change in

heading, $\Delta\psi$, as shown in Equation (11).

The cost function in Equation (13) uses LOS rate as a driving parameter as well, but in this form, it provides an inversion of the obstacle intercept $\dot{\chi}$ instead of a negation. This implementation is anticipated to have a similar effect as the negative $\dot{\chi}$, but in this case, each obstacle intercept $\dot{\chi}$ is treated as a singularity to be avoided. The resulting $\dot{\chi}_{com}$ is then used as it was in Equation (12) to produce a resultant change in heading.

Equation (14) is the last form of the cost functions, which is similar to the one used in Equation (12). Instead of negating $\dot{\chi}$, however, the difference is taken between the obstacle intercept $\dot{\chi}$ and a maximum $\dot{\chi}$ value, resulting in $\dot{\chi}_{Diff_i}$, shown below in Equation (15).

$$\dot{\chi}_{Diff_i} = -sgn\{\dot{\chi}_i\}(\dot{\chi}_{max} - |\dot{\chi}_i|) \quad (15)$$

$$sgn\{\dot{\chi}_i\} = \begin{cases} 1, & \dot{\chi}_i \geq 0 \\ -1, & \dot{\chi}_i < 0 \end{cases} \quad (16)$$

In Equation (15), $\dot{\chi}_{max}$ is the threshold value for avoidance criteria, such that only obstacles with a LOS rate less than this magnitude are considered for determining a solution to the cost function minimization. Equation (14) provides a form of the cost function that considers the magnitude and direction of $\dot{\chi}$ required to drive each obstacle LOS rate above the threshold.

6.1. Dynamic Avoidance

Figure 34 shows three avoidance paths, each representative of the three cost functions shown in Equations (12) - (14). The simulation scenario for these avoidance comparisons is similar to that performed for the multiple obstacle avoidance for the most imminent threat method; however, the medium-scale UAV model has been replaced with the small-

scale UAV model defined in Chapter 2. This change comes as a progression towards implementation into an urban environment, a scenario for which the medium-scale UAV would be too large and fast for practical purposes.

The scenario presented in Figure 34 sets up a waypoint objective in front of the UAV along the north axis, with two obstacles on an intercepting trajectory with UAV along the north axis. These obstacles have a slower velocity than the UAV and have an initial north and east offset position such that their intercept with the UAV will occur along the north axis.

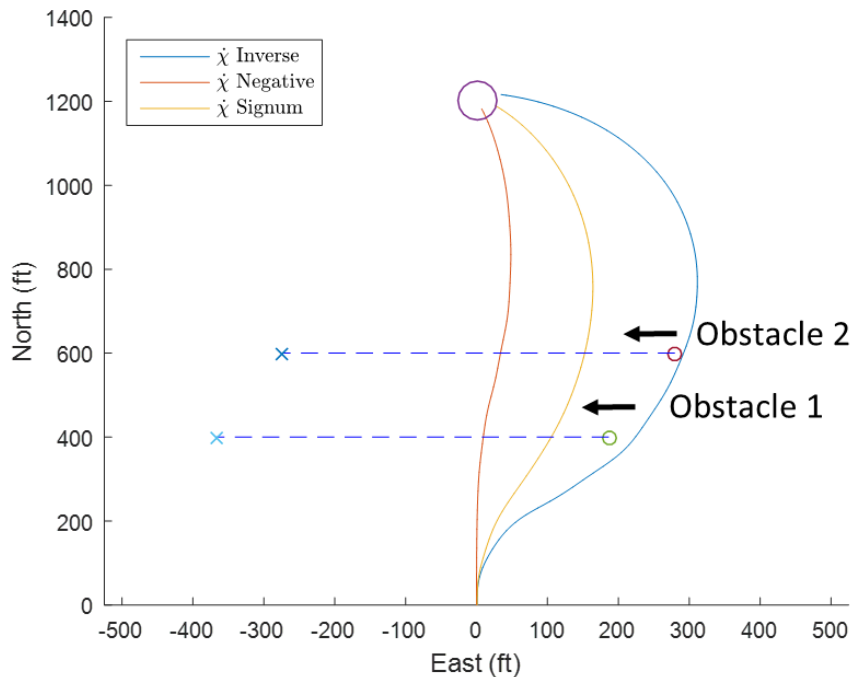


Figure 34: UAV avoidance path comparison using χ cost function forms - Dynamic

Comparing the three methods, the cost function implementing χ inverse resulted in the most aggressive avoidance maneuver, the negative χ application had the least aggressive, and the χ differencing method provided a maneuver aggressiveness in between the other two. Table 14 provides support for this classification, with the

differencing method cost function providing ample miss distances between the obstacles and the UAV, and are in between the values provided by the other methods of avoidance.

Table 14: Corresponding miss distance between UAV and obstacles for $\dot{\chi}$ based cost functions

Cost Function	Obstacle 1 Miss Distance (ft)	Obstacle 2 Miss Distance (ft)
$\dot{\chi}$ Inverse	157.2	259.7
$\dot{\chi}$ Negative	17.2	38.5
$\dot{\chi}$ Difference	85.5	136.6

In Figure 35, the corresponding heading history of the $\dot{\chi}$ cost function avoidance laws can be seen for the three paths in Figure 34. These headings provide greater insight into the aggressiveness of the guidance laws, such that large and sporadic heading changes correlate to a more aggressive avoidance law. This trend can be seen for the UAV heading produced by the $\dot{\chi}$ inverse method, where large heading changes are produced at the start of the simulation, when intercepting trajectories are first accounted for, and then again after the avoidance of the first obstacle has occurred. The cost function guidance law using the negative $\dot{\chi}$ avoidance method provides a relatively small miss distance between the UAV and obstacles, as seen in Table 14. Investigation into the UAV heading commands for this avoidance law suggests that an avoidance maneuver slowly grew as the obstacle $\dot{\chi}$ increased, resulting in a delayed avoidance. The resulting UAV heading produced by the $\dot{\chi}$ differencing cost function can be described as a trending increase and decrease relative to the UAVs avoidance criteria.

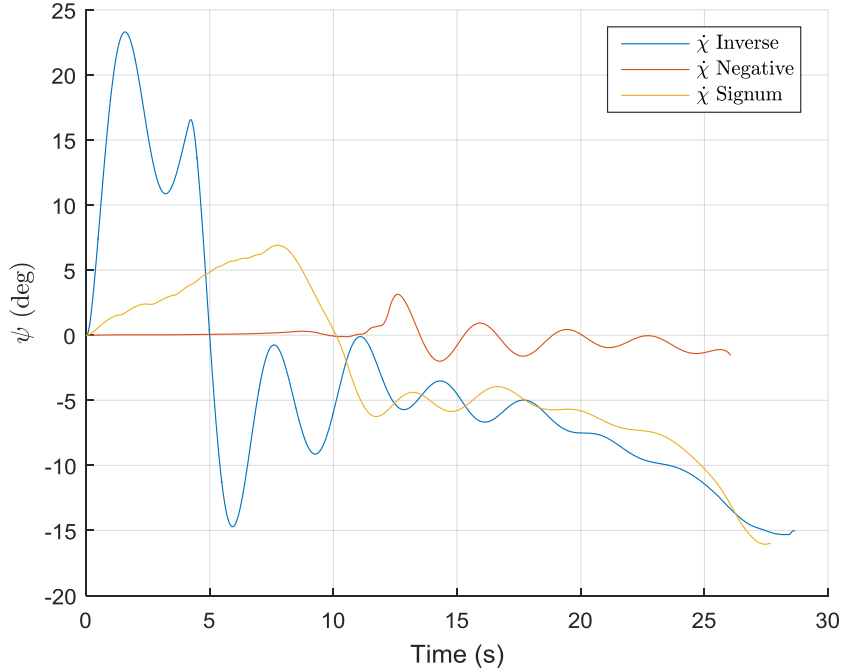


Figure 35: Comparison of actual UAV heading for corresponding χ cost function avoidance paths – Dynamic Case

Equations (17) through (19) provide a second set of cost functions that utilize the ProNav variable, $\dot{\psi}$, rather than the use of $\dot{\chi}$, in the same forms of the cost function shown in Equations (12) - (14). The implementation of these cost functions varies in that once a minimized solution is found for $\dot{\psi}_{com}$, the difference in this value and the UAV current $\dot{\psi}$ is used to produce a commanded bank angle.

$$J(\dot{\psi}_{com}) = W_{WP}(\dot{\psi}_{com} - \dot{\psi}_{WP})^2 + \sum_{i=1}^N W_i(\dot{\psi}_{com} - \dot{\psi}_{Neg_i})^2 \quad (17)$$

$$J(\dot{\psi}_{com}) = W_{WP}(\dot{\psi}_{com} - \dot{\psi}_{WP})^2 + \sum_{i=1}^N \frac{W_i}{(\dot{\psi}_{com} - \dot{\psi}_i)^2} \quad (18)$$

$$J(\dot{\psi}_{com}) = W_{WP}(\Delta\dot{\psi}_{com} - \Delta\dot{\psi}_{WP})^2 + \sum_{i=1}^N W_{O_i}(\dot{\psi}_{com} - \dot{\psi}_{Diff_i})^2 \quad (19)$$

The difference between Equations (17) - (19) and Equations (12) - (14) can be noted using Figure 36. The UAV avoidance paths have similar trends, specifically with respect

to the aggressiveness of each avoidance function used. However, using $\dot{\psi}$ as the primary function minimization variable leads to more significant oscillations in the UAV avoidance path, and in turn the heading.

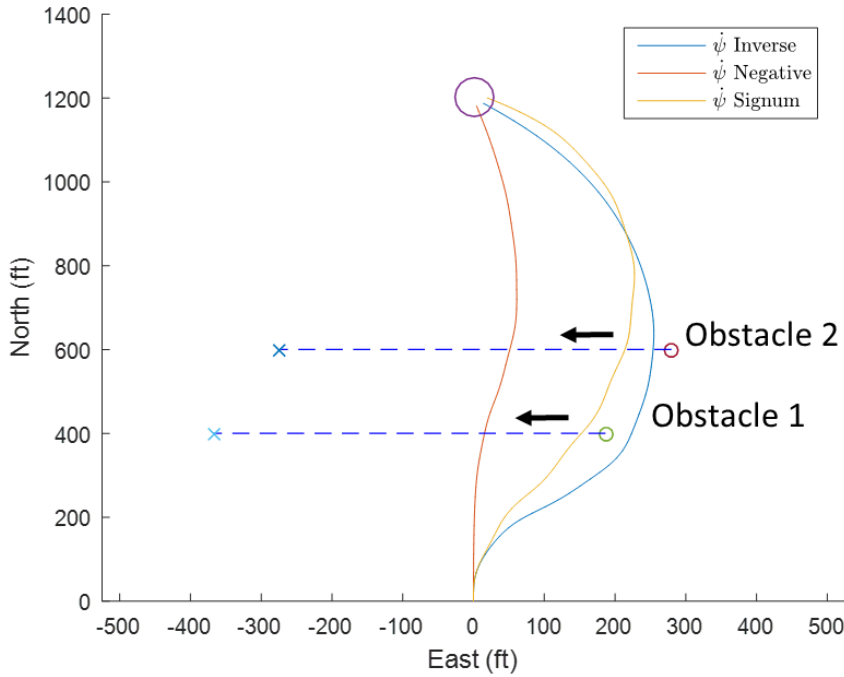


Figure 36: UAV avoidance path comparison using $\dot{\psi}$ cost function forms – Dynamic

Along with Table 15, it is also clear that the cost functions in Equations (17) through (19) cost functions trend towards larger miss distances between the UAV and obstacles. While this is favorable, it comes at a cost of an oscillating characteristic, which is clearly seen in Figure 37.

