Trajectory Optimization with Detection Avoidance for Visually Identifying an Aircraft

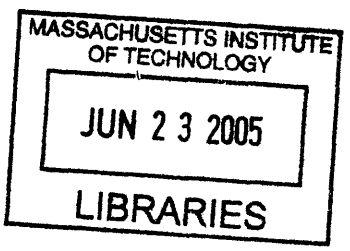
by

Leonard N. Wholey

B.S. Aeronautical Engineering
United States Air Force Academy, 2003

SUBMITTED TO THE DEPARTMENT OF AERONAUTICS AND ASTRONAUTICS
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE IN AERONAUTICS AND ASTRONAUTICS
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY



June 2005

Signature of Author _____
Department of Aeronautics and Astronautics
May 6, 2005

Certified by _____
Leena Singh
Charles Stark Draper Laboratory
Thesis Supervisor

Certified by _____
Brent Appleby
Lecturer in Aeronautics and CSDL Technical Supervisor
Thesis Advisor

Accepted by _____
Jaime Peraire
Professor of Aeronautics and Astronautics
Chair, Committee on Graduate Students

[This page intentionally left blank.]

Trajectory Optimization with Detection Avoidance for Visually Identifying an Aircraft

by

Leonard N. Wholey

Submitted to the Department of Aeronautics and Astronautics on
May 6, 2005, in partial fulfillment of the requirements for the
Degree of Master of Science in Aeronautics and Astronautics

Abstract

Unmanned aerial vehicles (UAVs) play an essential role for the US Armed Forces by performing missions deemed as "dull, dirty, and dangerous" for a pilot. As the capability of UAVs expand, they will perform a broader range of missions such as air-to-air combat.

The focus of this thesis is forming trajectories for the closing phase of an air-to-air combat scenario. A UAV should close with the suspected aircraft in a manner that allows a ground operator to visually identify the suspected aircraft while avoiding visual/eletronic detection from the other pilot.

This thesis applies and compares three methods for producing trajectories which enable a visual identification. The first approach is formulated as a mixed integer linear programming problem which can be solved in real time. However, there are limitations to the accuracy of a radar detection model formed with only linear equations, which might justify using a nonlinear programming formulation. With this approach the interceptor's radar cross section and range between the suspected aircraft and interceptor can be incorporated into the problem formulation. The main limitation of this method is that the optimization software might not be able to reach online an optimal or even feasible solution. The third applied method is trajectory interpolation. In this approach, trajectories with specified boundary values and dynamics are formed offline; online, the method interpolates between the given trajectories to obtain similar maneuvers with different initial conditions and end-states. With this method, because the number of calculations required to produce a feasible trajectory is known, the amount of time to calculate a trajectory can be estimated.

Thesis Supervisor: Dr. Leena Singh
Title: Senior Member of the Technical Staff

Thesis Advisor: Dr. Brent Appleby
Title: Division Leader - Control, Information, and Decision Systems

[This page intentionally left blank.]

Acknowledgments

I would like to thank my advisors Leena Singh and Brent Appleby for their help with this thesis. Leena, thank you for taking time out of your busy schedule to offer new ideas and point me in the right direction. I would also like to thank Brent and Draper Lab for choosing me as a Draper Fellow. This has been a great opportunity and would not have been possible without their support. Brent, thank you for your generosity with your time.

I would also like to thank a few other Draper Lab employees: Mark Homer, Chris Dever, and Jeff Cipolloni. Thank you for sharing your ideas and motivating me to pursue them.

This work could not have been possible without the advice from the following fighter pilots: Col John Kuconis, Maj Joseph Jezairian, Lt Col Barry Watts, Lt Mark Jackson, Lt Col Raymond O'Mara, and Lt Col Gregory Marzolf. Thank you for teaching me about the fascinating world of air-to-air combat.

Mom and Dad, thank you for your endless support. Thank you for always being there for me.

Craig, thanks for being a great friend. Just two weeks until hugeness.

Maria, you truly are the greatest. Thank you for your words of encouragement.

This thesis was prepared at The Charles Stark Draper Laboratory, Inc., under Internal Company Sponsored Research Project 12601, GCC DLF Support.

Publication of this thesis does not constitute approval by Draper or the sponsoring agency of the findings or conclusions contained herein. It is published for the exchange and stimulation of ideas.

As a member of the Air Force, I acknowledge that the views expressed in this thesis are mine and do not reflect the official policy or position of the United States Air Force, the Department of Defense or the United States Government.

Leonard N. Wholey

[This page intentionally left blank.]

Contents

[This page intentionally left blank.]

List of Figures

[This page intentionally left blank.]

List of Tables

[This page intentionally left blank.]

Chapter 1

Introduction

Unmanned aerial vehicles (UAVs) such as the Predator have proven their value by performing reconnaissance missions and accurately launching hellfire missiles on ground targets. It seems feasible that in the near future UAVs will perform additional missions associated with fighter pilots. Such a vehicle would be able to fight aggressively and if shot down, would not have some of the undesirable consequences of a piloted aircraft. For instance, a search and rescue mission is not necessary for a UAV.

An autonomous UAV which can perform all of the required missions with proficiency commensurate to a human pilot would be very difficult to design. There are many situations throughout even the latest conflicts where a human's cognition and situational awareness avoided grave consequences. Major Robert Nolan describes one such incident during the first Gulf War where Capt Landis Cook, an A-10 pilot, was given clearance to fire on a convoy of tanks [21]. However, Capt Cook decided to visually identify the targets despite the aircraft controllers' assertion that there were no friendly forces in the area. After approaching the suspected targets, Capt Cook identified the tanks as British Challengers. It seems that only a UAV with the trait of what Major Nolan refers to as "judgment" would be able to properly handle this situation. The current level of technology has not reached this difficult goal, and as a result, a human in the loop will be necessary.

Including a human in the decision making process for UAVs may involve more than choosing whether or not to destroy a target. But, certain decisions seemingly

can be made within the cockpit or a ground control unit. Lt Col David Brown's experience with the USAF's drone program for four and a half years during the 1990's demonstrates this point [5]. While flying the QF-106 from the ground against manned aircraft in a dogfight scenario, Brown made the observation, "The ONLY combat capability the QF-106s lacked that day was offensive firepower!"

UAV's with the capability of performing an air-to-air intercept could have been useful in recent conflicts. During Operation Desert Storm, the Coalition Forces were able to quickly obtain and maintain air superiority. Coalition aircrews achieved a 33-1 air-to-air kill ratio, and Iraqi pilots essentially ceased air combat operations within the first two weeks of the conflict [10]. However, as many as 121 Iraqi aircraft escaped to Iran when Iraqi pilots realized their inability to fly against the better trained and equipped Coalition pilots [10].

Stopping these aircraft from fleeing the country proved to be a very difficult task [10]. Coalition fighter pilots guarded the border between Iraq and Iran with missions reaching as a long as seven hours. Additionally, airborne warning and control system (AWACS) aircraft were unable to detect all of the Iraqi aircraft because of the low altitudes at which the Iraqi pilots fled. Finally, Iraqi pilots would often wait until there were holes in the Coalition's barrier or when there was no barrier at all. Considering the limited amount of Coalition aircraft and pilots which may have been assigned to more important missions such as close air support, UAVs capable of performing air-to-air intercepts would have been ideal for this situation.

1.1 Objective

A UAV may be able to conduct successful air-to-air intercepts even without the capability to dogfight. During the Gulf War of 1991, over 40% of the air-to-air kills were performed in the beyond visual range region mainly due to improved radar and missile technology [20]. However, if dictated by the rules of engagement or possibly by an imperfect electronic identification system, a visual identification (VID) is necessary before firing a missile. A VID should be performed in minimum time while reducing

the chances of being detected by the suspected aircraft.

This thesis addresses some options for improving a fighter pilot's ability to perform a VID on a single suspected aircraft. A computer could cue the pilot to an optimal trajectory which could decrease the amount of time required to complete the maneuver and reduce the chances of being detected by radar by managing range and attitude. Additionally, this scenario could be adapted to a UAV equipped with a camera in order that a ground based operator could ultimately identify the suspected aircraft.

Three methods for calculating trajectories which enable a VID are applied in this thesis. The first method is mixed integer linear programming (MILP), which has been shown to be a powerful and effective method for solving rendezvous problems in real time [26]. The drawback of this approach for low-visibility trajectories is that it is difficult to accommodate certain nonlinear criteria, such as radar cross section, involved in producing a stealthy trajectory. As a result, the problem is also formulated as a nonlinear program using a direct trajectory optimization method sometimes referred to as direct collocation [16]. Although this allows a more accurate radar avoidance constraint, a real time solution is not guaranteed. A trajectory interpolation approach, which is the third applied method, may be one solution to this problem [12]. A library of maneuvers parameterized by boundary values and aircraft dynamics are calculated offline; online, the automatic trajectory synthesis interpolates to obtain a desired trajectory.

1.2 Thesis Outline

The thesis is organized as follows: Chapter 2 begins with a description of the different phases of air-to-air combat. It describes in detail the different tactics used for the closing phase which is used to visually identify the suspected aircraft. Chapter 3 presents a brief summary of the different techniques used for trajectory optimization within the past 35 years. Chapters 4-6 contain the different optimization methods used in producing trajectories. Finally, Chapter 7 concludes the thesis with a summary and suggestions for future research.

[This page intentionally left blank.]

Chapter 2

Air-to-Air Combat Phases

2.1 Five Phases of Aerial Engagements

The technology and tactics of air-to-air combat have changed significantly since the first recorded dogfight. However, Mike Spick, author of *Fighter Pilot Tactics*, states that there are five phases of aerial engagements which have remained throughout air-to-air combat's history: (1) detection, (2) closing, (3) attack, (4) maneuver, (5) disengagement [33].

Detection is extremely important for air-to-air combat because it provides the pilot with the opportunity to make the first decisions. For the oldest cases of aerial engagements, pilots only could detect the enemy visually. Inexperienced pilots suffered the highest casualty rates by focusing on flying the aircraft rather than maintaining their situational awareness. As a result, many pilots were shot down without even seeing their adversaries.

Detection remained essential in more recent versions of air-to-air combat. F4 pilot Bill Jenkins commented about the importance of spotting the enemy during the Vietnam war, "The MIG 21 is a very small airplane. From head on it's difficult to see at more than 2 miles although a formation is easier. American aircraft were much larger and could be seen at greater distances, not least due to the smoky exhaust trail" [33]. Although the US possessed missiles which could function beyond visual range (BVR), two incidences of fratricide forced the pilots to return to positive visual

identification [33].

Throughout Operation Desert Storm superior technology allowed Coalition fighter pilots to enjoy significant advantages in detecting Iraqi aircraft. The Airborne Warning and Control System (AWACS), which is capable of detecting aircraft hundreds of miles away, soon became know as the "third wingman" [19]. Additionally, non-cooperative target identification systems installed on aircraft allowed fighter pilots to determine independent of an AWACS whether an aircraft is friendly [19]. These advancements directly contributed to the large number of BVR engagements. However, William Lewis an F-15E pilot states, "This ideal long-range engagement cannot always occur, though, because of enemy defensive maneuvering and the constraints imposed by rules of engagement [20]."

Closing is the second phase which would be implemented when BVR combat is not feasible. A beam intercept is one possible maneuver used to close with the suspected aircraft. This method decreases the chances of being visually/electronically detected by the enemy while increasing the chances of making a visual identification (VID).

The Israeli Air Force completely routed Syria's fighter planes during the Bekáa Valley conflict in June of 1982 by using a sound strategy for the closing phase:

> Although the IAF later maintained that it took no shots at Syrian fighters from beyond visual range, it evidently made extensive use of blind-side tactics by employing the E-2C to vector F-15s and F-16s into beam attacks against Syrian MIGs, where their radar warning systems were reportedly least effective [19].

At the end of the conflict, the Israelis had destroyed 85 Syrian aircraft without any losses to its own fighter force [19].

After the fighter pilot has detected the target and begun the closing phase, he must consider when to fire his munitions, which is referred to as the attack phase. Throughout World War I and much of World War II, the desired attack position was from behind the enemy's aircraft. Attacks from astern gave the fighter pilot the best possible firing opportunity as well as a good defensive position. However, this

position was no longer required with the advent of homing missiles. Spick now defines the attack phase as "the moment when the attacker starts to align his weapons [33]."

The fourth phase is maneuver; this phase is also known colloquially as "dogfight" [33]. Spick states, "It is the most spectacular phase of combat and for this reason is often considered to be the most important [33]." History has shown that this is not the case; only about one in five victims of air-to-air combat are shot down during the maneuver phase [33]. Nevertheless, maneuvering is crucial when the first three phases do not finish the intercept. This would be the most difficult phase for a UAV to accomplish due to the high level of situational awareness required to win a dogfight.