Table 15: Corresponding miss distance between UAV and obstacles for $\dot{\psi}$ based cost functions

Cost Function	Obstacle 1 Miss Distance (ft)	Obstacle 2 Miss Distance (ft)
$\dot{\psi}$ Inverse	194.1	309.1
$\dot{\psi}$ Negative	27.2	48.5
$\dot{\psi}$ Difference	108.8	175.4

The magnitudes of the UAV heading history in Figure 37 are smaller than those shown in Figure 35, where $\dot{\chi}$ is used as the primary Pro-Nav variable in the cost function equations; however, the presence and frequency of the heading oscillations is highly undesirable.

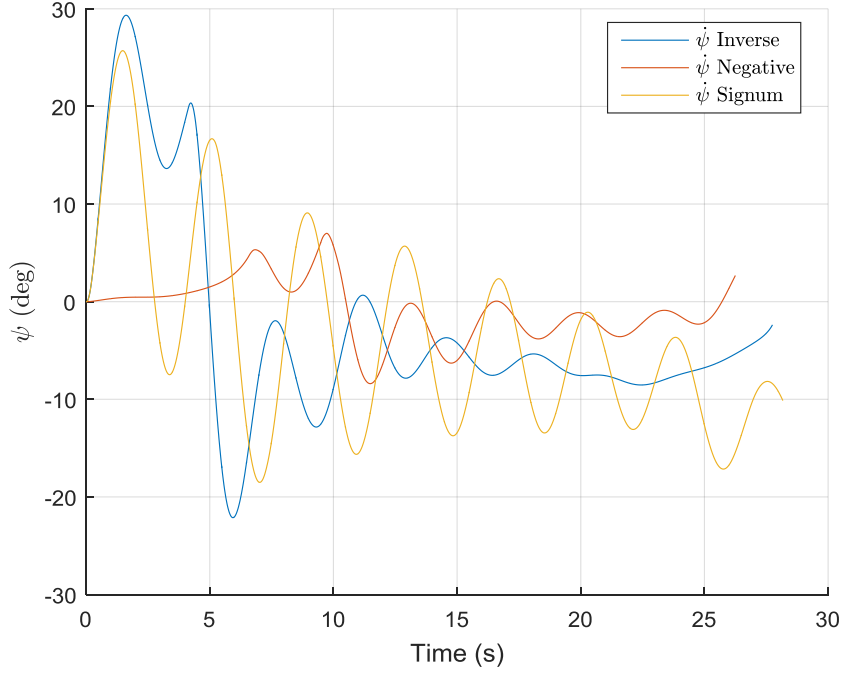


Figure 37: Comparison of UAV actual heading for corresponding $\dot{\psi}$ cost function avoidance paths – Dynamic Case

Equations (20) - (22) represent a third set of cost functions that utilize the Pro-Nav variable $\Delta\psi$, rather than the use of $\dot{\chi}$ of $\dot{\psi}$. In this cost function set, $\Delta\psi_{Neg_i}$, $\Delta\psi_i$, and $\Delta\psi_{Diff_i}$ are the individual obstacle Pro-Nav headings referenced previously in Equation (11), such that the Pro-Nav constant, N_H , and time step, Δt , are used.

$$J(\Delta\psi_{com}) = W_{WP}(\Delta\psi_{com} - \Delta\psi_{WP})^2 + \sum_{i=1}^N W_i(\Delta\psi_{com} - \Delta\psi_{Neg_i})^2 \quad (20)$$

$$J(\Delta\psi_{com}) = W_{WP}(\Delta\psi_{com} - \Delta\psi_{WP})^2 + \sum_{i=1}^N \frac{W_i}{(\Delta\psi_{com} - \Delta\psi_i)^2} \quad (21)$$

$$J(\Delta\psi_{com}) = W_{WP}(\Delta\psi_{com} - \Delta\psi_{WP})^2 + \sum_{i=1}^N W_i(\Delta\psi_{com} - \Delta\psi_{Diff_i})^2 \quad (22)$$

Implementing $\Delta\psi$ into the cost functions has the potential to provide a more direct approach by determining a weighted change in heading based on all the obstacles that pose a threat, rather than producing a commanded heading using a weighted $\dot{\chi}$ value. The ensuing avoidance paths for these cost functions can be seen in Figure 38, which show that a similar trend in aggressiveness once again applies to each subsequent cost function method using $\Delta\psi$.

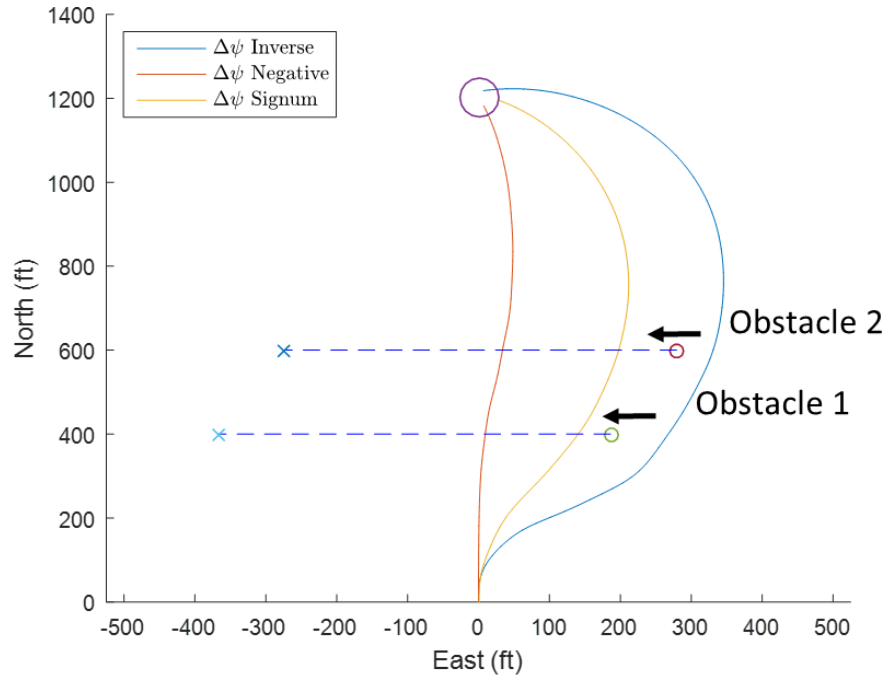


Figure 38: UAV avoidance path comparison using $\Delta\psi$ cost function forms – Dynamic Case

The miss distances between the UAV and obstacles using the cost function set for $\Delta\psi$ are like those produced using the $\dot{\chi}$ implementation, as shown by the avoidance distances in Table 16. The paths do differ however, most noticeably for the Inverse $\Delta\psi$ cost function, which exhibits a larger waypoint intercepting radius than previously seen.

Table 16: Corresponding miss distance between UAV and obstacles for $\Delta\psi$ based cost functions

Cost Function	Obstacle 1 Miss Distance (ft)	Obstacle 2 Miss Distance (ft)
$\Delta\psi$ Inverse	172.9	257.8
$\Delta\psi$ Negative	22.2	53.3
$\Delta\psi$ Difference	127.0	196.3

The UAV heading histories for the $\Delta\psi$ cost functions in Figure 39 show similar trends as the previous heading histories in Figure 35 and Figure 37 with regards to the consistency of the heading change using the differencing function and the large change in heading using the inverse cost function. Oscillations do still exist throughout the entire flight; however, they are much less aggressive than the oscillations present in with the implementation of the ψ cost functions.

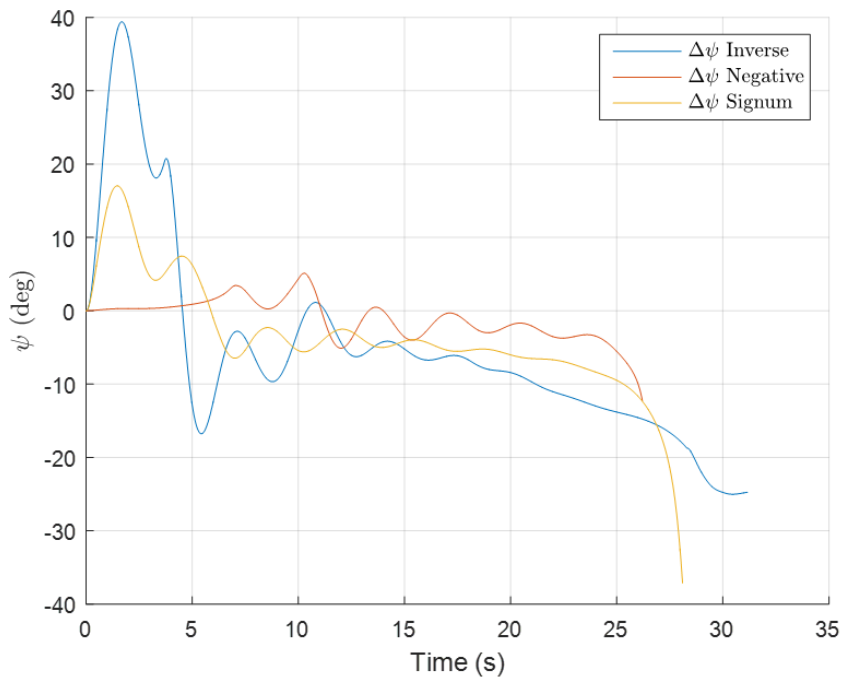


Figure 39: Comparison of UAV heading for corresponding $\Delta\psi$ cost function avoidance paths – Dynamic Case

6.2. Static Avoidance

The primary motivation for developing a cost function based avoidance law was to overcome the complication of avoiding multiple obstacles successfully. The effort in previous sections shows that considering only the most imminent static threat at any time during flight can result in a back-and-forth maneuvering effort when considering multiple obstacles, eventually resulting in a collision. In this section, avoidance results using the cost functions provided in the dynamic avoidance section are presented for static avoidance cases. This static scenario places a wall of point-masses in front of the UAV, which are sought to be avoided. Simulations are once again generated for the three cost function methods using the three primary Pro-Nav variables.