The final phase is disengagement, which although important for a UAV is not nearly as essential as for a manned aircraft. A UAV can pursue a target until it expends its fuel, whereas a piloted aircraft must return to base after reaching his "bingo" fuel state. However, if the UAV must leave the fight, Spick describes two methods of disengagement:

> If the attacker is at close range, i.e., 800 yds or less, the hardest possible turn into the direction of attack would cause the enemy fighter to overshoot it. At longer, i.e., missile ranges, disengagement can be made by a series of turns hard enough to make it difficult for the attacker to achieve a good missile firing position, but not so hard that an undue amount of speed is lost [33].

The five phases of air-to-air combat parameterize an air-to-air intercept. This thesis focuses on the closing phase of aerial combat. The goal is to produce optimal trajectories which effect a VID in minimum time, while reducing the chances of the target detecting the interceptor.

2.2 Closing Phase

As previously mentioned, missile technology is such that an aircraft can be destroyed in the BVR region. Merges with suspected aircraft will still occur if the electronic identification is not working, or the rules of engagement dictate a visual identification.

In order to visually identify the aircraft, a UAV could carry a camera slewed to the radar track of the suspected aircraft. This will allow a human to make the final decision on whether or not to destroy the target. Ideally. this will result in earlier VID's because the ground operator can focus on identifying the aircraft rather than the additional task of flying.

Four tactics for performing a VID are described in this section: (1) proportional navigation, (2) forward-quarter intercept, (3) stern conversion, (4) beam intercept.

2.2.1 Proportional Navigation

Using proportional navigation is potentially the fastest method for an interceptor to merge with the target aircraft. This technique is commonly used for missile guidance but can also be used by an aircraft. The objective of proportional navigation is to minimize $|\dot{\lambda}|$, where λ is the line of sight angle, shown in Figure 2-1.

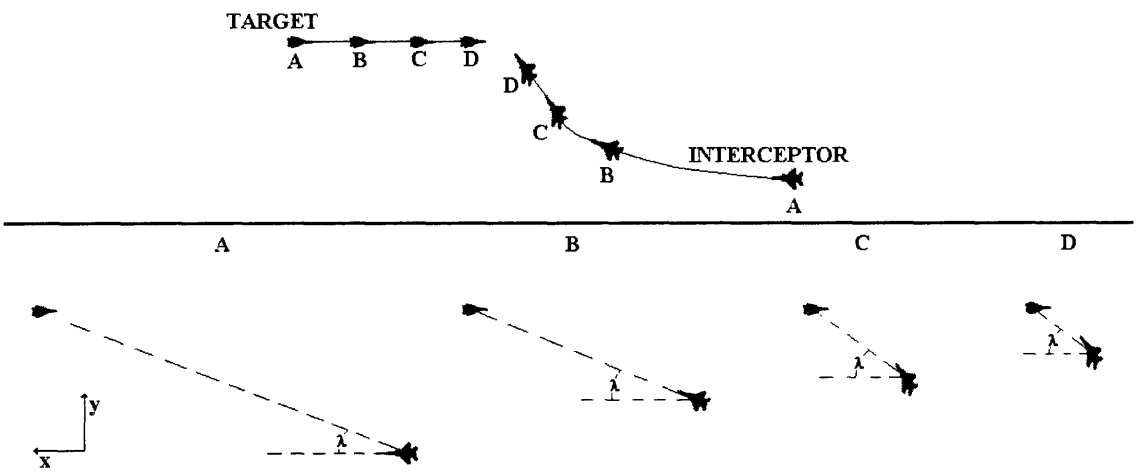


Figure 2-1: The top figure shows an example target-interceptor trajectory formed using proportional navigation. The bottom figure shows values of line of sight angle λ at specific points along the trajectory. λ is measured as the angle between the x-axis and the line connecting the interceptor and target. At points A and B the interceptor turns in order to obtain a constant value of λ which occurs at points C and D. If the two aircraft continue along their respective headings at point D, they will collide.

The guidance law for two-dimensional proportional navigation is described in [38]

as

$$n_c = N' V_c \dot{\lambda} \tag{2.1}$$

where n_c is the acceleration command which is perpendicular to the interceptor's velocity vector, N' is a user specified gain, V_c is the the closing velocity, and $\dot{\lambda}$ is the rate of change of the line of sight angle. V_c and $\dot{\lambda}$ are derived with a few simple equations in [38].

2.2.2 Forward-Quarter Intercept

When a visual identification is required, the forward-quarter (FQ) intercept is also a viable tactic. The goal of this maneuver is for the interceptor to position itself with a good firing opportunity while denying one to the target. The FQ intercept achieves two specific objectives. It obtains a certain target aspect angle (TAA), which is defined in Figure 2-2. It also achieves a desired amount of distance to be traveled

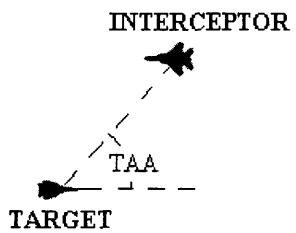


Figure 2-2: Target aspect angle (TAA) is the angle between the target's velocity vector and the line of sight vector between the interceptor and the target.

along the final collision heading which is from point 2 to point 3 in Figure 2-3.

There are several advantages to using this tactic. One is that some missiles work more effectively from an FQ firing position than from head-on. Additionally, an FQ position can increase the range of visual identification because more area of the target is exposed from the side than straight on [32]. However, this maneuver may take longer to achieve a firing solution than if the aircraft were to use proportional navigation.

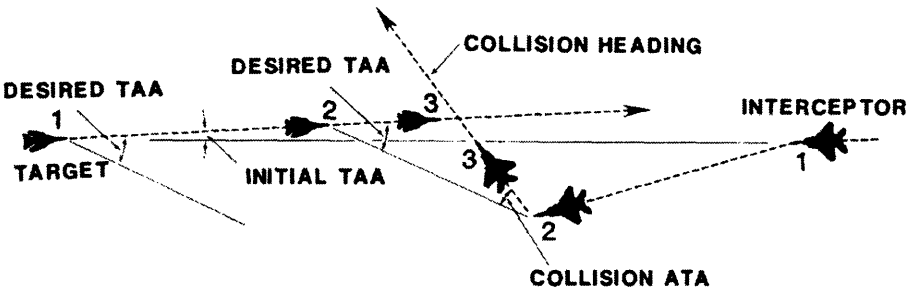


Figure 2-3: The FQ intercept consists of two turns. The purpose of the first turn at point 1 is to increase the TAA. The interceptor makes a turn away from the target and then continues along a straight line until the desired TAA is obtained at point 2. Then, the interceptor makes another turn to maintain the TAA. Both aircraft are on collision course at point 3. [32]

2.2.3 Stern Conversion

Another option for the pilot to merge with a suspected aircraft is to perform a stern conversion [32]. This maneuver positions the interceptor directly behind the target aircraft, which is a highly advantageous position.

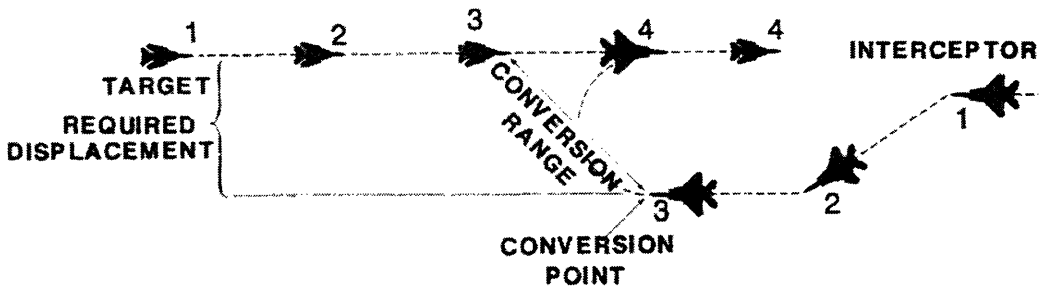


Figure 2-4: The steps to determining this trajectory occur in reverse order to the actual performance of the maneuver. The first is to decide the final desired range between the target and the interceptor once the maneuver is completed at point 4. The second step is to determine the turn radius between points 3 and 4 to place the interceptor behind the target at the desired range. Typically, a turn which points the interceptor at the target throughout the majority of the turn is chosen. The third step is to calculate the conversion range and required displacement at point 3, which will enable a successful completion of the maneuver. The final step is to fly to the conversion point, which consists of two turns occurring at points 1 and 2. [32]

The major advantage to performing this maneuver is that the target will have great difficulty obtaining a firing opportunity once the trajectory is completed. However,

there are several disadvantages which make this approach atypical for pilots to use. This is by far the longest duration trajectory of the four methods for performing a VID. Also, with the advent of all aspect missiles, it is unnecessary for the interceptor to fly behind the target to obtain a firing opportunity. The stern conversion is still practiced under different circumstances such as refuelling missions.

2.2.4 Beam Intercept

In order to visually identify the suspected aircraft, a fighter pilot will typically perform a maneuver called a beam intercept [11, 19, 24]. An example of this maneuver is shown in the Figure 2-5, where the interceptor, approaching from the right, is performing the maneuver on the target, on the left. Three regiments characterize this trajectory at the final state: (1) interceptor is a prescribed distance away from the target, (2) interceptor's velocity vector is pointed at the target, (3) the target aspect angle (TAA) is between 90 and 110 degrees [11, 24]. For the simulations presented in this paper, the desired TAA is set to 90°.

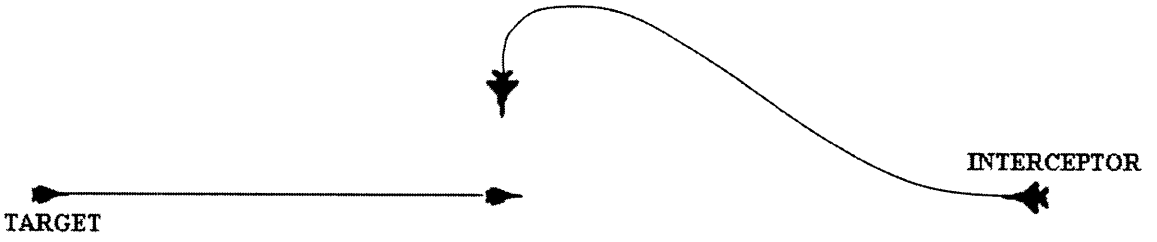


Figure 2-5: The beam intercept usually consists of two turns. The first is a turn away from the target to obtain the desired separation range. The second is a turn towards the target to point the interceptor's velocity vector at the target.

There are several advantages to performing a beam intercept as opposed to using proportional navigation, which might allow a faster merge:

· Decreases the chances of being detected by the target's radar

· Potentially decreases time to VID, because looking at the target from the side rather than straight on will increase the target aircraft's visual surface area

· Increases difficulty for the target to visually identify the interceptor, because during the final turn of the maneuver the interceptor is essentially pointed at the target

· Reduces the closing velocity which will allow more time for the VID, without this excess time the interceptor might have to reposition itself to obtain a firing opportunity

· Ultimately places the interceptor outside of the target's weapons cone

· Places the interceptor in an offensive position, in the event that the target is an enemy

These advantages motivate the decision in this thesis to use the beam intercept as the maneuver for performing a visual identification on a suspected aircraft.

Radar Model

For an aircraft flying over a ground based radar, it is possible to reduce significantly the amount of time during which the aircraft may be detected [23]. The same should be true in performing a beam intercept with a possibility of avoiding radar detection completely. There are instances throughout history where pilots have complained of having a radar lock on an aircraft and then losing it [34]. In some cases, this was the result of the aircraft changing its attitude, which altered the presented radar cross section [34].

The amount of signal energy that an aircraft's radar receives is approximated in [34] as

$$signal\ energy \cong k\frac{P_{avg}G\sigma A_e t}{R^4} \tag{2.2}$$

where $k = \left(\frac{1}{4\pi^2}\right)$, P_{avg} is the transmitter's average power output, G the antenna gain, σ the radar cross section of the interceptor, A_e the effective area of the antenna, t the time the radar is trained on the interceptor, and R the range. Of these parameters,

the interceptor can directly affect σ and R since all of the other parameters are dependent on the target's radar system. This equation is an approximation because the true amount of signal energy depends on the radar system's efficiency in analyzing the data [34]. When integrating the radar signal, other factors such as noise must be taken into account in order to reduce the occurrence of false-positive detections.

In order to avoid radar detection, the interceptor can affect three variables. First, the maximum half cone angle for aircraft radar typically ranges between 45 to 60 degrees. As a result the interceptor can fly a significant portion of the beam intercept outside of the radar cone where radar detection is impossible. The second and third variables which the interceptor can influence are range and radar cross section. Within the cone, the signal energy received by radar is inversely proportional to range R^4 and directly proportional to radar cross section σ

$$signal\ energy \propto \frac{\sigma}{R^4} \tag{2.3}$$

In performing a beam intercept, the interceptor can maintain a large distance between itself and the target until it has flown outside of the target's radar cone. Additionally, the interceptor can affect its radar cross section, which is a function of the interceptor's azimuth κ_r and elevation θ_r angles relative to the target aircraft. These angles are determined with the equations in [23]

$$\hat{x}_b = R_{be}\hat{x}_e \tag{2.4}$$

$$\kappa_r = \arctan(\frac{-\hat{x}_{b,2}}{\hat{x}_{b,1}}) \tag{2.5}$$

$$\theta_r = \arctan(\frac{-\hat{x}_{b,3}}{\sqrt{\hat{x}_{b,1}^2 + \hat{x}_{b,2}^2}}) \tag{2.6}$$

where R_{be} is a rotational matrix which rotates the unit relative position vector between the target and the interceptor $\hat{x}_e$ from an earth frame to a body frame $\hat{x}_b$. $\hat{x}_b$ is a three dimensional vector where $\hat{x}_{b,i}$ represents the i^{th} component of $\hat{x}_b$. R_{be} consists of two rotational matrices [23]

$$R_{be} = L_{bw} L_{we} \tag{2.7}$$

$$L_{we} = \begin{bmatrix} \cos\gamma\cos\Psi & -\cos\gamma\sin\Psi & -\sin\gamma \\ \sin\phi\sin\gamma\cos\Psi + \cos\phi\sin\Psi & -\sin\phi\sin\gamma\sin\Psi + \cos\phi\cos\Psi & \sin\phi\cos\gamma \\ \cos\phi\sin\gamma\cos\Psi - \sin\phi\sin\Psi & -\cos\phi\sin\gamma\sin\Psi - \sin\phi\cos\Psi & \cos\phi\cos\gamma \end{bmatrix} \tag{2.8}$$

$$L_{bw} = \begin{bmatrix} \cos\alpha\cos\beta & \cos\alpha\sin\beta & -\sin\alpha \\ -\sin\beta & \cos\beta & 0 \\ \sin\alpha\cos\beta & \sin\alpha\sin\beta & \cos\alpha \end{bmatrix} \tag{2.9}$$

The first rotation L_{we} is from the earth to the wind axis system, and the second rotation L_{bw} is from the wind to the body axis system [23]. The angles in the rotation matrices represent the interceptor's attitude where γ is flight path angle, ψ heading, ϕ bank angle, α angle of attack, and β sideslip. The body fixed coordinate system is defined with positive x axis out the interceptor's nose, positive y axis out the interceptor's left wing, and positive z axis follows the right hand rule [23].

For the direct collocation development described in chapter 5, a generic aircraft radar cross section model is used similar to the one found in [23], where a bivariate cubic spline is used to map the interceptor's azimuth and elevation angles to σ, as shown in Figure 2-6.

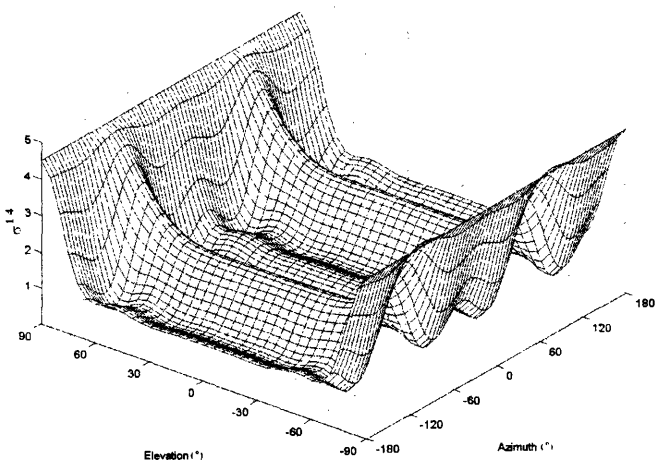


Figure 2-6: Generic aircraft radar cross section model [23]

The presented radar cross section can vary greatly even with small aircraft vibrations [34] so that the smooth representation shown in Figure 2-6 may be inaccurate. A radar cross section model for an actual aircraft must include these variations to ensure that avoiding radar detection is still possible.

In this research, it is assumed that the interceptor has perfect knowledge of the state of the target. This information could be provided by the aircraft's own radar or an AWACS.

[This page intentionally left blank.]

Chapter 3

Aircraft Trajectory Optimization

There has been a great deal of research done on aircraft trajectory optimization over the past 35 years. Initially, the calculus of variations was used to produce optimal solutions in analytic form, which are solvable in real time. However, solutions of this form exist for very few problems, and the models describing the aircraft dynamics need to be simplified, to admit analytic, closed form solutions.

If an analytic solution is not found, the problem may be solved using a shooting method approach. An indirect shooting method also uses the calculus of variations to cast the problem, whereas the direct shooting method does not. For both of these methods, the iteration process in finding an optimal solution involves integrating the trajectory forwards or backwards in time from the known initial states to the desired final states, which serve as boundary conditions to the solver.

A promising method for trajectory optimization is mixed integer linear programming (MILP). For this approach, the aircraft dynamics and constraints are modelled in linear form, and a globally optimal solution can be found for many scenarios in real time.

In order to avoid integrating the trajectories as required with shooting methods, the direct collocation approach can be used. With this method the problem is discretized, and the dynamics of the system are approximated with a set of algebraic constraints. The problem is then cast and solved as a nonlinear program.

The final method considered is trajectory interpolation. This new approach in-

volves interpolating between two similar trajectories to obtain a family of maneuvers.

3.1 Analytic Solutions

Analytic solutions for optimal control problems are typically derived using the calculus of variations, which forms the Euler Lagrange equations [6]. An optimal solution must satisfy these equations, which occur in the form of a two point boundary value problem (TPBVP) where some boundary conditions are given at the initial time and others at the final time. In very few instances an optimal solution can be found in closed form. Solutions of this type for aircraft trajectory problems typically involve flight constrained to the horizontal plane. Research in this area mainly consists of flight from point A to point B in minimum time or flight from one heading to another.

In 1971, Erzberger and Lee published a paper describing methods for producing three types of trajectories in the horizontal plane for a constant speed aircraft: (1) point A to point B with designated final heading, (2) point A to a line with final heading along the line, (3) point A to point B with unfixed final heading [14]. The solution for each of these problems occurs in bang-off-bang form, where the aircraft is either flying along a straight path or turning at maximum bank angle. Rather than solving a TPBVP, various solutions are produced geometrically where the optimal solution is the one with the shortest path length. The authors found that an optimal trajectory consists of no more than four turns.

Clements produced in 1990 an analytic solution for the problem of an aircraft's flight in the presence of winds from point A to point B with final heading unfixed [8]. The aircraft flies at a constant altitude and constant indicated airspeed. Using geometry consisting of turning circles and the solution to a fixed-point equation, the algorithm is able to produce solutions in real time. To follow this paper, Clements also solved a similar problem in 1992, including a limit on maximum roll rate [9].

The problem of producing trajectories for a varying speed aircraft from point A to point B with a fixed final heading has also been addressed. In 1994, Ben-Asher solved this scenario using a multiple shooting method [2]. The author notes the challenges of

onboard implementation due to the difficulty in producing reasonable values for the costates of the TPBVP. To follow this work, in 1997 Shapira and Ben-Asher produced analytic solutions to a similar problem, limiting the solution space to bang-off-bang trajectories and posing the speed of the aircraft as a linear function of the aircraft's turn rate [31]. Although this method may not produce the true optimal solution, a solution is guaranteed in real time.

Analytic solutions are also available for the problem of changing the aircraft's heading to a specified value and finishing the turn with a desired speed [15, 35]. In 1998, Grimm and Hans showed that their method for various scenarios differs only slightly from the optimal solution obtained using a direct multiple shooting approach [15]. However, in order to maintain an analytic solution, a constraint on load factor was not included.

3.2 Shooting Methods

Shooting methods are commonly used to solve TPBVP's, and consist of three categories: direct shooting, indirect shooting, and multiple shooting [3].

For direct shooting, the method involves solving for a subset of the initial boundary values, final boundary values, and parameters which create an optimal trajectory. The parameters are a set of coefficients for the controls, which must be written in explicit or implicit form. For each iteration, the method integrates the differential equations of the system forward or backward, and then updates the variables based on the resulting objective function value and the errors in the boundary conditions. This method works especially well when the number of unknown variables is small. However, a major shortcoming is that integrating small errors in the initial values can lead to enormous errors at the endstate [3].

With the indirect shooting method, one must define the necessary conditions for optimality in the form of Euler Lagrange equations. The set of values to be solved are the costate variables at the initial time and the time to complete the trajectory. Often, we assume a form for the propagation of the costate, and the

terminal state constraints are mapped onto the costate. The costate is then solved while the states are propagated forward to find the optimal solution. Typically, it is hard to provide a good initial guess for the costate variables due to their lack of intuitiveness. Additionally, a small error in these values can lead to highly erratic values at the endstate [3].

In order to address the instability found in the two previous methods, a multiple shooting approach was introduced. The problem is discretized into multiple time steps, so that small initial errors only translate from one time step to the next. Additional constraints are added in order to ensure continuity of the states at each of the nodes. The problem is then solved as many TPBVP's using either the indirect or direct approach [3].

With all of the shooting methods, a major difficulty is incorporating inequality constraints that are a function of the states along the trajectory [3]. The portion where an inequality constraint is active is referred to as a constrained arc. In order to solve these types of problems, the number and location of constrained arcs must be defined beforehand [6].

Shooting methods have been applied to many different aircraft trajectory problems. Seywald, Cliff, and Well in 1991 used the multiple shooting method for finding optimal trajectories for an aircraft flying in the vertical plane where the objective is to maximize range [30]. The problem is constrained with maximum thrust, load factor, and dynamic pressure limits where boundary conditions specify the initial and final specific energy, altitude, and flight path angle. The dynamic pressure limit adds constrained and unconstrained arcs to the problem. The authors are able to produce optimal trajectories for various scenarios, but they mention that the problem has not been solved for "long flight times", due to the difficulty in finding an appropriate switching structure for the costate variables.

In 1993, Bocvarov, Lutze, and Cliff used shooting methods to determine the effect of adding thrust vectoring to a high fidelity F/A-18 model in performing certain maneuvers [4]. The authors note difficultly in finding a good initial guess for the costate variables, the time to complete the maneuver, and the location of the con-

strained arcs. In order to address this problem, the authors create optimal trajectories for simpler maneuvers. Then, the boundary conditions or parameters, such as the amount of thrust vectoring power, are varied slightly towards the desired values, using the previous solution as an initial guess for the costate variables. This homotopy method allows an efficient means to calculating trajectories for a more complex set of maneuvers [3].

3.3 Mixed Integer Linear Programming

MILP is an optimization-based approach which can be used to solve trajectory generation problems in real time. The problem is discretized with N nodes, where linear equations describe the vehicle's state at each node. Constraints on the vehicle's states and controls can be added as long as they occur in linear form, and adjoint binary variables are used in order to impose or relax relevant constraints.

Recent research has shown MILP to be a promising method for trajectory optimization. The first paper using this approach, published in 2001, outlines the algorithm for moving multiple vehicles from an initial state to a desired state while minimizing time or fuel burn [29]. The method ensures that the vehicles avoid collisions with each other as well as stationary or moving obstacles. Both a fixed final time and a receding horizon control approach, which reduces the computation time for creating paths, were tested.

In 2002, Richards and How applied the previous research to a specific vehicle model, which is an aircraft with a maximum speed and turn rate [25]. The aircraft is also constrained to a constant altitude. In this paper, the authors present solutions for path planning problems where multiple aircraft are required to visit waypoints while avoiding obstacles.

In order to reduce the computation time required for producing aircraft trajectories with MILP, a new receding horizon control approach was introduced in 2002 [1]. Trajectories are calculated using a combination of a modified version of Dijkstra's algorithm in addition to MILP. This method has the advantage of avoiding unescapable

obstacles, and it is also a candidate for producing solutions in real time with only a small reduction in optimality.

Recently in 2003, Richards, Kuwata, and How successfully demonstrated that MILP can be used for real-time path planning [26]. The authors present two experiments which involve small ground vehicles about the size of a remote control car. The first uses receding horizon control to guide a vehicle around an obstacle to a desired final state. The second is a rendezvous problem, where one vehicle travels along a straight path, and the other maneuvers to reach the desired relative position.