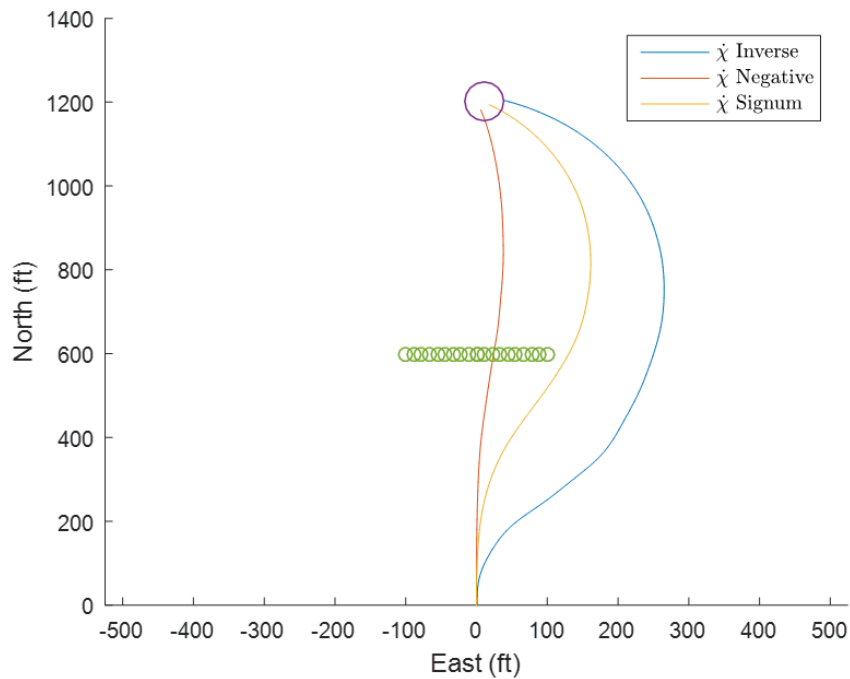


Figure 40: UAV avoidance path comparison using χ cost function forms – Static Case

Figure 40 shows a comparison of avoidance paths taken using the χ based avoidance cost functions in the static obstacle environment. The same trend as noted with the

dynamic obstacle simulations again presents itself, with the inverse cost function providing the largest avoidance maneuver, the negative Pro-Nav function providing the smallest avoidance maneuver, and the differencing avoidance path having an in-between level of aggressiveness.

Avoidance distances between the UAV and obstacles for these simulations are not included. The reason for this is that in these static scenarios, the point-masses representing the wall are placed close enough together such that any maneuver that places the UAV path directly through the wall can be considered a collision. This property, along with static obstacles being easier to visualize than dynamic obstacles in two dimensions, makes it so collisions can be easily inferred.

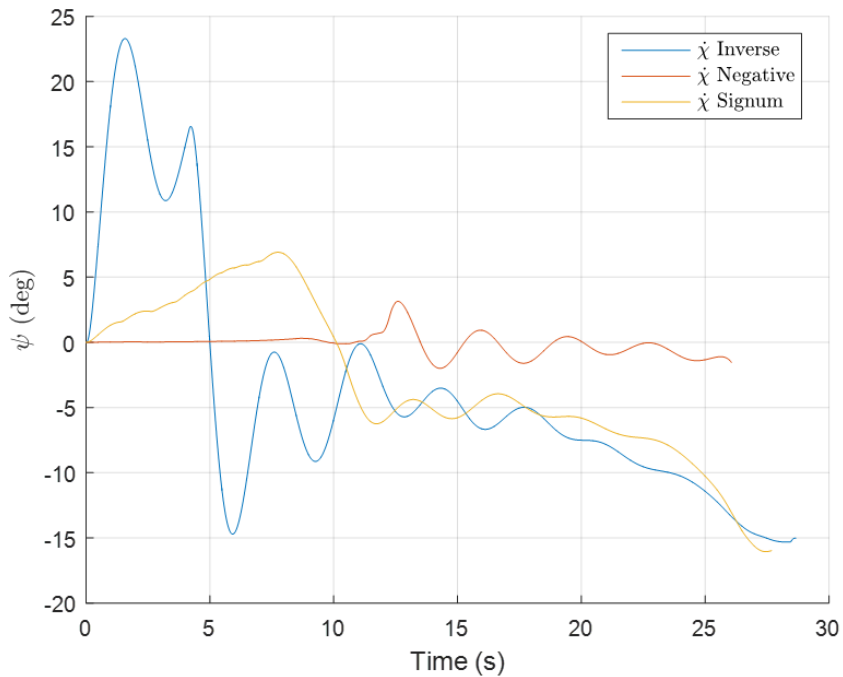


Figure 41: Comparison of UAV actual heading for corresponding $\dot{\chi}$ cost function avoidance paths – Static Case

To aid in the understanding of the maneuvers produced, the UAV heading history is once again provided in Figure 41 for the avoidance law using the three $\dot{\chi}$ cost functions.

The inverse cost function again produces aggressive heading commands, resulting in large oscillations about the UAV yaw axis. The negative Pro-Nav cost function produces very little heading change throughout the flight, corresponding to the straight, non-avoiding flight path seen in Figure 40.

Figure 42 shows a comparison of avoidance paths taken using the $\dot{\psi}$ based avoidance cost functions. A larger avoidance maneuver can be seen in the path generated using the negative cost function method and a decreased aggressiveness is evident in the avoidance path generated using the inverse cost function. The most notable aspect in Figure 42 is the differencing cost function path, which makes an avoidance maneuver left of the point-mass wall. This is due to the representative signum function used for the differencing cost function, for which if a $\dot{\chi}$ of zero occurs, an avoidance maneuver to the left is commanded.

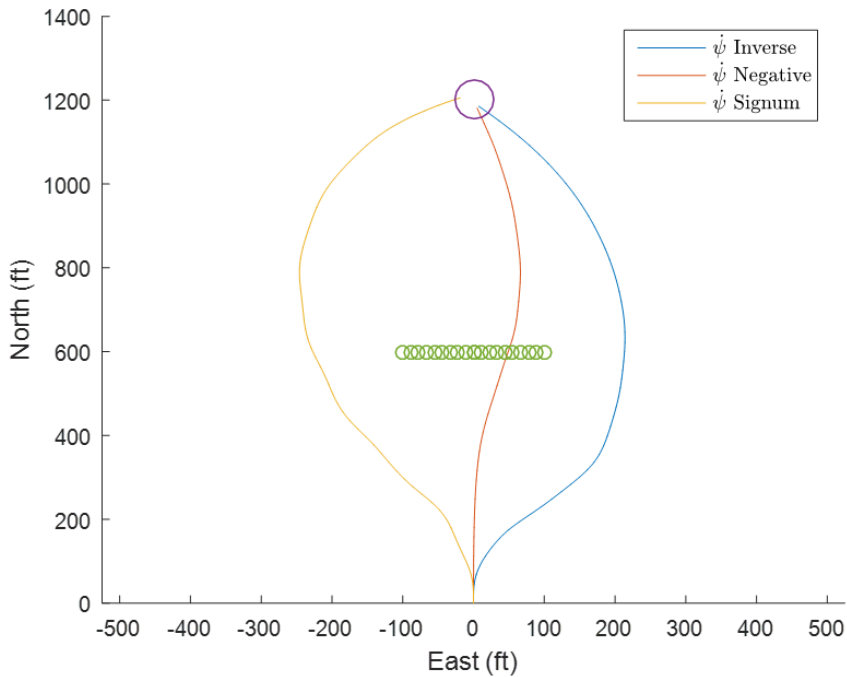


Figure 42: UAV avoidance path comparison using $\dot{\psi}$ cost function forms – Static Case

The corresponding heading history for the static $\dot{\psi}$ cost function avoidance law is shown in Figure 43. Though the negative Pro-Nav cost function provides another unsuccessful avoidance maneuver, it exhibits the most well behaved maneuvers when attempting to avoid and returning to a waypoint intercept trajectory. The inverse Pro-Nav cost function once again generates an aggressive maneuver, but does not show the large oscillations in heading that were seen using the $\dot{\chi}$ based cost functions. The differencing cost function for this scenario exhibits large oscillations, which is clearly shown in the heading history. These large heading oscillations are similar to those obtained using the $\dot{\chi}$ based differencing equation. In this scenario, the effect of the oscillation is large enough to be noticed in the flight path visualized in Figure 42.

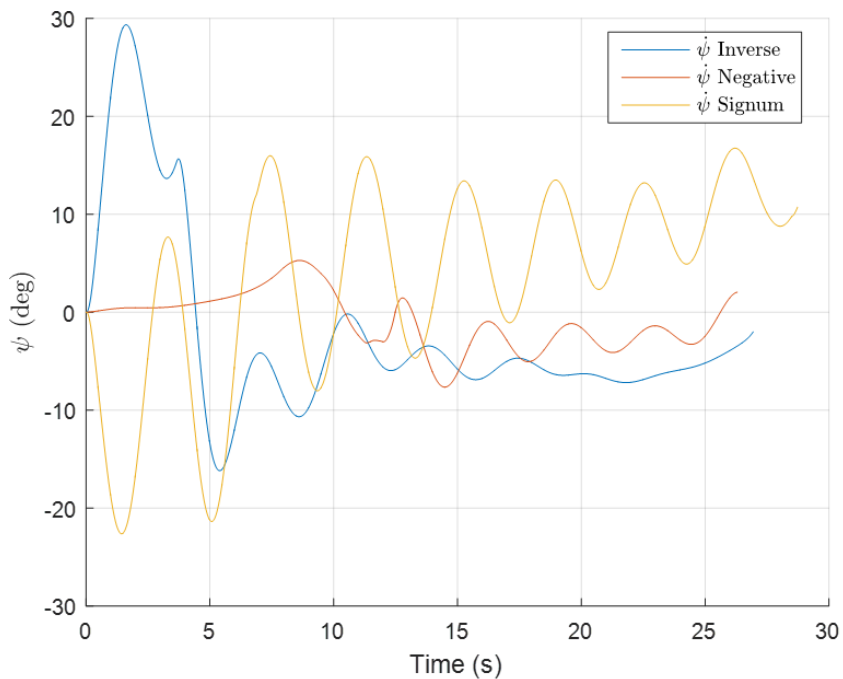


Figure 43: Comparison of UAV actual heading for corresponding $\dot{\psi}$ cost function avoidance paths – Static Case

The final static scenario in Figure 44 shows the avoidance paths generated using the cost function set with $\Delta\psi$. Again, the use of the negative Pro-Nav cost function for avoidance leads to a collision with the point-mass wall, while the other two avoidance

functions lead to successful avoidance maneuvers. The magnitude of the avoidance maneuver generated by the inverse Pro-Nav and differencing cost functions results in a greater deviation than the previous methods using χ and ψ , shown in Figure 40 and Figure 42.