3.4 Direct Collocation

With the direct collocation method for trajectory optimization, the states are approximated with cubic polynomials, and the dynamics are discretized with N nodes using collocation of the states and controls [16]. The problem is formulated as a nonlinear program where the dynamics are approximated with algebraic constraints.

The seminal paper describing this approach was published by Hargraves and Paris in 1987 [16]. They demonstrated the method with three numerical examples, one of which is the fighter aircraft minimum time to climb problem. Their optimal trajectory for this maneuver closely matched the results previously obtained using shooting methods [7].

In 1999, Ringertz demonstrated the performance of a direct collocation solution with live flight testing [27]. A pilot flying a Saab J35 Draken administered the optimal commands to accelerate the aircraft to a specified airspeed; the test results were very similar to the computed solution.

To follow Ringertz's previous work, in 2000 he produced trajectory solutions for a minimum fuel turn [28]. The aircraft dynamics are approximated as a reduced-order model of the full six degrees of freedom model which assumes an inertial reference frame, a constant aircraft mass, and a rigid body. The optimal trajectory for this maneuver is an out of horizontal plane maneuver where the aircraft varies its altitude in order to minimize the amount of fuel burned. The author discusses the difficulty

in obtaining solutions in real time. He states that the method's performance can be improved through solving the problem with a small number of nodes and then using this solution as an initial guess for solving subsequent problems with additional nodes.

Horie and Conway also used the direct collocation technique in 2000 to design trajectories for an evader aircraft to reposition itself behind a pursuit aircraft [18]. This tactic consists of a vertical maneuver using post-stall flight and thrust vectoring to slow the evader's velocity and allow the pursuer to pass below. Using a reduced-order aircraft model similar to an F-16's dynamics, the authors are able to produce solutions with various boundary conditions.

In 2003, Norsell showed that it is possible to significantly reduce the amount of time that an aircraft is detected while flying over ground-based radar. Using the same dynamic model as Ringertz in [28], the author produces optimal trajectories with the goal of decreasing detection time and minimizing fuel burn. A user selected gain varies the weight on the importance of these two objectives. For this type of trajectory, a real time solution is not essential due to the fixed position of the ground radar; a trajectory can be calculated offline prior to the actual flight.

3.5 Trajectory Interpolation

In 2004, Dever, Mettler, Feron, Popović, and McConley introduced a new approach for creating trajectories [12]. The method uses sets of similar maneuvers, which have different boundary conditions, and interpolates between them to obtain a set of additional trajectories. The new paths maintain a similar analytic structure between desired boundary conditions; a desired trajectory with specified initial conditions and endstates can be obtained using this process. For example, a desired trajectory for a helicopter is to transition from an initial forward velocity to a hover state. Two trajectories with different initial forward velocities could be produced offline; online the method interpolates between them to obtain a feasible maneuver for the given initial forward velocity. This method has been successfully tested on a three degree

of freedom rotorcraft. Trajectory interpolation potentially has the major advantage of being able to create complex maneuvers in real-time.

3.6 Method Selection

Three of the five approaches listed above are tested in this thesis. It is unlikely that an analytic solution exists for creating beam intercept trajectories in which the interceptor avoids radar detection. Also, due to the inherent difficulties of including constrained arcs, a shooting method approach is not included.

The methods of MILP, direct collocation, and trajectory interpolation are applied in this thesis. MILP has the advantage that the amount of time to converge to a solution can be estimated. A shortcoming of this approach is the difficulty in formulating a complex aircraft model and putting constraints, such as radar detection avoidance, into a linear form.

The direct collocation method has the advantage of ease of use. The problem is simply formulated as a nonlinear program and then solved with optimization software. However, a major disadvantage is the existence of multiple local minima. The method can converge to a solution which may be an extremely inefficient trajectory. These convergence problems make an onboard implementation difficult.

The trajectory interpolation approach is included because it has the potential to produce trajectories in real time. Although the method does not maintain optimality, the solutions may be sufficiently close. For onboard implementation, a nearly optimal beam intercept delivered within seconds is certainly better than the true optimal trajectory produced a day later.

Chapter 4

MILP Approach

In this chapter, we investigate the beam intercept problem defined in Section 2.2.4 using mixed integer linear programming (MILP). The approach closely follows the method presented in [25] with variations in the cost function and constraints to reduce the likelihood of the target detecting the interceptor. While the rendezvous problem of [26] models the interceptor dynamics in a reference frame relative to the target, the dynamics for this approach include only the interceptor's states. The target aircraft's position and velocity are projected to produce its baseline trajectory.

4.1 Radar Detection Avoidance Included in the Cost Function

Let $f_{x,k}$ and $f_{y,k}$ be the interceptor's applied forces in the inertial x and y directions at node k. Solving for the controls $f_{x,k}$ and $f_{y,k}$ produces the trajectory according to the following dynamics and constraints. Using a set of N nodes, the dynamic model of the aircraft is discretized with the linear equations

$$\ddot{x}_{int,k} = \frac{f_{x,k}}{mass} \tag{4.1}$$

$$\ddot{y}_{int,k} = \frac{f_{y,k}}{mass} \tag{4.2}$$

$$\forall \quad k \in [1 \ldots N-1] \tag{4.3}$$

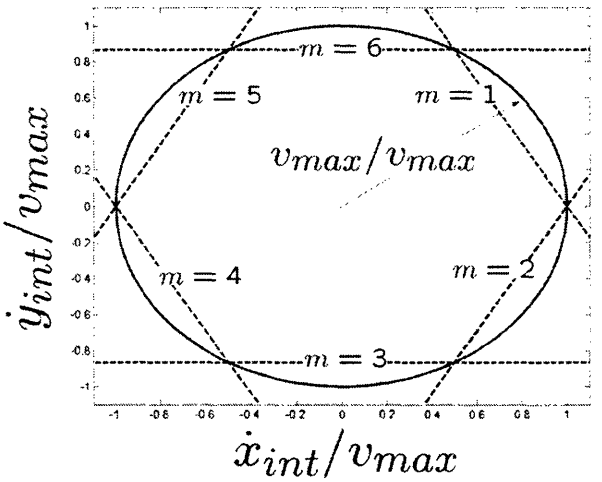


Figure 4-1: Linear constraints are used to approximate the maximum speed limit v_{max}. The lines $m = 1 \ldots 6$ represent each of the linear inequalities.

where $(x_{int,k}, y_{int,k})$ are the interceptor's x and y position in the inertial frame. Similarly $(\dot{x}_{int,k}, \dot{y}_{int,k})$ and $(\ddot{x}_{int,k}, \ddot{y}_{int,k})$ are the interceptor's velocities and accelerations in the inertial frame. $\Delta t(k)$ is the user specified time between each node.

The maximum allowable speed v_{max} and force f_{max} at each node are approximated with the linear constraints

$$\dot{x}_{int,k} \sin \frac{2\pi m}{M} + \dot{y}_{int,k} \cos \frac{2\pi m}{M} \leq v_{max} \cos \frac{\pi}{M} \tag{4.4}$$

$$\forall \quad k \in [1 \ldots N], m \in [1 \ldots M] \tag{4.5}$$

$$f_{x,k} \sin \frac{2\pi m}{M} + f_{y,k} \cos \frac{2\pi m}{M} \leq f_{max} \cos \frac{\pi}{M} \tag{4.6}$$

$$\forall \quad k \in [1 \ldots N-1], m \in [1 \ldots M] \tag{4.7}$$

where M is the number of linear constraints used to approximate the 2-norm on speed and thrust. With larger values of M, the set of constraints approaches the actual magnitude constraints on velocity and force vectors, as shown in Figure 4-1.

In the MILP approach, the operation convenes by first synthesizing an entire set of candidate terminal interceptor states at each point in time termed $(x_{des,k}, y_{des,k})$, $(\dot{x}_{des,k}, \dot{y}_{des,k})$, which are the interceptor's position and velocity relative to the tar-

get's estimated position and velocity at node k. Not all of the sets $(x_{des,k}, y_{des,k})$, $(\dot{x}_{des,k}, \dot{y}_{des,k})$ are reachable or admissible. The search process involves propagating the interceptor dynamics such that they eventually approach or intersect this time history of terminal points $(x_{des,k}, y_{des,k})$, $(\dot{x}_{des,k}, \dot{y}_{des,k})$. Any intersection thus produces a feasible engagement. The least expensive of all these feasible trajectories that intersect the set of candidate terminal states is output as the trajectory.

The interceptor's desired endstate changes in order to include a maneuvering target, whose position and velocity at each node k are specified before the MILP step

$$x_{des,k} = x_{tgt,k} + R_s \cos(\Psi_{tgt,k} \pm 90°) \tag{4.8}$$

$$y_{des,k} = y_{tgt,k} + R_s \sin(\Psi_{tgt,k} \pm 90°) \tag{4.9}$$

$$\dot{x}_{des,k} = \frac{\dot{x}_{tgt,k}\cos(\mp 90°) - \dot{y}_{tgt,k}\sin(\mp 90°)}{\sqrt{\dot{x}_{tgt,k}^2 + \dot{y}_{tgt,k}^2}} v_s \tag{4.10}$$

$$\dot{y}_{des,k} = \frac{\dot{x}_{tgt,k}\sin(\mp 90°) + \dot{y}_{tgt,k}\cos(\mp 90°)}{\sqrt{\dot{x}_{tgt,k}^2 + \dot{y}_{tgt,k}^2}} v_s \tag{4.11}$$

$$\forall \quad k \in [1 \ldots N] \tag{4.12}$$

where v_s is the final desired interceptor speed. (x_{tgt}, y_{tgt}) and $(\dot{x}_{tgt}, \dot{y}_{tgt})$ are the target's x and y position and x and y velocity respectively. $\Psi_{tgt,k}$ is the target's heading and R_s is the desired range between the target and the interceptor at the end of the intercept. The interceptor's preferred terminal state to make a positive identification is $\pm 90°$ to the target orientation, which is shown in Figure 4-2. Additional binary variables or a heuristic can be used to select which desired final heading and position produces the best solution.

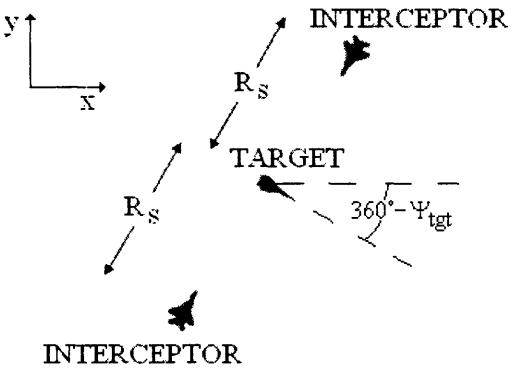


Figure 4-2: Potential endstates for beam intercept

The following binary constraints model the completion of the beam intercept

$$X_{int} - X_{des} \leq Hw_k \tag{4.13}$$

$$-X_{int} + X_{des} \leq Hw_k \tag{4.14}$$

$$\forall \quad k \in [1 \ldots N] \tag{4.15}$$

where X_{int} represents the set of variables $(x_{int,k}, y_{int,k})$, $(\dot{x}_{int,k}, \dot{y}_{int,k})$ and X_{des} represents $(x_{des,k}, y_{des,k})$, $(\dot{x}_{des,k}, \dot{y}_{des,k})$. H is a large positive number, and w_k is a binary variable which allows the constraints to be relaxed. This formulation admits solutions that drive the interceptor to intersect the candidate set of desired interceptor states produced earlier. If the interceptor has reached the candidate relative position and velocity at node k, then $w_k = 0$ which is shown in Figure 4-3. This relaxation of the constraints is represented in the objective function as the variable t_k. At node k if $w_k = 0$, then $t_{k \ldots N} = 0$ otherwise $t_k = 1$. In order to enforce that the maneuver is completed at some node k, an additional constraint is added

$$\sum_{k=1}^{N} t_k \leq N - 1 \tag{4.16}$$

The optimization objective is to minimize the time to complete the beam intercept and the accelerations over the course of the mission. In order to approximate the

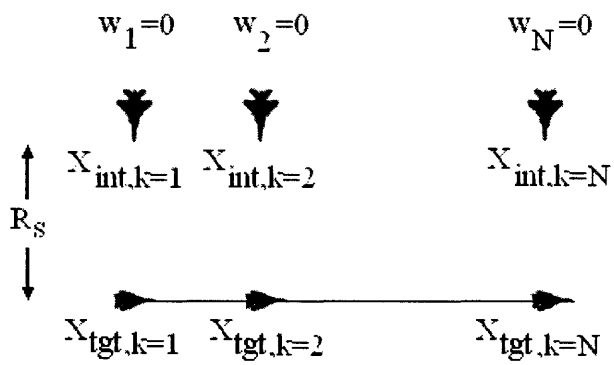


Figure 4-3: This figure shows the candidate endstates of the interceptor at each node k for a target with a constant heading. $X_{int,k}$ and $X_{tgt,k}$ contain the states of the interceptor and target respectively, and R_s is the desired separation distance. If there is a feasible solution at an endstate, the binary variable w_k can be set to zero, otherwise it is set to one in order to relax the desired endstate constraint. MILP software solves multiple linear programming problems where the binary variables are varied in order to find a feasible, optimal solution.

objective of maximizing the range between the two aircraft within the linear cost formulation structure that MILP permits, the term $c|y_k|$ is introduced

$$\min_{f_{x,k},f_{y,k}} J = \sum_{i=1}^{N}(\epsilon|f_{x,k}| + \epsilon|f_{y,k}| + t_k - c|y_k|) \tag{4.17}$$

Through $c|y_k|$, the interceptor is encouraged to maximize its cross-range relative to the target throughout the trajectory. This formulation enables a penalty for proximity with the target. The target's initial position is placed at the origin with its velocity vector pointed along the x-axis, so that with larger values of c the interceptor will maintain higher values of range throughout the trajectory. This could potentially decrease the chances of being detected by the target's radar because the amount of power that the aircraft's radar receives is heavily dependent on range.