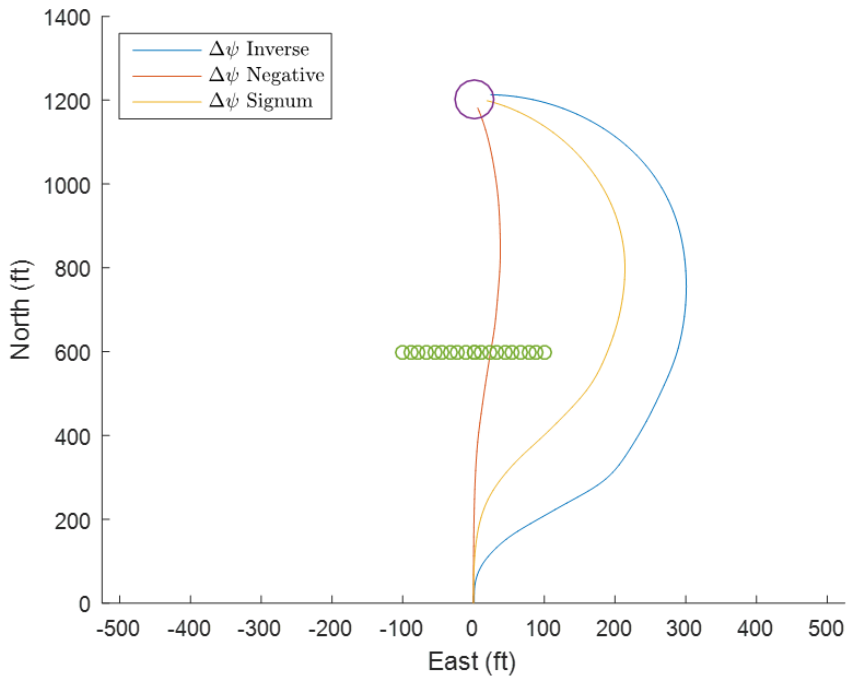


Figure 44: UAV avoidance path comparison using $\Delta\psi$ cost function forms – Static Case

The heading histories for the three cost functions using $\Delta\psi$ can be seen in Figure 45. Each history shows a similar trend as the previously provided histories in Figure 41 and Figure 43; however, there is an overall decrease in aggressiveness for all avoidance paths, noted by the decrease in oscillation magnitudes. The heading history shown for the differencing method is particularly important because of the relatively smooth transition of heading changes that correlate to the avoidance path. This path does not have significant oscillations or sharp turns, like those exhibited in the inverse Pro-Nav path.

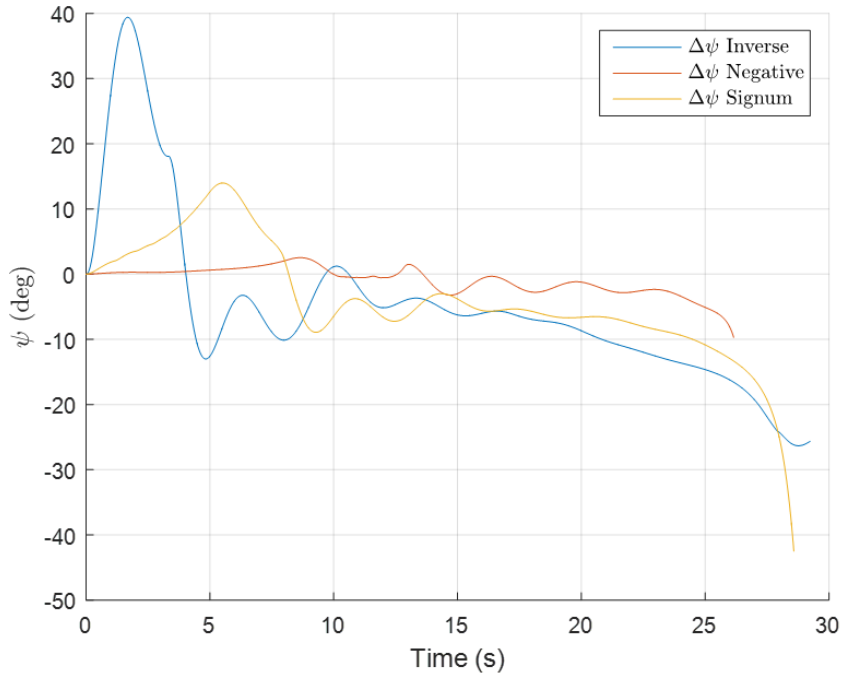


Figure 45: Comparison of UAV actual heading for corresponding $\Delta\psi$ cost function avoidance paths – Static Case

The results provided using the three sets of cost functions for three Pro-Nav variables provide information that can be used to determine an all-around successful avoidance law. Overall, all methods had positive and negative aspects with regards to capabilities and trends relating to the characteristics of using each Pro-Nav variable.

Implementation of the negative Pro-Nav cost function for all dynamic and static simulations provides the least aggressive path for avoidance. While this is a positive aspect for avoidance of dynamic obstacles, since effort is required to make a successful avoidance maneuver, it failed to provide enough avoidance in the case of static obstacles. Regardless of the Pro-Nav variable used, there is a consistency in the lack of avoidance provided by the negative Pro-Nav method, particularly when applied with $\dot{\chi}$ as the cost function objective.

The use of inverse Pro-Nav for each investigated variable provides an overall successful avoidance maneuver in each dynamic and static simulation. The maneuvers do

tend to exhibit large, aggressive changes in the avoidance headings and, in turn, the resulting UAV paths. This trend is particularly prevalent with the use of $\dot{\psi}$ as the cost function optimization variable. In Figure 37 and Figure 43, the magnitude and frequency of the heading oscillations imply a negative aspect of the guidance law, being that a constant overcompensation leads to an overreactive recovery towards a waypoint intercept path, which once again places the UAV on a collision path that must be avoided with another aggressive maneuver.

The differencing cost function, which uses the difference in a maximum LOS rate and the obstacle LOS rate for optimization, shows favorable traits in all dynamic and static scenarios with the use of $\Delta\psi$ as the optimization variable. The avoidance paths shown in Figure 39 - Figure 45 provide a practical miss distance between the dynamic and static obstacles, while the UAV heading maintained a consistent progression that did not show overly aggressive maneuvers or significant heading oscillations. Based on the simulation results, the use of $\Delta\psi$ as the optimization variable with the differencing cost function provides the most viable avoidance function for collision avoidance of multiple dynamic and static obstacles. Therefore, further investigation into this method of guidance is presented in the next section of this thesis.

7. PRO-NAV AVOIDANCE LAW – OBJECTIVE FUNCTION USING $\Delta\psi$ DIFFERENCING

Among the developed objective based cost functions, the LOS rate differencing method for use of optimizing avoidance based on a change in heading was chosen as the most suitable method for further application, development, and characterization.

Therefore, several new scenarios were constructed and introduced that incorporate the avoidance of multiple groups of static obstacles, and the integration of dynamic and static obstacles into one scenario.

The first of these new scenarios presents static obstacles, represented by point-masses, aligned in the shape of a large corner as an attempt to simulate an urban scenario which has a building present in between the UAV and waypoint. Generating a solution to this scenario would provide a better insight into how the guidance law will react when presented a set of feature points at a larger angle of trajectory, such as if the UAV were to approach a building directly into its corner.

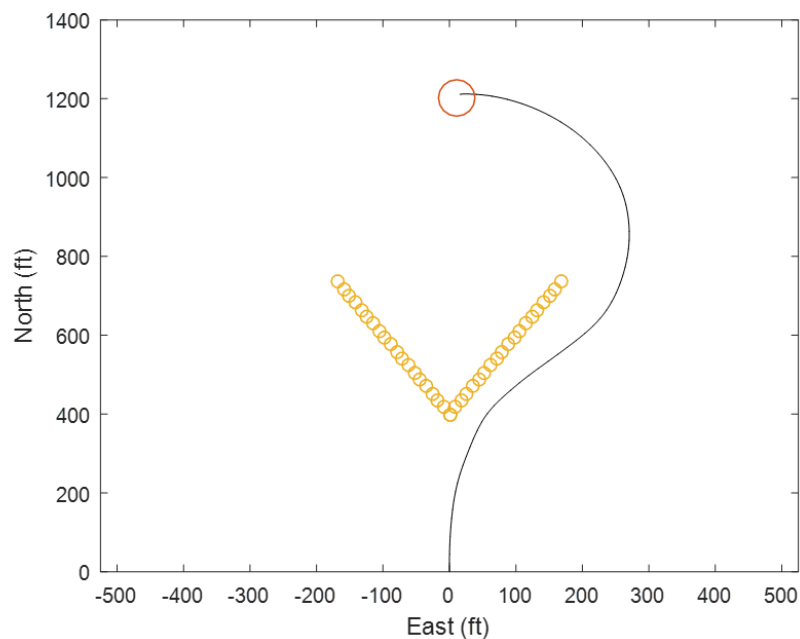


Figure 46: UAV avoidance path for static corner avoidance

Figure 46 shows the generated UAV avoidance path for the scenario presenting the mentioned collection of feature points in the shape of a corner, the vertex of which lies 400 ft north of the UAV starting position. The UAV successfully avoids the static obstacle set by deviating to the east and briefly traveling parallel to the obstacle. Finally, once clear of the obstacle, the UAV makes a continuous heading adjustment to intercept the waypoint. This heading change, along with the history describing the avoidance portion of the path, can be seen in Figure 47.