In Section 4.3, the problem is solved once, given the initial states of the interceptor and the target's projected trajectory. However, the problem could be resolved during the maneuver if the target deviates significantly from its previously calculated path.

4.2 Additional State Constraint for Radar Detection Avoidance

Another method for avoiding radar detection involves breaking up the trajectory planning problem into two segments. The initial portion entails that the interceptor maintain at least a certain range with respect to the target until it crosses the target's radar cone. We mathematically formulate this requirement by having the interceptor fly to a point at least a certain distance RR along the target's radar cone, as shown in Figure 4-4. If the value for RR is sufficiently large given the radar cross section of the interceptor and the capability of the target's radar, then it is reasonable to assume that the interceptor will not be detected by the target's radar. Beyond this point, the solution for the intercept problem is formulated independent of range. This method has the advantage over the previous approach in that the interceptor only maintains distance from the target while within the target's radar cone. Also, it is easier to chose a value of RR than the coefficient c in order to avoid proximity with the target.

This method uses the equations as previously described but with an additional requirement that the interceptor must fly to two waypoints. The $c|y_k|$ term is also removed from the cost function. The first point is a distance of at least a certain range RR along the target's radar cone

$$
\begin{aligned}
(x_{int,k} - x_{tgt,k}) - RR\cos(RCA + \Psi_{tgt,k}) &\leq Hv_k && (4.18)\\
-(x_{int,k} - x_{tgt,k}) + RR\cos(RCA + \Psi_{tgt,k}) &\leq Hv_k && (4.19)\\
-(y_{int,k} - y_{tgt,k}) + RR\sin(RCA + \Psi_{tgt,k}) &\leq Hv_k && (4.20)\\
\forall \quad k \in [1 \ldots N] && && (4.21)
\end{aligned}
$$

where RCA is the radar cone angle limit and v_k is a binary variable. Now, $t_{k\ldots N} = 0$ if both v and w are zero at previous time steps. The second point is the previously defined completion state of the maneuver.

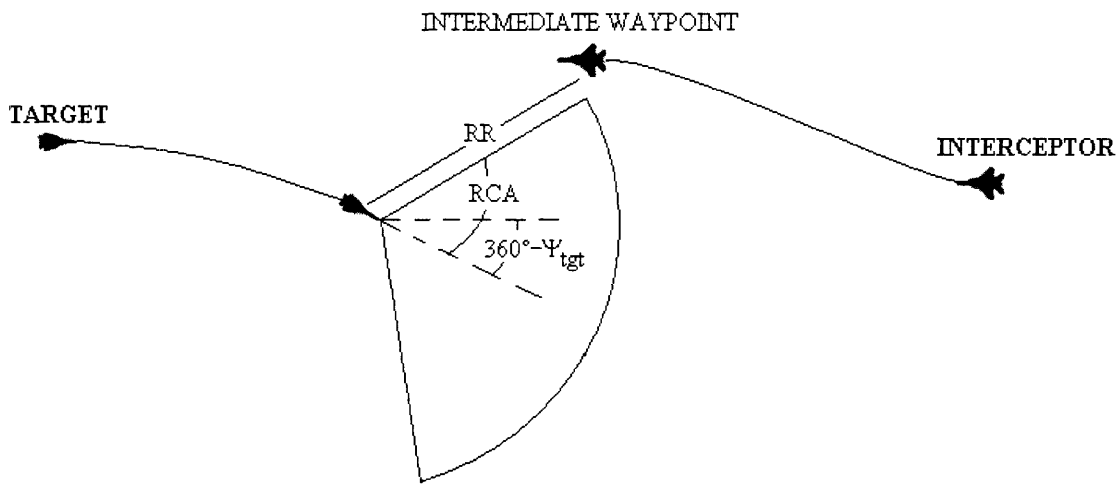


Figure 4-4: In order to reduce the chances of being detected by the target's radar, the MILP problem is formulated with an additional constraint that the interceptor is required to fly to an intermediate waypoint of at least a distance RR along the radar cone. RCA indicates the maximum angle at which the target's radar will function and Ψ_{tgt} is the target's heading.

4.2.1 MILP Software and Limitation of MILP Approach

Xpress software [36] was chosen for the following simulations because it was available at Draper Laboratory. It can be used to solve linear, quadratic and integer programming problems. CPLEX, another MILP optimization software, could also be used for this approach.

A limit on minimum speed is not included in the problem formulation. This can result in infeasible trajectories, where the solution produced by the optimizer may exceed a maximum turn rate or the aircraft might stall. A minimum speed constraint with the same accuracy as the maximum speed approximation would require $N \times M$ additional binary variables. This slowed the run time of the MILP optimization software Xpress [36] significantly. Otherwise, the turn rate constraint can also be satisfied by decreasing f_{max} [25].

4.3 MILP results

For the following the simulations, initially the target and interceptor are heading towards each other and are separated by $27.5km$. The two aircraft have the same initial speed, and the target is turning to the right with a constant turn rate. The constants for the simulations are shown in Table 4.1. Two different methods using MILP are applied to solve the beam intercept problem. The first involves using an objective function that includes time to completion and range from the target throughout the maneuver. The second MILP approach adds an additional requirement that the interceptor must maintain a certain range from the target until it crosses the radar cone. The objective is to complete this task and finish the beam intercept in minimum time.

Table 4.1: Constants for MILP beam intercept simulations

N	60	M	10
$mass$	$10,442kg$	H	$100,000$
v_{max}	$167m/s$	f_{max}	$313kN$
$x_{int.k=1}$	$27.5km$	$y_{int,k=1}$	$0km$
$x_{tgt,k=1}$	$0km$	$y_{tgt,k=1}$	$0km$
$\dot{x}_{int,k=1}$	$-152.5m/s$	$\dot{y}_{int,k=1}$	$0m/s$
$\dot{x}_{tgt,k=1}$	$152.5m/s$	$\dot{y}_{tgt.k=1}$	$0m/s$
v_s	$152.5m/s$	R_s	$3.05km$
ϵ	1×10^{-9}	$\Delta t(k)$	$2s$

4.3.1 MILP Results with Adjusted Cost Function

As presented in Section 4.1, the cost function for the beam intercept scenario is

$$\min_{f_{x,k}, f_{y,k}} J = \sum_{i=1}^{N} (\epsilon |f_{x,k}| + \epsilon |f_{y,k}| + t_k - c|y_k|) \tag{4.22}$$

where the $c|y_k|$ term encourages the interceptor to maintain a standoff range from the target vehicle.

The first results show solutions to the problem with variations on the weighting factor c of $c|y_k|$. Using a 1.50 GHz desktop with 1.00 GB of RAM, Xpress software

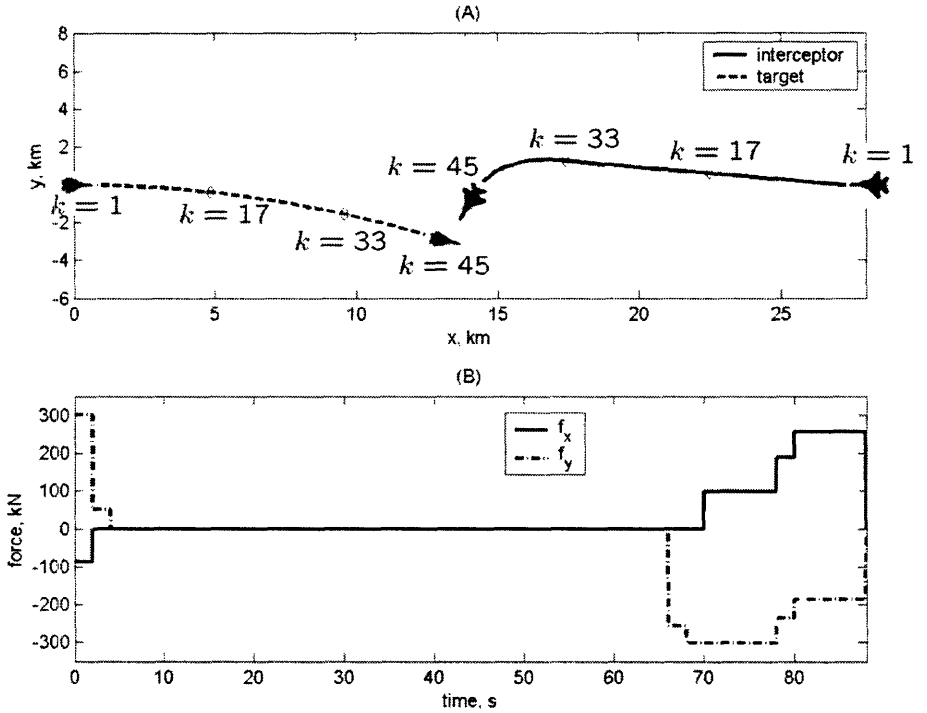


Figure 4-5: Beam intercept using MILP with y weighting coefficient $c = 0$. Figure (A) shows the interceptor and target trajectory, and Figure (B) shows the interceptor's controls during the maneuver. The target is turning at a constant rate of $0.3°/s$. A large control control action is required at the beginning of the trajectory in order to change the heading of the interceptor. For the majority of the maneuver, the interceptor maintains this heading, and then finally a large control effort is applied in order to obtain the desired relative heading and position.

[36] is typically able to solve this problem in 2-4 seconds.

Figures 4-5 and 4-6 show two examples of a beam intercept and the corresponding control histories. With the largest value of the y weighting coefficient, the maneuver requires significantly more control effort and the time to completion is 30 seconds longer than the shortest trajectory, as shown in Table 4.2. In both scenarios, at the end of the maneuver the interceptor has a $3.05km$ separation distance from the target and a terminal heading 90° different than target's.

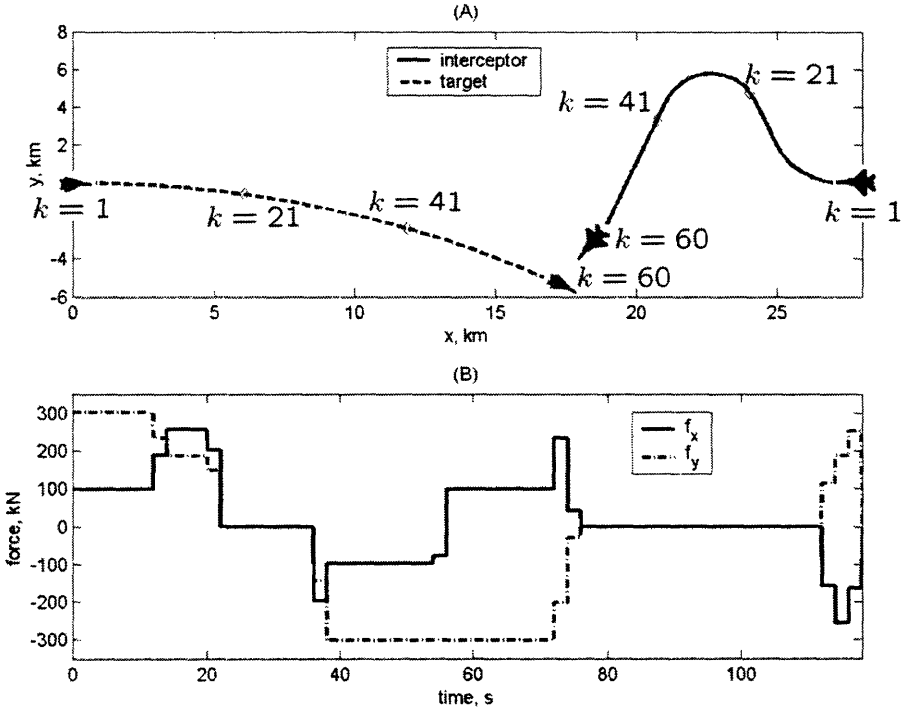


Figure 4-6: Beam intercept using MILP with y weighting coefficient $c = 0.1$ which encourages the interceptor to increase its cross-range from the target. Figure (A) shows the interceptor and target trajectory, and Figure (B) shows the interceptor's controls during the maneuver. The target is turning at a constant rate of $0.3°/s$. Initially, the interceptor makes a sharp turn to the right in order to obtain a larger distance from the target. Then, the interceptor briefly travels along a straight path and banks left to head towards the target. After following along another straight path, a small turn is applied to complete the maneuver.