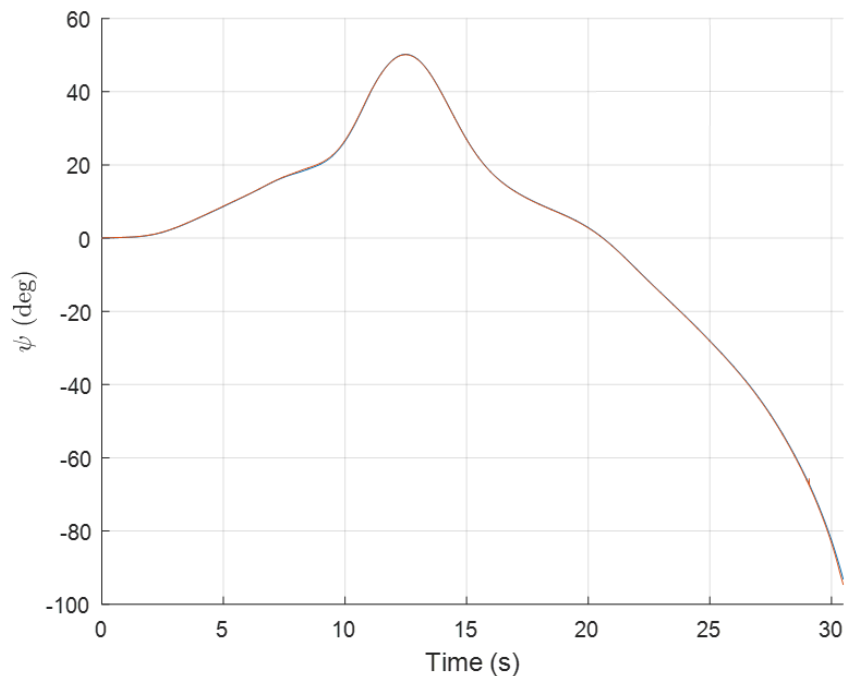


Figure 47: UAV heading history for static corner avoidance

Figure 47 shows the avoidance heading history for the UAV path in Figure 46. The increase in heading until 13 seconds correlates to the east avoidance maneuver generated by the guidance law to navigate the UAV around the obstacle set. The heading then decreases accordingly to keep the UAV on an intercepting path with the waypoint.

The next simulation in Figure 48 places two static point-mass walls in between the intercepting trajectories of two dynamic obstacles. This setup locates the walls near the

path taken by the UAV when only the dynamic obstacles were present. Again, it is easy to see the avoidance of static obstacles in this scenario, but difficult to interpret the avoidance of the dynamic obstacles. The distances between the first and second dynamic obstacle and the UAV are 16.7 ft and 45.9 ft . Though these distances seem small, this is a relatively successful miss distance for a small-scale UAV.

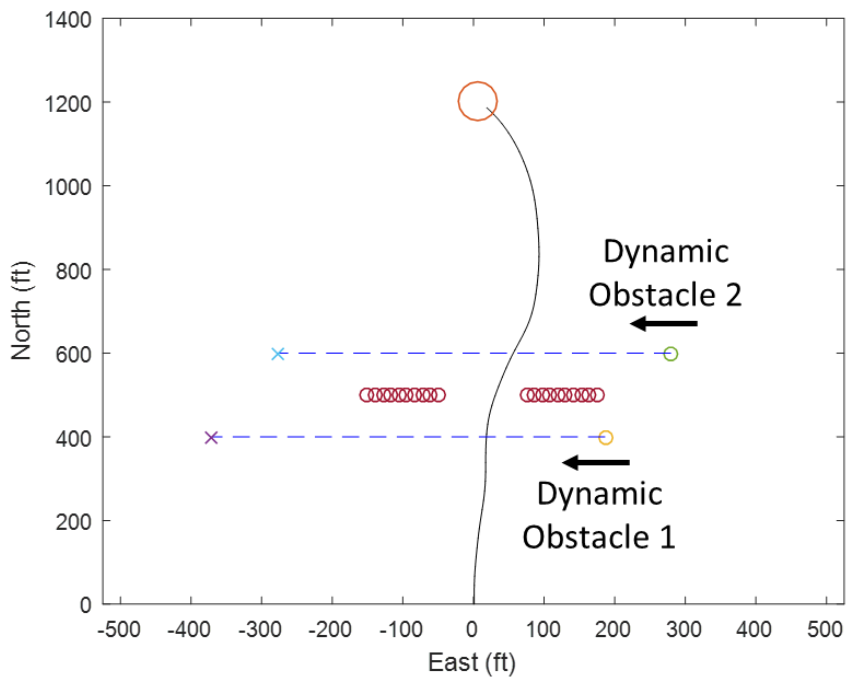


Figure 48: UAV Avoidance Path of dynamic and close static obstacles

The different aspects of the entire avoidance path are made more clear in the heading history provided in Figure 49. The initial reaction of the avoidance law results in a positive heading change, as the UAV maneuvers to avoid the dynamic obstacles. This maneuver moves the UAV off a collision path with the first dynamic obstacle, but also changes heading to ensure avoidance of the right static wall. Once the static wall is out of view and no longer a threat, the UAV makes a large heading change to ensure avoidance of the second dynamic obstacle.

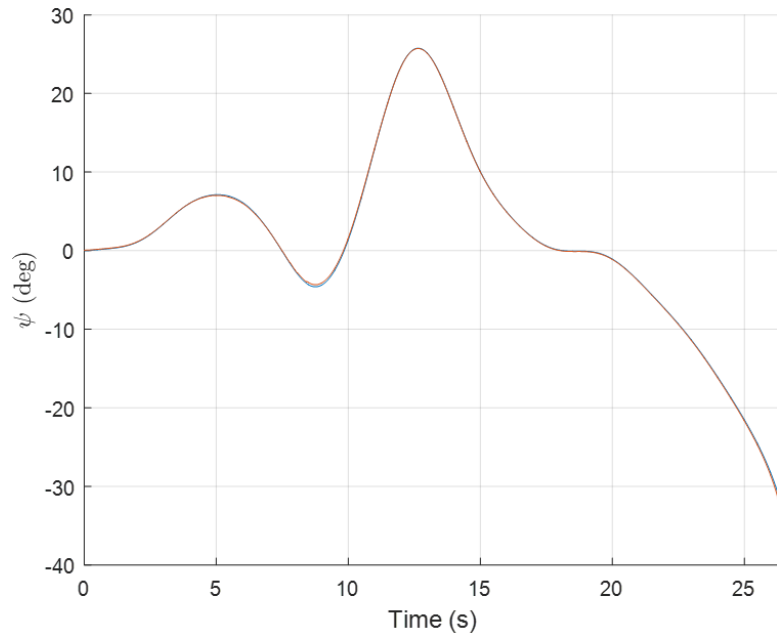


Figure 49: UAV actual heading history for avoidance of dynamic and close static obstacles

The next simulation in Figure 50 places two static point-mass walls farther aft of the intercepting trajectories of the dynamic obstacles. This setup locates the walls farther away for similar reasons as the previous simulation; however, placing them near the path taken by the UAV when only the dynamic obstacles were present at a farther range presents a more challenging scenario. The avoidance path places the UAV at a minimum distance of 50.7 ft and 166.0 ft for the first and second dynamic obstacle.

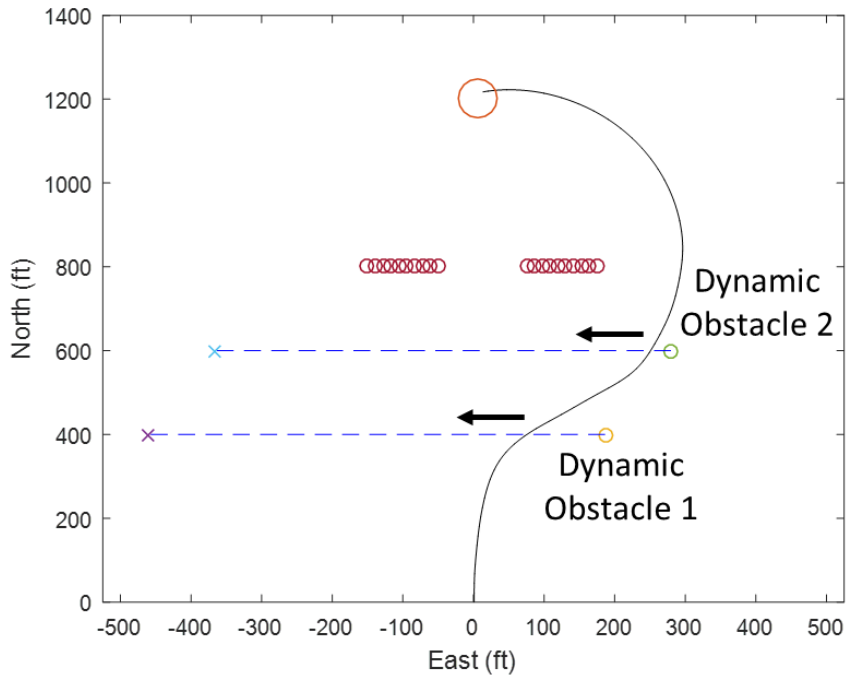


Figure 50: UAV Avoidance Path of dynamic and far static obstacles

The heading history for UAV avoidance of dynamic and far static obstacles is shown in Figure 51. The gradual and then steep increase of the heading change relates directly to the path shown in Figure 50, in which the avoidance maneuver increases drastically close to the first dynamic obstacle, then progresses back towards an intercepting path with the waypoint.



Figure 51: UAV heading history for avoidance of dynamic and far static obstacles

8. VIRTUAL REALITY SIMULATION

The point-mass obstacle simulations discussed in the previous sections of this thesis provide a thorough investigation into the understanding and development of a successful, vision-based Pro-Nav avoidance law, capable of multi-obstacle avoidance in dynamic and static scenarios. The characteristics determined in each progression of development have provided an overall “best” avoidance method that utilizes an objective-based cost function to optimize a change in heading, $\Delta\psi_{com}$, required for avoidance of all obstacles that pose a collision threat. These simulations were performed under “ideal” conditions, with exact obstacle point coordinates known, no camera information to be translated, and no noise introduced into the guidance law. The reason for this was so that the characteristics of the Pro-Nav guidance laws could be studied under best-case scenarios. This section extends the previous simulation results to provide an investigation of the chosen Pro-Nav avoidance law in a “real-world” environment using a virtual reality simulator.

8.1. MetaVR Virtual Reality Scene Generator

MetaVR Virtual Reality Scene Generator (VRSG) is a distributed interactive simulation (DIS) protocol based virtual environment that allows for the design and control of virtual environments and objects within the environment. MetaVR was used for this thesis to provide a virtual environment in which a “camera-in-the-loop” simulation could be conducted using a viewpoint generated in MetaVR. The process of this simulation begins with MATLAB Simulink providing UAV position and orientation information such that the viewpoint in MetaVR can be controlled. The 6 DoF nonlinear model in Simulink provides realistic control of the UAV based on the commands directed

by the Pro-Nav guidance law. Once updated, the viewpoint in MetaVR is transferred to Simulink, in the form of a synthetic camera image, which is then processed by a feature point tracking algorithm. Based on tracked feature points, the Pro-Nav guidance law generates an updated command and maneuver that is once again sent to MetaVR for position and orientation update. Figure 52 shows a high-level overview of the simulation process, which initially begins with the host simulator, MATLAB/Simulink, providing initial position and orientation information for MetaVR, the image generator, for a viewpoint to be generated.



Figure 52: process overview for Pro-Nav guidance virtual reality integration

Common Image Generator Interface (CIGI) is the communication interface that translates the data packages from MATLAB/Simulink into DIS protocol packets for MetaVR to interpret and apply. This simulation setup can provide both a point of view (POV) camera image, as well as Lidar information about the environment. The use of simulated Lidar has practical implications; however, its use is beyond the scope of this thesis.

The first of the virtual environment scenarios places the objective waypoint directly in front of the UAV, with a large building in between. The buildings and textures used in this simulation are premade and provided with MetaVR installation software. The scale of buildings may be increased or decreased accordingly; however, depending on the original size of the rendered building, the textures may decrease in quality with a large

increase in scaling. The most appropriately sized building has been chosen for this simulation such that the quality of the texture can still provide “good” quality feature point definitions.

8.2. Feature-Point Tracking

In the virtual reality simulations, the bearings to static obstacles are provided using a Kanade-Lucas-Tomasi (KLT) feature point tracking algorithm. Corners, or intersections of high gradient profiles in the image provide good feature points that have the potential and quality to be found and tracked throughout a dynamic scene. The Harris corner detection algorithm identifies a corner, defined when a small area, or window, of pixels in the image exhibits a large intensity change in all directions when the window is shifted from its original location. The KLT tracking algorithm attempts to find these corners in the next image generated and determines the direction and magnitude of the change in each corner’s location. This is then interpolated for a set number of frames, at which point the interpolation may provide improved accuracy. As tracked feature points leave the field of view, new corner detections are necessary to reidentify feature points to track for the next set of images. An example use of corner detection can be seen in Figure 53, where features on a sample MetaVR building are identified as corners and tracked throughout the simulation, although the tracking visualization is not visible through this still image.

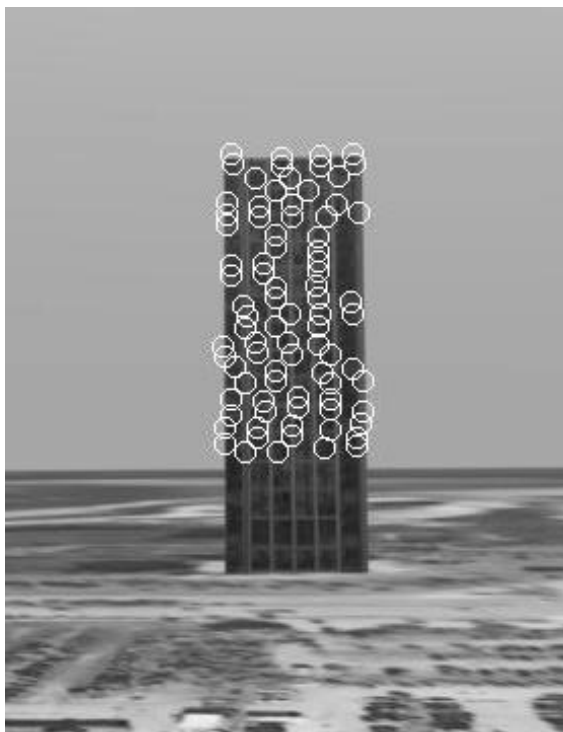


Figure 53: Example corner detection of building features in MetaVR

During this feature point tracking process, feature point will eventually be removed from the dynamic tracking list, due to their movement out of view of the camera or a decrease in quality based on camera angle or noise. This is compensated by re-establishing feature points every certain number of frames recorded. By doing so, previous points still in view and of sufficient quality are detected and tracked again while new points are added for use in the Pro-Nav avoidance law. It is important to note that KLT feature tracking is reliable only when changes to the feature location are small. This condition suits this application because the UAV is traveling at a relatively low velocity with a relatively high frame rate.

8.3. Single Obstacle Avoidance

The first scenario presented in this section places a single building in between the UAV and waypoint within the virtual environment, 1,500 *ft* from the UAV. It is important to note that the waypoint is still represented by a point mass, whose LOS angle and rate measurements are derived using the same method of the previous point mass simulations. Figure 54 shows the initial camera view captured from MetaVR, with a total vertical and horizontal view angles of 36° and 60°, respectively.



Figure 54: Camera view from UAV of single building MetaVR simulation

This simulation, and the ones to follow, utilize the chosen cost-function-based avoidance law that uses the signum differencing method for $\Delta\psi$. Due to the nature of a virtual environment, new criteria implemented for handling tracked feature points, such that both their horizontal and vertical LOS rates must be below a threshold to be considered for cost function minimization, rather than just the horizontal LOS rate being used previously. Figure 55 shows an instantaneous capture of the camera view point with overlaid feature points selected for the avoidance law to use in determining the UAV maneuver.



Figure 55: Feature point overlay of UAV camera view for single building avoidance

Figure 56 shows the result of this cost function avoidance implemented for the virtual environment. The left image in this figure is representative of the physical travel and avoidance distances while the image on the right shows a top down view of the path in relation to the virtual environment. The paths shown are considered successful avoidances and resembles those seen in the previous point-mass simulation. Similarities can also be seen in the recorded heading history.

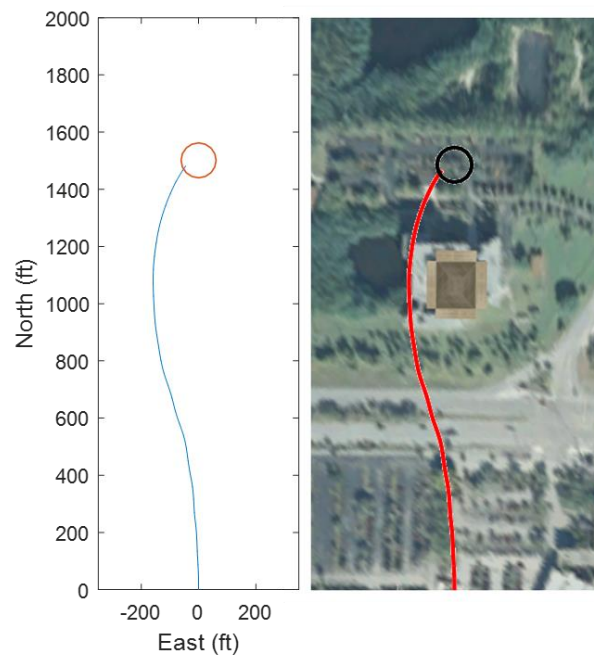


Figure 56: UAV avoidance path for single building MetaVR simulation

Figure 57 shows similar trends in the heading history for the single building MetaVR simulation as seen in previous point-mass simulations. There is a slight oscillation in the heading but an overall increase related to the avoidance maneuver around the building, followed by a decrease in heading while the UAV returns to a waypoint intercept trajectory. The slight oscillation in the later portion of the history reflects on the temporary appearance of feature points detected along the textured ground and horizon line, something that did not need consideration for the point mass simulations.

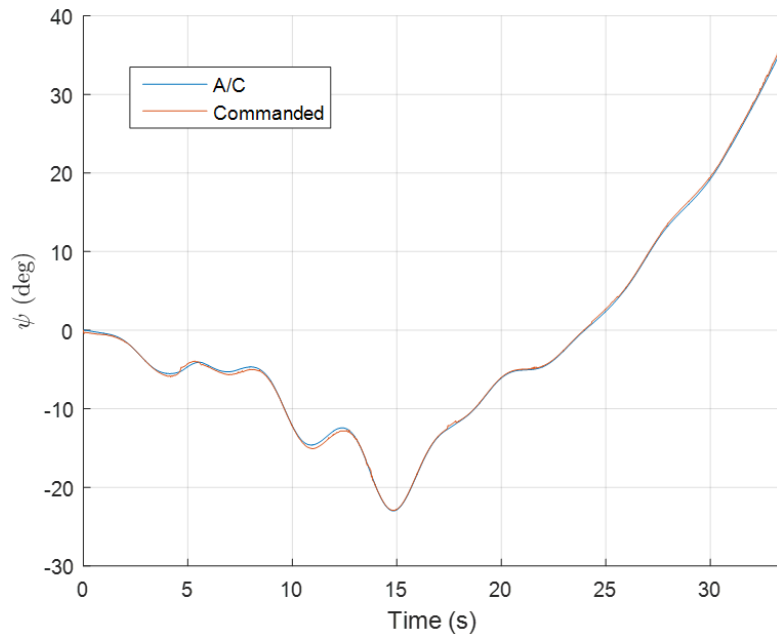


Figure 57: UAV heading history for single building MetaVR simulation

8.4. Urban Canyon Avoidance

The next scenario presented in this section places the UAV in the middle of what can be called an “urban canyon”. The reason for this scenario description becomes apparent in Figure 58, in which buildings of a custom urban setup surround the UAV on the left and right side, resembling the geography seen by natural canyons. The objective waypoint is placed directly in front of the UAV at 2,000 *ft*, giving the UAV avoidance

law a relatively simple, but enlightening task of navigating the UAV straight through the urban canyon. As mentioned for the scenario of a single building, the waypoint is still represented by a point mass, whose LOS angle and rate measurements are derived using the same method of the previous point mass simulations. Figure 58 shows the initial camera view captured from MetaVR, with total vertical and horizontal view angles of 36° and 60° , respectively.

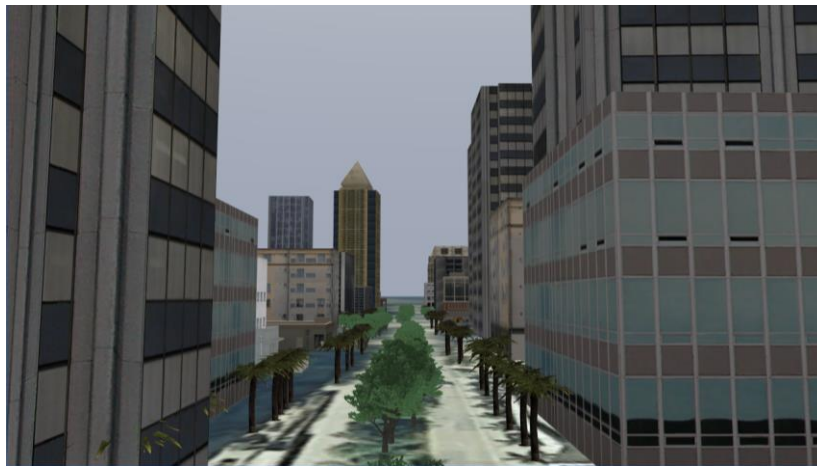


Figure 58: Camera view from UAV of Urban Canyon MetaVR simulation

Using the chosen cost function-based avoidance law that employs the signum differencing method for $\Delta\psi$, Figure 60 shows a successful UAV path with regards to both physical distances, on the left-hand side, and the virtual environment, on the right-hand side. The path is not perfectly straight as one might expect, due to the location of the feature points detected within the field of view and the number of feature points that have horizontal LOS rates below the threshold for avoidance. A scene capture with overlaid feature points can be seen in Figure 59 for a single instance of the entire UAV flight.