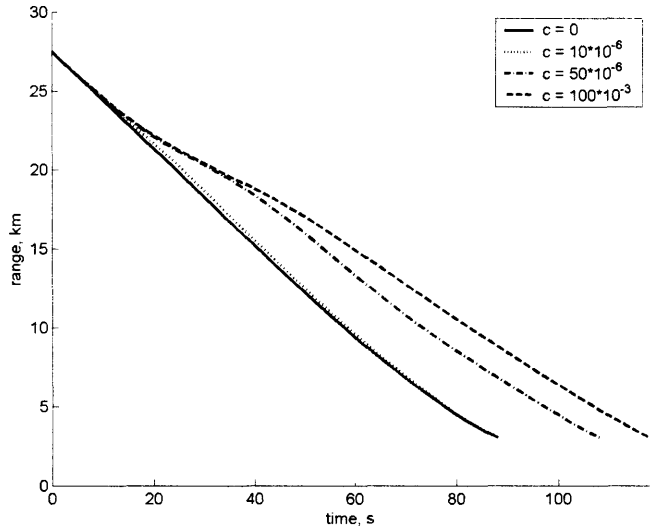


Figure 4-7: With higher values of the constant c, the interceptor maintains a larger distance from the target throughout the trajectory.

Figure 4-7 shows how the $c|y_k|$ term influences the range between the interceptor and target during the beam intercept. The four curves are from the trajectories in Table 4.2.

A range of trajectories can be created by varying c. However, there is no guarantee that the interceptor will avoid radar detection. After the trajectory is created, one could include a separate evaluation to determine if the interceptor would be detected by the target's radar. If the trajectory fails the test, the algorithm could run iteratively with larger values of c.

Table 4.2: Results from MILP beam intercept trajectories including $c|y_k|$ term in cost function

	path length (km)	trajectory time (s)	computation time (s)	active f_x,f_y duration (s)	min/max speed (m/s)
$c = 0$	14.2	88	2.9	26	150/159
$c = 10 \cdot 10^{-6}$	14.7	88	4.7	48	153/167
$c = 20 \cdot 10^{-6}$	15.0	90	5.2	50	153/167
$c = 50 \cdot 10^{-6}$	17.8	108	5.3	74	150/167
$c = 100 \cdot 10^{-3}$	18.2	118	3.4	68	106/167

4.3.2 MILP Results with Additional State Constraint for Radar Detection Avoidance

For this formulation, the interceptor is required to fly to a point of at least a certain distance RR along the radar cone, where the radar cone angle is limited to 45°. The cost function for this scenario is

$$\min_{f_{x,k},f_{y,k}} J = \sum_{i=1}^{N} (\epsilon|f_{x,k}| + \epsilon|f_{y,k}| + t_k) \tag{4.23}$$

so that now the interceptor must fly to both waypoints in minimum time with a small weight on control effort. With the additional binary variables required for this approach, the computation times are significantly longer as seen by comparing the results in Table 4.3 to the results without the constraint in Table 4.2.

Table 4.3: Results from MILP beam intercept trajectories including a constraint to fly to point of at least a distance RR along the radar cone edge

	path length (km)	trajectory time (s)	computation time (s)	active f_x,f_y duration (s)	min/max speed (m/s)
$RR = 9.2km$	14.5	90	10.1	32	151/163
$RR = 12.2km$	16.2	102	29.9	48	124/165

In Figures 4-8 and 4-9, the interceptor flies to a point of exactly the required distance along the radar cone and finishes the trajectory with the desired relative heading of 90° and range of $3.05km$.

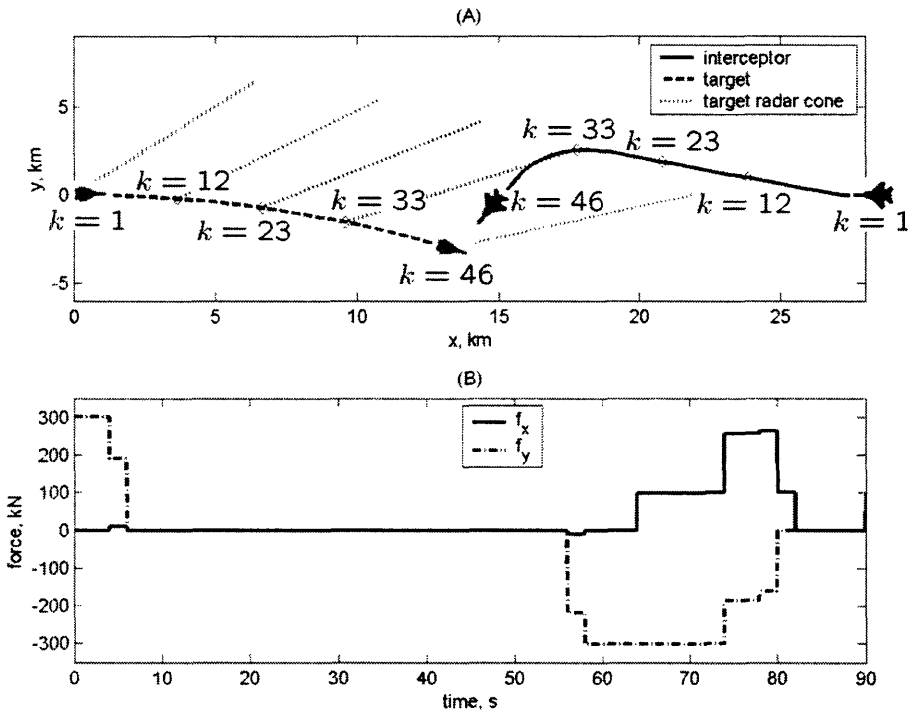


Figure 4-8: Beam intercept using MILP where interceptor is required to fly to a distance of at least $9.2km$ along the target's radar cone. Figure (A) shows the aircrafts' trajectories along with the radar cone limit of the target, and Figure (B) is the interceptor's control history. The target is turning at a constant rate of $0.3°/s$. The interceptor begins the maneuver with a right turn in order reach the end of the target's radar cone. Shortly before reaching this point, a hard left turn is executed to wrap around the radar cone and to point the interceptor towards the desired endstate.

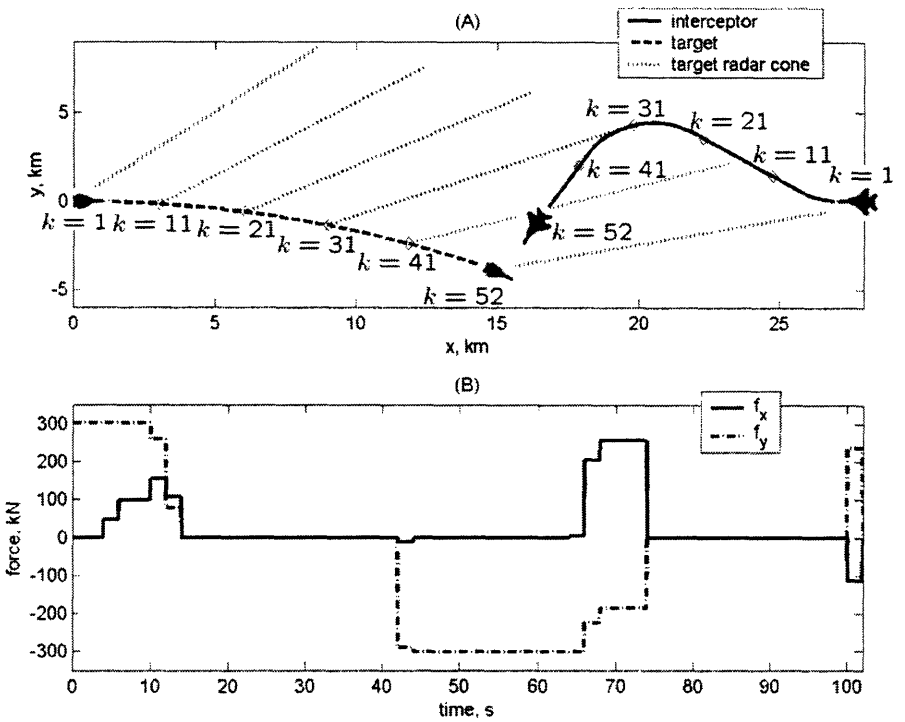


Figure 4-9: Beam intercept using MILP where interceptor is required to fly to a distance of at least $12.2km$ along the target's radar cone. Figure (A) shows the aircrafts' trajectories along with the radar cone limit of the target, and Figure (B) is the interceptor's control history. The target is turning at a constant rate of $0.3°/s$. The interceptor performs the maneuver in a similar manner as Figure 4-8 but with exaggerated turns in order to fly around the stronger radar.

4.3.3 Discussion of Results

As stated in Section 2.2.4, the interceptor can avoid radar detection from the target by flying outside of the target's radar cone, maintaining larger distances from the target, and controlling the radar cross section it presents to the target. Both MILP approaches are able to handle the first two criteria. However, they are limited because it is difficult to include the radar cross section of the interceptor without using large amounts of binary variables. If the interceptor's maximum radar cross section is sufficiently small, this value could serve as a constant in setting c and RR. With larger values of radar cross section, a method which directly accounts for this variable could be more effective.

[This page intentionally left blank.]

Chapter 5

Direct Collocation Approach

While the MILP approach can create feasible trajectories for a beam intercept in real time, it is difficult to formulate a linear constraint that models the radar cross section of an aircraft. The following direct collocation approach can address this issue, although the nonlinear programming formulation might result in longer computation times without a guarantee of converging to a feasible solution.

5.1 Description of Direct Collocation Method

With direct collocation the optimal control problem is cast as a nonlinear program (NLP) with constraints. As shown in Figure 5-1, the problem is discretized with $N+1$ nodes occurring at discrete points in time, $(t_0, t_1, t_2, ..., t_N)$, where t_N is the final time t_f.

The NLP problem is written in terms of $(N+1)*m$ state variables consisting of $x = [x_0', x_1', x_2', \ldots, x_N']$ where x_k is an m-element state vector defining the state at node k. Additionally, there are $(N+1)*n$ variables for the controls $u = [u_0', u_1', u_2', \ldots, u_N']$ where u_k is an n-element column vector of controls at node k. There is also a variable for final time which is denoted as t_f.

The objective function is a scalar in the following form

$$J = \omega(x, u, t_f) \tag{5.1}$$

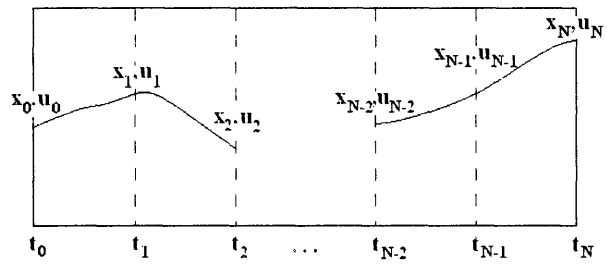


Figure 5-1: Direct collocation involves discretizing the trajectory in order that the states and controls are defined at specific nodes. The nodes are not constrained to equal spacing in time.

where ω is a function of the states, controls, and final time and is subject to the following constraints of system dynamics, boundary conditions, path constraints, and state and control limits.

The dynamic equations describing the system are a function of the states and controls

$$\dot{x} = f(x, u) \tag{5.2}$$

There are also constraints describing the boundary values at the initial and final time

$$b_{l0} \leq b(x_0, u_0, t_0) \leq b_{u0} \tag{5.3}$$

$$b_{lf} \leq b(x_N, u_N, t_N) \leq b_{uf} \tag{5.4}$$

Additionally, the solution is constrained by path constraints

$$g_l \leq g(x, u, t) \leq g_u \tag{5.5}$$

Finally, there are limits on the states and controls

$$x_l \leq x \leq x_u \tag{5.6}$$

$$u_l \leq u \leq u_u \tag{5.7}$$

The boundary conditions, path constraints, and limits on the states and controls are

addressed through simply entering equations (5.3-5.7) directly as they appear into the NLP. These constraints are only enforced at the nodes, and the trajectory must be checked afterwards to ensure feasibility. If the trajectory is found to violate a path constraint, additional nodes can be added to fix the problem.