Figure 59: Feature point overlay of UAV camera view within the urban canyon

In Figure 60, the initial deviation to the right indicates that the best course of action for the UAV was to maneuver slightly until more threatening features on the left-hand side of the image were no longer a threat, or until the threat balanced out with potential threats on the right-hand side. Later in the path, slight deviations can be seen as the UAV briefly turns towards the left, but returns to a far-right position as before. Finally, the UAV makes a maneuver to intercept the waypoint as the waypoint LOS rate increases, outweighing any potential threats along the sides of the canyon.

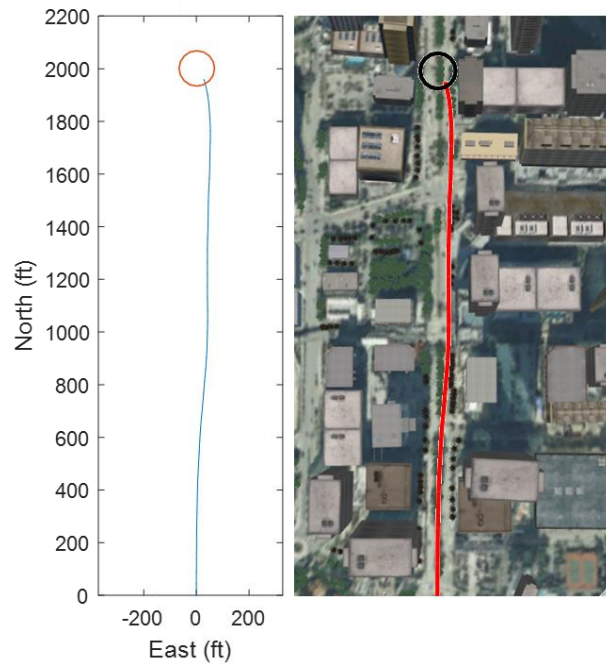


Figure 60: UAV avoidance path for Urban Canyon MetaVR simulation

The UAV heading history for the urban canyon scenario can be seen Figure 61. Here, the heading trend throughout the entire flight correlates to the flight path shown in in Figure 60 and also provides insight on the not so obvious smaller deviations that the UAV exhibited during the flight. Small oscillations are seen throughout the process of increasing and decreasing the UAV heading, due to the maneuver placing old feature points outside the threatening threshold of LOS rate and placing newer points within that threshold, creating a temporary new heading. This oscillation is similar to that of the single building scenario heading history, seen in Figure 57. However, in this scenario, due to more buildings and potential feature points being present, the effect is more noticeable.

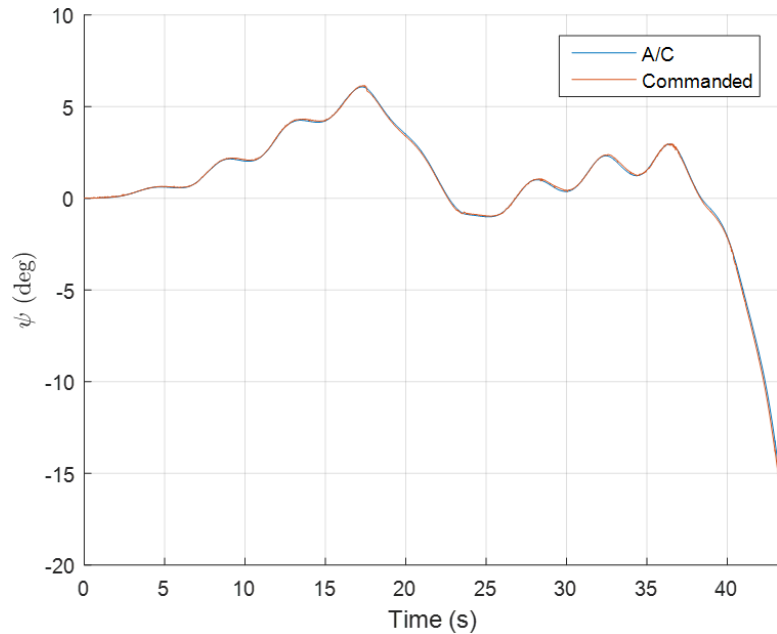


Figure 61: UAV heading history for Urban Canyon MetaVR simulation

8.5. Urban Navigation and Avoidance

This final simulation constructed in the virtual environment places the UAV in a location that presents a more challenging scenario. Figure 62 shows a viewpoint from the UAV at its starting location in the simulation with the waypoint placed 4,000 *ft* north, placing it beyond the urban environment area. The objective for the UAV was to navigate through this environment to reach the waypoint without colliding with any obstacles.

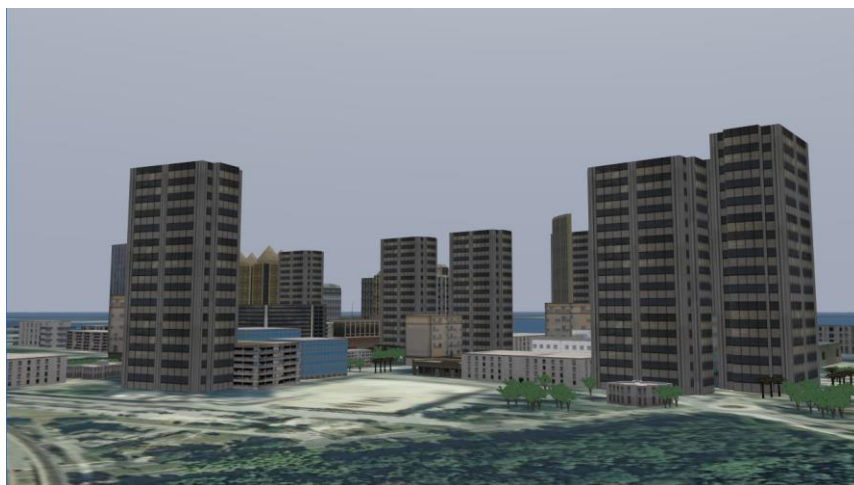


Figure 62: Complex Virtual Urban Environment

The UAV flight path generated in Figure 63 by the avoidance law navigates the UAV semi-successfully, creating a collision with a single building during flight, noted by the yellow square in the figure. The UAV continues on a clear avoidance path the rest of the flight until it reaches the waypoint, leading between two building sets that can be considered close encounters; however, this was an act of precision rather than random maneuvering. The left-hand side of Figure 63 shows the physical distances travels by the UAV while the right-hand side shows the avoidance path in relation to the virtual environment.



Figure 63: UAV path for complex urban virtual reality scenario

The UAV avoidance path results in a semi-successful attempt, avoiding all but one building during the flight. This instance can be shown in Figure 64, in which the UAV viewpoint with feature points shows that the building which was collided with had a lack of feature points relative to the rest of the scene, until the UAV was much closer to it. This prevented the avoidance law from accounting for the building earlier, and once it was detected, it was too late to make a significant avoidance maneuver.



Figure 64: UAV complex urban virtual scenario - pre-collision

One of the two close encounters can be seen in Figure 65 as an example of how close the UAV comes to the obstacles. Because the space chosen to navigate through is particularly small, this encounter can be considered an achievement of the avoidance law regarding its ability to navigate the UAV through safe zones, regardless of the available space.

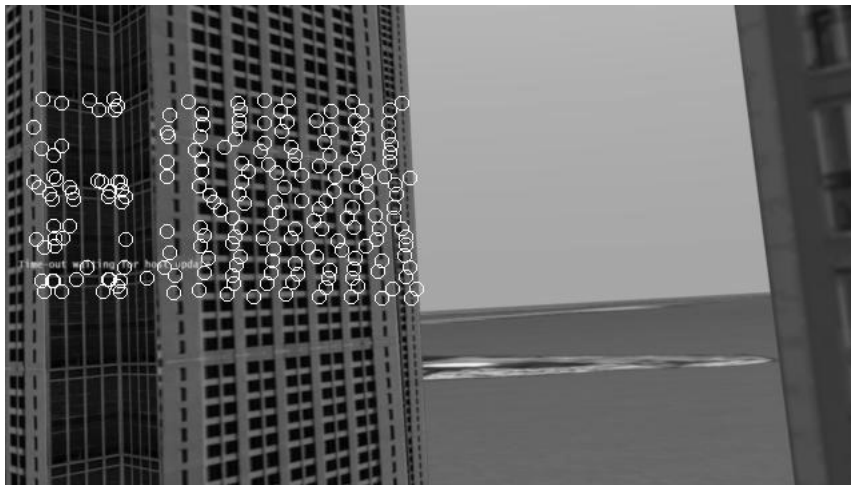


Figure 65: Complex urban virtual scenario – post-collision

Overall, the implementation of the chosen cost function using $\Delta\psi$ proved successful for all scenarios except for one. In that scenario, the limitation seemed to be related more to the feature point detection capabilities or the feature qualities of the building itself.

9. CONCLUSION

Current SAA technologies and algorithms provide reliable but costly means of detection and avoidance for UAVs with regards to cost and computational power. Electro-optical devices have received considerable interest due to their light weight, low cost, and low algorithm requirements with respect to computational power. For this reason, this thesis has investigated UAV obstacle avoidance and navigation using vision-based Pro-Nav guidance laws without the use of range information. The development of these guidance laws propagates from an investigation into numerous methods of guidance, including the use of LOS rate thresholding, avoidance of the most imminent threat detected, and objective based cost optimization. A final form of the avoidance law, determined using point-mass simulations, was applied to a real world, virtual environment which utilized “camera-in-the-loop” simulation techniques.

The path of this thesis has taken a developmental approach of a final Pro-Nav avoidance law capable of avoiding dynamic and static obstacles using Pro-Nav concepts, without the use of range information. The approach systematically characterized guidance law traits that contribute both positively and negatively, such that these could be taken into consideration for the next iteration of avoidance law development.

Pro-Nav guidance concepts were first applied for collision avoidance by using a LOS rate threshold. The threshold implemented produced a defined heading command when the magnitude of an obstacle’s horizontal LOS rate, $\dot{\chi}$, was smaller than the specified threshold, which indicates a potential collision threat. Using a single obstacle on an intercepting trajectory, the results showed that the LOS rate and change in LOS rate of an intercepting or near intercepting obstacle will vary depending on initial obstacle position

and velocity. Obstacles that have a collision path occurring further away from the UAVs initial position have a greater avoidance distance generated between the UAV and obstacle. In similar fashion, UAV avoidance distances will be larger when obstacles with a larger velocity are on an intercepting trajectory.