5.1.1 Collocation

In order to address the dynamics of the system, the method of collocation is used to approximate the equations of (5.2) with equality constraints [16]. The first step in forming these constraints is to represent each of the states as piecewise cubic polynomials between the nodes in the form

$$x = C_0 + C_1 S + C_2 S^2 + C_3 S^3 \tag{5.8}$$

where S is an interpolation variable such that $S = 0$ at t_k and $S = 1$ at t_{k+1}. Equation 5.8 and its derivative are evaluated at $S = 0$, $S = 1$, and $S = 1/2$ to obtain the following equations

$$x_c = (x_1 + x_2)/2 + T(f_1 - f_2)/8 \tag{5.9}$$

$$x_c' = -3(x_1 - x_2)/2T - (f_1 + f_2)/4 \tag{5.10}$$

$$\Delta = f_c - x_c' \tag{5.11}$$

where x_c is the interpolated value of states x_1 and x_2 midway between the nodes at t_1 and t_2. x_c' is the slope of equation (5.8) evaluated at $S = 1/2$. f_1, f_2, and f_c are the same as equation (5.2) and are evaluated using (x_1, u_1), (x_2, u_2), (x_c, u_c) respectively. u_c is simply $\frac{u_1+u_2}{2}$, and the variable T is the time between nodes.

Equation 5.11 is referred to as the defect, and a set of $N * m$ of these equations enter the NLP. The states and controls at each adjacent node are varied in order to set the defects equal to zero. Figure 5-2 shows a graphical representation of this process [16].

Once the NLP program has been solved, a continuous control history can be found

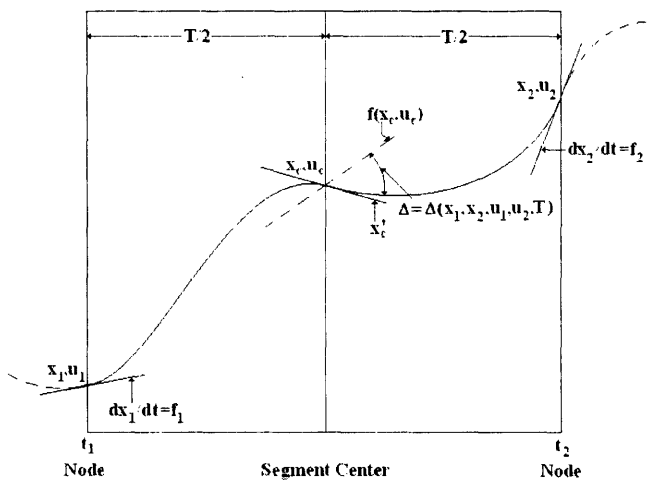


Figure 5-2: With collocation the states and controls at adjacent nodes, for example (x_1, u_1) and (x_2, u_2), are varied in order to drive the defect Δ to zero. This provides an accurate approximation of the system dynamics. [16]

by linearly interpolating the controls at adjacent nodes [16].

5.1.2 Scaling, Analytic Gradients, and Initial Guess

Optimization software typically will converge more efficiently if the variables are of similar magnitude. For the simulations in this chapter the states, controls, and final time are scaled with

$$x_s = \frac{x}{x_{max}} \tag{5.12}$$

$$u_s = \frac{u}{u_{max}} \tag{5.13}$$

$$t_{f_s} = \frac{t_f}{t_{max}} \tag{5.14}$$

where x_{max}, u_{max}, and t_{max} are estimations of the maximum value for each state, control, and final time respectively. The constraints for the NLP are also scaled considering equations (5.12-5.14) in order to maintain consistency.

In the following simulations, analytic gradients for the cost function and all constraints are provided to the optimization software. This improved the convergence time to an optimal solution significantly.

The initial guess for the NLP variables is set to half of the estimated maximum values for the states, controls, and final time.

5.2 Aircraft Models

Three different aircraft dynamic models are presented for this approach: (1) the interceptor maintains a constant speed, (2) the interceptor varies its speed while maintaining a constant altitude, (3) the interceptor has variable speed and altitude. Simulations for the simpler models typically were solved with less computation time; however, the extra degrees of freedom in the more complex models allow the interceptor potentially to complete the maneuver in a faster, more efficient manner.

5.2.1 Interceptor Constant Speed Model

The dynamic model for this section assumes a constant speed for the interceptor and a constant speed and turn rate for the target. Also, the interceptor and target are at the same altitude. The model dynamics are described in a relative frame with the following equations

$$\dot{x}_{rel} = v\cos\Psi - v_{tgt}\cos\Psi_{tgt} \tag{5.15}$$

$$\dot{y}_{rel} = v\sin\Psi - v_{tgt}\sin\Psi_{tgt} \tag{5.16}$$

$$\dot{\Psi} = \frac{\tan(\phi)g}{v} \tag{5.17}$$

$$\dot{\Psi}_{tgt} = d \tag{5.18}$$

where x_{rel} and y_{rel} are the x and y position of the interceptor with respect to the target, v and v_{tgt} the interceptor and target speed respectively, ϕ the interceptor's bank angle, Ψ and Ψ_{tgt} the interceptor and target heading respectively, g the gravitational constant, and d the target's turn rate. The origin is set to the initial starting point of the target with $\Psi_{tgt}(t_0)$ aligned along the x-axis. Ψ and Ψ_{tgt} are defined as 0° along the x-axis and increase in the counter-clockwise direction.

The problem is formulated as an NLP with the following variables:

- states ($m = 3$): $x_{rel}(k)$, $y_{rel}(k)$, $\Psi(k)$
- controls ($n = 1$): $\phi(k)$
- final time: t_f

where $\forall k \in [0 \ldots N]$.

The objective is to perform the maneuver in minimum time with control weighting to assist in converging to a locally optimal, feasible solution

$$\min_{\phi(k)} J = t_f + p \sum_{k=0}^{N} \phi^2(k) \tag{5.19}$$

where p is a small constant.

The constraints for the NLP include dynamic feasibility which is approximated using collocation with the equations in (5.20). These defects, shown as $\Delta_{k,j}$, represent the dynamics of equations (5.15-5.17), whereas equation (5.18) is easily integrated to find the target's heading at each node. An example of calculating these equations is shown in Appendix A. The constraints also include a limit on the maximum bank angle, radar detection avoidance, and final desired heading and position

$$\begin{aligned}
\Delta_{k,j} &= 0 && (5.20)\\
&\forall \quad k \in [1 \ldots N], \forall j \in [1 \ldots m] \\
\phi_{min} &\leq \phi(k) \leq \phi_{max} && (5.21)\\
R_d(k) &\leq \sqrt{x_{rel}^2(k) + y_{rel}^2(k)} && (5.22)\\
\Psi_{tgt}(N) - \Psi(N) &= 90^\circ && (5.23)\\
x_{rel}(N) &= R_s \cos(\Psi_{tgt}(N) + 90^\circ) && (5.24)\\
y_{rel}(N) &= R_s \sin(\Psi_{tgt}(N) + 90^\circ) && (5.25)
\end{aligned}$$

where $R_d(k)$ is the range at which the target's radar can detect the interceptor at node k, and R_s is the final separation range to complete the beam intercept. $R_d(k)$ is defined as

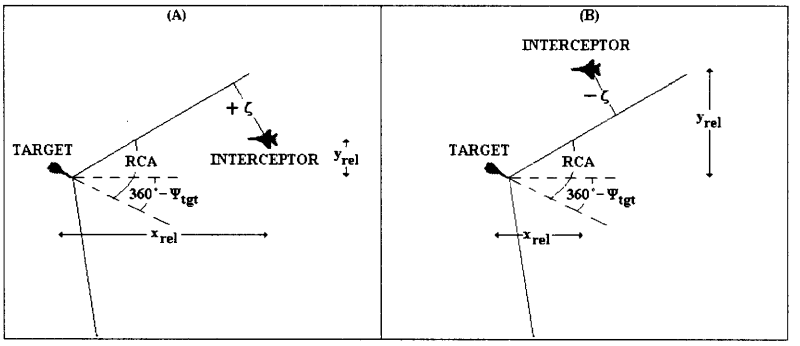


Figure 5-3: ζ is the distance from the interceptor to the edge of the target's radar cone and is defined as positive (Figure A) if the interceptor is within the region of detection and negative otherwise (Figure B). RCA is the target's radar cone angle, and Ψ_{tgt} is the target's heading.

$$R_d(k) = \frac{\nu\sigma^{\frac{1}{4}}(k)}{1+e^{-a\zeta(k)}} \tag{5.26}$$

$$\zeta(k) = \cos(90° - (RCA + \Psi_{tgt}(k)))x_{rel}(k) - \sin(90° - (RCA + \Psi_{tgt}(k)))y_{rel}(k) \tag{5.27}$$

where ν is a variable dependent on the power of the target's radar, $\sigma(k)$ the interceptor's radar cross section, a a constant which is explained later, $\zeta(k)$ the distance from the radar cone (shown in Figure 5-3), and RCA the radar cone angle.

For this scenario, $\sigma(k)$ is a function of $\phi(k)$, $\Psi(k)$, $x_{rel}(k)$, and $y_{rel}(k)$ where the function of these variables is described in Section 2.2.4. It is assumed that the interceptor's angle of attack is nearly a constant throughout the trajectory and does not significantly change the radar cross section of the aircraft.

With the detection range constraint, the sigmoid form of equation (5.26) allows the aircraft to fly outside of the radar cone without concern for radar cross section or range. For this thesis, $a = 16.39/km$ so that if $\sigma = 1m^2$, then the radar detection range can be described with the curves in Figure 5-4 where R_d varies according to the strength of the target's radar. With larger values of a, the radar constraint becomes more like a binary switch. However, if this value was set too large, then the

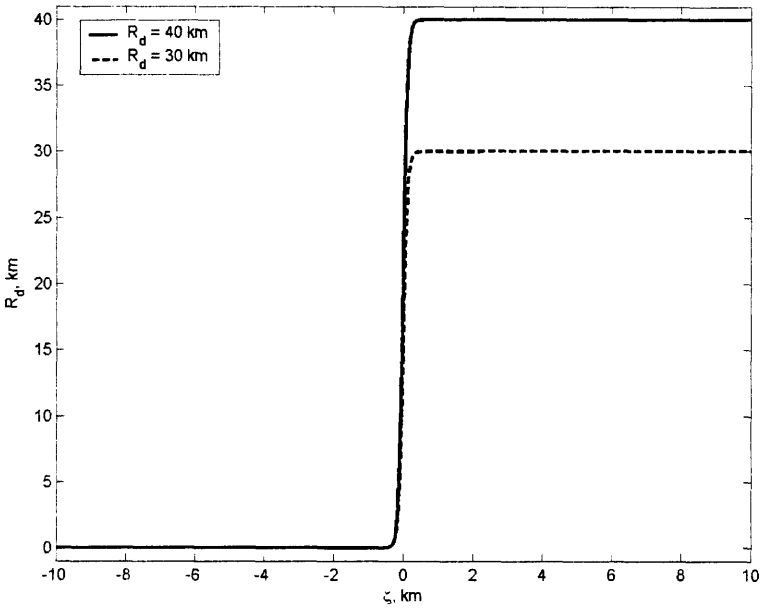


Figure 5-4: Radar detection range as a function of distance from the radar cone ζ with radar cross section held constant

optimization software had difficulty in converging to an answer.