The success of threshold-based avoidance prompted its application to a multi-obstacle scenario, in which two obstacles were placed on a path that would collide with the UAV if no avoidance was made. The same avoidance law was applied, but a modification was required, such that the obstacle of most imminent threat (i.e., with smallest LOS rate) would be actively avoided using the same thresholding heading commands. Selected simulations placed the obstacles in multiple locations, such that investigation into the reactivity of the guidance law could be performed. In this simulation, a primary obstacle was placed on an initial trajectory that intercepted the UAV along the north axis while a secondary obstacle was placed such that an avoidance of the primary intercepting obstacle would place the UAV on a collision course with the other. These simulations showed that due to the smaller change in LOS rate from the obstacle far away, avoidance was prioritized around obstacles that were further away. This led to multiple cases in which avoidance of at least one obstacle was comparatively small. This method of guidance was also shown to fail in some static obstacle scenarios, in which the UAV would be bound between avoiding the two most imminent threats, inducing an oscillating avoidance path. The overall interpretation of this avoidance law was that, although successful in single and some multi-obstacle avoidance cases, the lack of consideration of all threatening obstacles in the guidance law led to insufficient avoidance paths in many cases.

A set of objective-based cost functions was then investigated as an alternative to the most imminent threat method. The cost functions were designed to consider a primary objective of intercepting a waypoint, while taking into consideration all obstacles that posed a potential collision threat. This definition of collision threat relates back to the thresholding technique, such that any obstacle in view with a horizontal LOS rate below a specified value was considered and applied to the optimization of the cost function. It was also clear that applying a defined, constant heading change as an avoidance maneuver was not specific enough to relate the avoidance maneuver to the obstacle dynamics. For this reason, three Pro-Nav parameters $\dot{\chi}$, $\dot{\psi}$, and $\Delta\psi$ were applied to cost functions to provide avoidance maneuvers that had a connection to the obstacle dynamics and to investigate the similarities and differences, if any, between using one Pro-Nav parameter over the others for the primary avoidance law. The cost function types used the inverse Pro-Nav, negative Pro-Nav, and Pro-Nav difference to generate an optimized commanded guidance parameter. The results of simulating UAV avoidance with these three cost function sets showed that using the horizontal LOS rate as an optimization variable provided the most minimal, but successful avoidance of dynamic obstacles, but failed to avoid the static wall of obstacles between the UAV and waypoint successfully. The use of the inverse Pro-Nav cost function for every scenario using all three Pro-Nav variables resulted in excessive avoidance and aggressive maneuvers. The optimization of $\Delta\psi$ using the LOS rate differencing cost function showed the best overall performance, providing smooth transitioning avoidance maneuvers while not exhibiting overly aggressive heading commands. This method was therefore chosen for further development and testing using a variety of other scenarios to determine if any

modifications may produce better results or to find negative characteristics that were not apparent in the initial simulation scenarios.

The final set of point-mass simulations correspond to an arrangement of static and dynamic obstacles combined into a single scenario along with a simulated “urban canyon” of static points. The combined static and dynamic scenarios utilized two static point-mass walls that were varied in location and position and placed in particularly inconvenient locations relative to the path that would be generated by the avoidance law if only the dynamic obstacles were active. The results from these simulations showed that the UAV was capable of avoiding all obstacles; however, in particular cases the avoidance distance was much less than previously shown with individual dynamic and static avoidance configurations.

Finally, a virtual environment was used in conjunction with the simulation environment to provide a realistic, camera-in-the-loop scenario, which provided images of the virtual environment that were used for obtaining feature points. These feature points were used in the chosen cost function guidance law for navigation of the UAV in an urban environment. Scenarios implementing a single building, a straight-line path through an urban canyon, and a complex navigation of an urban environment provided substantial results to support the effectiveness of the cost function guidance law using $\Delta\psi$ as a solution for vision-based Pro-Nav.

Future considerations and development for this proposed vision-based Pro-Nav law revolve around adapting the guidance law to better fit three-dimensional avoidance maneuvers in realistic engagement scenarios. This would first require work in determining an acceptable choice of action in the horizontal and vertical flight path

direction when avoidance maneuvering is required. This would then provide a stable basis for dynamic scenarios to be simulated in the virtual environment using actual 3-D models of small or large-scale obstacles. This is also currently limited by the interface between MetaVR and simulation environment not being capable of controlling 3-D obstacles (only viewpoints are currently controlled). A different image processing method can also be implemented to identify borders of building and dynamic obstacles as single entities, rather than hundreds of feature points spread out across the image, acting individually regardless of whether they are representative of the same object.

REFERENCES

- Altman, A. (1998). Design Methodology for Low Speed High Altitude Long Endurance Unmanned Aerial Vehicles. *21st International Council of Aeronautical Sciences Congress* (p. Paper 133). Melbourne, Australia: Mechanical and Aerospace Engineering Publications.
- Angelov, P. P. (2012). *Sense and Avoid in UAS: Research and Applications*. Hoboken, New Jersey: John Wiley & Sons.
- Beard, R. W. (2006). Decentralized Cooperative Aerial Surveillance Using Fixed-Wing Miniature UAVs. *Proceedings of the IEEE*, 1306–1324.
- Beard, R., Curtis, J., Eilders, M., Evers, J., & Cloutier, J. (2007). Vision Aided Proportional Navigation for Micro Air Vehicles. *AIAA Guidance, Navigation, and Control Conference*. Hilton Head, SC: American Institute of Aeronautics and Astronautics.
- Billingsley, T., Kochenderfer, M., & Chryssanthacopoulos, J. (2011). Collision Avoidance for General Aviation. *Digital Avionics Systems Conference* (pp. 1-17). IEEE/AIAA.
- Chen, W., Cao, L., Wu, Z., & Huang, S. (January 2014). A Stereo Range Computation Method Using Thermal Infrared and Visible Cameras. *Infrared Physics & Technology* 62, 115-123.
- EuroControl. (n.d.). *ACAS Overview Principles*. Retrieved from Eurocontrol: http://www.eurocontrol.int/msa/public/standard_page/ACAS_Overview_Principles.html
- Gageik, N., Strohmeier, M., & Montenegro, S. (2013). An Autonomous UAV with and Optical Flow Sensor for Positioning and Navigation. *International Journal of Advanced Robotic Systems*.
- Geyer, C., Dey, D., & Singh, S. (2009). *Prototype Sense-and-Avoid System for UAVs*. Pittsburgh: Robotics Institute, Carnegie Mellon University.
- Griffith, J., Kochenderfer, M., & Kuchar, J. (2008). Electro-Optical System Analysis for Sense and Avoid. *AIAA Guidance, Navigation and Control Conference and Exhibit*. Honolulu: American Institute of Aeronautics and Astronautics.
- Han, S., Bang, H., & Yoo, C. (2009). Proportional Navigation-Based Collision Avoidance for UAVs. *International Journal of Control Automation, and Systems*, 553-565.
- Kim, B. S., Lee, J. G., & Han, H. S. (1998). Biased PNG law for impact with angular

- constraint. *IEEE Transactions on Aerospace and Electronic Systems*, 277-288.
- Kim, M., & Grider, K. V. (1973). Terminal Guidance for Impact Attitude Angle Constrained Flight Trajectories. *IEEE Transactions on Aerospace and Electronic Systems*, 852-859.
- Lau, D. (2012, April 1). *3-D Imaging Advances Capabilities of Machine Vision: Part I*. Retrieved from Vision Systems Design: <http://www.vision-systems.com/articles/print/volume-17/issue-4/departments/leading-edge-views/3-d-imaging-advances-capabilities-of-machine-vision-part-i.html>
- Lentilhac, S. (2009). UAV Flight Plan Optimized for Sensor Requirements. *International Radar Conference - Surveillance for a Safer World* (pp. 1-4). IEEE.
- Losey, L. (2001, 8 1). *Lockheed ER-2 #809 high altitude research aircraft in flight*. Retrieved from Dryden Flight Research Center Photo Collection: <https://www.dfrc.nasa.gov/Gallery/Photo/ER-2/HTML/EC01-0232-6.html>
- Moses, A., Rutherford, M., & Valavanis, K. (2011). Radar-Bsed Detection and Identification for Miniature Air Vehicles. *IEEE International Conference on Control Application (CCA)*. Denver: Institute of Electrical and Electronics Engineers.
- MrDrone.net. (2017). *"History of Drones – MrDrone.Net – All things Drone."*. Retrieved from MrDroneNet All things Drone.: <http://www.mrdrone.net/history-of-drones/>
- Mueller, T. J. (2009). On the Birth of Micro Air Vehicles. *International Journal of Micro Air Vehicles*, 1-12.
- Murtaugh, S. A., & Criel, H. E. (1966). Fundamentals of proportional navigation. *IEEE Spectrum*, 75-85.
- Nelson, R. (1998). *Flight Stability And Automatic Control*. McGraw-Hill Higher Education.
- OpenMVG. (2013). *Cameras*. Retrieved from OpenMVG.readthedocs.io: <http://openmvg.readthedocs.io/en/latest/openMVG/cameras/cameras/>
- Ryabinin, A. (2017, January 18). *VisionLiDAR*. Retrieved from http://www.geoplus.com/visionlidar_lidar_software/
- SARA Inc. (2012). *UAV Acoustic Collision - Alert System*. Retrieved from SARA Inc.: http://www.sara.com/isr/uav_payloads/pancas.html
- Sharma, R., Saunders, J. B., & Beard, R. W. (2012). Reactive Path Planning for Micro Air Vehicles Using Bearing-only Measurements. *Journal of Intelligent & Robotic Systems*, 409-416.

- Trinh, M. H., Ko, G.-H., Pham, V. H., Oh, K.-K., & Ahn, H.-S. (2012). Guidance using bearing-only measurements with three beacons in the plane. *Control Engineering Practice*, 81-91.
- US Navy. (2008, July 5). *Photo ID 080805-N-0000X-001*. Retrieved from Navy.mil: <http://www.navy.mil/management/photodb/photos/080805-N-0000X-001.jpg>
- Valovage, E. (2007). Enhanced ADS-B Research. *IEEE Aerospace and Electronic Systems Magazine* 22.5, pp. 35-38.
- Yu, X., & Zhang, Y. (2015). Sense and Avoid Technologies with Applications to Unmanned Aircraft Systems: Review and Prospects. *Progress in Aerospace Sciences* 74, 152–166.
- Zeitlin, D. A., & McLaughlin, M. P. (2007). Safety of Cooperative Collision Avoidance for Unmanned Aircraft. *IEEE Aerospace and Electronic Systems Magazine* 22, pp. 9-13.
- Zhao, S. e. (2012). Homography-Based Vision-Aided Inertial Navigation of UAVs in Unknown Environments. *AIAA Guidance, Navigation, and Control Conference*. Minneapolis: American Institute of Aeronautics and Astronautics.