5.2.2 Interceptor Variable Speed Model

The time to perform a beam intercept can be reduced significantly if the interceptor is permitted to vary its airspeed. The following model describes the dynamics of the system, which is adjusted from [22] in order to formulate the problem in the relative frame. The target aircraft is assumed to have a constant speed and turn rate throughout the trajectory.

$$\dot{x}_{rel} = v\cos\Psi - v_{tgt}\cos\Psi_{tgt} \tag{5.28}$$

$$\dot{y}_{rel} = v\sin\Psi - v_{tgt}\sin\Psi_{tgt} \tag{5.29}$$

$$\dot{v} = \frac{T_{max}\delta\cos\alpha - D}{\mu} \tag{5.30}$$

$$\dot{\Psi} = \frac{T_{max}\delta\sin\alpha\sin\phi + L\sin\phi}{\mu v} \tag{5.31}$$

$$\dot{\Psi}_{tgt} = d \tag{5.32}$$

where T_{max} is the interceptor's maximum thrust available, δ the throttle setting, α the interceptor's angle of attack, μ the interceptor's mass, and L and D are the interceptor's lift and drag respectively and are calculated with

$$L = C_L \frac{1}{2} \rho v^2 S \tag{5.33}$$

$$D = C_D \frac{1}{2} \rho v^2 S \tag{5.34}$$

A model similar to an F-16 found in [18] is used for the wing planform area S, and the lift and drag coefficients

$$\begin{aligned} C_L &= 0.0174 + 4.3329\alpha - 1.3048\alpha^2 + 2.2442\alpha^3 \\ &\quad -5.8517\alpha^4 (0 \leq \alpha \leq \pi/6) \end{aligned} \tag{5.35}$$

$$\begin{aligned} C_D &= 0.0476 - 0.1462\alpha + 0.0491\alpha^2 + 12.8046\alpha^3 \\ &\quad -12.6985\alpha^4 (0 \leq \alpha \leq \pi/6) \end{aligned} \tag{5.36}$$

The density is modelled with [18]

$$\rho = \rho_s [1 - 0.00688(h/1000)]^{4.256} \tag{5.37}$$

where h is altitude measured in feet, and ρ_s is the density at sea level.

The maximum thrust available T_{max} is a constant equal to the weight of the interceptor [18]. It is assumed that with changes in airspeed and density, the true thrust available is always greater than or equal to T_{max}.

Now, the nonlinear program has the following variables:

- states ($m = 4$): $x_{rel}(k)$, $y_{rel}(k)$, $v(k)$, $\Psi(k)$
- controls ($n = 3$): $\phi(k)$, $\alpha(k)$, $\delta(k)$
- final time: t_f

where $\forall k \in [0 \ldots N]$.

The same cost function is used as the previous aircraft model with the addition of the controls $\alpha(k)$ and $\delta(k)$.

$$\min_{\phi(k),\alpha(k),\delta(k)} J = t_f + p\sum_{k=0}^{N}\phi^2(k) \tag{5.38}$$

The constraints are also the same as Section 5.1 with a few additions including a constraint on the force balance in the vertical plane, equation (5.45), minimum and maximum speed, angle of attack, and throttle setting. Also the defect equations (5.39) refer to the system dynamic equations (5.28-5.31)

$$\begin{aligned}
\Delta_{k,j} &= 0 && (5.39)\\
&\forall \quad k \in [1\ldots N], \forall j \in [1\ldots m] &&\\
\phi_{min} &\leq \phi(k) \leq \phi_{max} && (5.40)\\
R_d(k) &\leq \sqrt{x_{rel}^2(k) + y_{rel}^2(k)} && (5.41)\\
\Psi_{tgt}(N) - \Psi(N) &= 90^\circ && (5.42)\\
x_{rel}(N) &= R_s\cos(\Psi_{tgt}(N) + 90^\circ) && (5.43)\\
y_{rel}(N) &= R_s\sin(\Psi_{tgt}(N) + 90^\circ) && (5.44)\\
0 &= T_{max}\delta(k)\sin\alpha(k)\cos\phi(k) + L(k)\cos\phi(k) - mg && (5.45)\\
v_{min} &\leq v(k) \leq v_{max} && (5.46)\\
\alpha_{min} &\leq \alpha(k) \leq \alpha_{max} && (5.47)\\
\delta_{min} &\leq \delta(k) \leq \delta_{max} && (5.48)
\end{aligned}$$

A limit on dynamic pressure is not included, because the maximum speed used for the simulations is a more conservative constraint. The maximum speed occurs well within the subsonic region, and in order to include a supersonic aircraft model, the drag coefficient would need to be modelled as a function of Mach number in addition to α.

$R_d(k)$ is still calculated using equations (5.26-5.27), but now the radar cross section is a function of $x_{rel}(k)$, $y_{rel}(k)$, $\phi(k)$, $\Psi(k)$, and $\alpha(k)$. This function is described in

Section 2.2.4.

5.2.3 Interceptor Variable Speed and Altitude Model

The intercept problem is not limited to two-dimensions. In many cases the interceptor should have at least a small altitude offset from the target, which can improve the view of certain aircraft features such as the horizontal tail. The following dynamics are the same as the model in [23] with a small change to make the x and y states with respect to the target's position. Also, the dynamics do not include fuel burn because the duration of the trajectory is short. The same assumptions are made as the previous section for the target aircraft.

$$\dot{x}_{rel} = v\cos\gamma\cos\Psi - v_{tgt}\cos\Psi_{tgt} \tag{5.49}$$

$$\dot{y}_{rel} = v\cos\gamma\sin\Psi - v_{tgt}\sin\Psi_{tgt} \tag{5.50}$$

$$\dot{v} = \frac{T_{max}\delta\cos\alpha - D - \mu g\sin\gamma}{\mu} \tag{5.51}$$

$$\dot{\gamma} = \frac{T_{max}\delta\sin\alpha\cos\phi + L\cos\phi - \mu g\cos\gamma}{\mu v} \tag{5.52}$$

$$\dot{\Psi} = \frac{T_{max}\delta\sin\alpha\sin\phi + L\sin\phi}{\mu v\cos\gamma} \tag{5.53}$$

$$\dot{h} = v\sin\gamma \tag{5.54}$$

$$\dot{\Psi}_{tgt} = d \tag{5.55}$$

where h is the interceptor's altitude, and γ the interceptor's flight path angle. The same models for maximum thrust, lift, and drag are used as the previous section.

The variables for the nonlinear program consist of

- states ($m = 6$): $x_{rel}(k)$, $y_{rel}(k)$, $v(k)$, $\gamma(k)$, $\Psi(k)$, $h(k)$
- controls ($n = 3$): $\phi(k)$, $\alpha(k)$, $\delta(k)$
- final time: t_f

where $\forall k \in [0 \ldots N]$.

The same objective function is used for this model

$$\min_{\phi(k),\alpha(k),\delta(k)} J = t_f + p \sum_{k=0}^{N} \phi^2(k) \tag{5.56}$$

subject to the following constraints

$$\begin{aligned}
\Delta_{k,j} &= 0 && (5.57)\\
&\forall \quad k \in [1 \ldots N], \forall j \in [1 \ldots m] \\
-\phi_{min} &\leq \phi(k) \leq \phi_{max} && (5.58)\\
R_d(k) &\leq \sqrt{x_{rel}^2(k) + y_{rel}^2(k) + h_{rel}^2(k)} && (5.59)\\
\Psi_{tgt}(N) - \Psi(N) &= 90^\circ && (5.60)\\
x_{rel}(N) &= R_s \cos(\Psi_{tgt}(N) + 90^\circ) && (5.61)\\
y_{rel}(N) &= R_s \sin(\Psi_{tgt}(N) + 90^\circ) && (5.62)\\
v_{min} &\leq v(k) \leq v_{max} && (5.63)\\
\alpha_{min} &\leq \alpha(k) \leq \alpha_{max} && (5.64)\\
\delta_{min} &\leq \delta(k) \leq \delta_{max} && (5.65)\\
\gamma_{min} &\leq \gamma(k) \leq \gamma_{max} && (5.66)
\end{aligned}$$

where equations (5.57) are the defects which approximate the system dynamics of equations (5.49-5.54).

Now, the radar detection constraint equation (5.59) includes the relative altitude between the interceptor and target $h_{rel}(k)$ for calculating range. The range at which the target can detect the interceptor $R_d(k)$ is also modelled differently

$$R_d(k) = \frac{\nu\sigma^{\frac{1}{4}}(k)}{1+e^{-aX'(k)}} \tag{5.67}$$

$$X'(k) = \eta(k) - \zeta(k) \tag{5.68}$$

$$\eta(k) = x_{rot}(k)\tan(RCA) \tag{5.69}$$

$$\zeta(k) = \sqrt{h_{rel}^2(k) + y_{rot}^2(k)} \tag{5.70}$$

$$x_{rot}(k) = \cos(-\Psi_{tar}(k))x_{rel}(k) - \sin(-\Psi_{tar}(k))y_{rel}(k) \tag{5.71}$$

$$y_{rot}(k) = \sin(-\Psi_{tar}(k))x_{rel}(k) + \cos(-\Psi_{tar}(k))y_{rel}(k) \tag{5.72}$$

As stated before, the sigmoid form of equation (5.67), allows the radar constraint to be switched on and off depending on whether the interceptor is inside or outside of the target's radar cone. $\sigma(k)$ is a function of $x_{rel}(k)$, $y_{rel}(k)$, $h_{rel}(k)$, $\phi(k)$, $\Psi(k)$, and $\alpha(k)$. A set of coordinate transformations and a few equations in Section 2.2.4 map these variables to the radar cross section. The purpose of equations (5.68-5.72) is to set X' as a negative number if the interceptor is within the target's radar cone, otherwise, X' is positive. A graphical representation of the variables used to determine X' is shown is Figure 5-5.

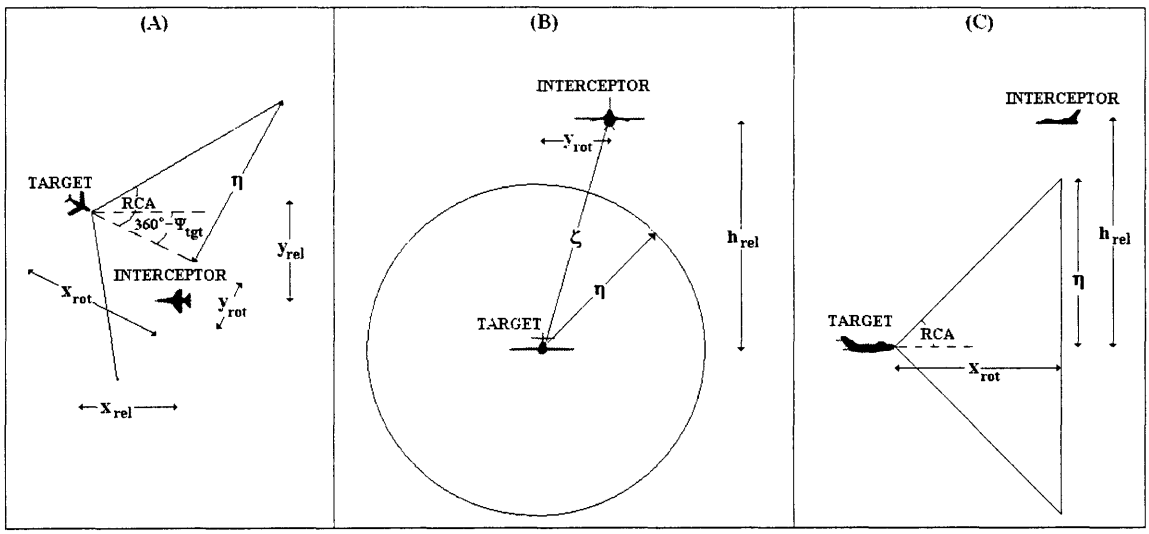


Figure 5-5: (x_{rot}, y_{rot}) are (x_{rel}, y_{rel}) rotated by the negative of the target's heading angle Ψ_{tgt}. RCA is the target's radar cone angle, and η is the radius of the radar cone at x_{rot}. h_{rel} is the interceptor's altitude relative to the target's. ζ is the distance from the target to the interceptor in the plane containing altitude and the rotated relative y position y_{rot}.

5.3 Results

5.3.1 Interceptor Constant Speed Model Results

For the following scenarios, the interceptor and target maintain a constant speed and altitude. The interceptor's objective is to complete the maneuver in minimum time with a small weight on bank angle

$$\min_{\phi(k)} J = t_f + p \sum_{k=1}^{N} \phi^2(k) \tag{5.73}$$

Additionally, the interceptor must avoid radar detection throughout the trajectory. Table 5.1 shows the constants used for the first two simulations. In Figure 5-6, a trajectory is shown where the interceptor ignores the radar of the target, and in Figure 5-7 radar constraints are added with the target's radar detection range set to $12.2km$ when the interceptor's radar cross section is $1m^2$. In exchange for attempting to avoid radar detection, the trajectory duration is 11.6 seconds longer.

Table 5.1: Constants for direct collocation simulations with constant speed aircraft model

N	50	g	$9.8m/s^2$
ϕ_{min}	$-60°$	ϕ_{max}	$60°$
$\Psi(t_0)$	$180°$	$\Psi_{tgt}(t_0)$	$0°$
v	$152.5m/s$	v_{tgt}	$152.5m/s$
$x_{rel}(t_0)$	$27.5km$	$y_{rel}(t_0)$	$0km$
R_s	$3.05km$	RCA	$45°$
p	0.1		

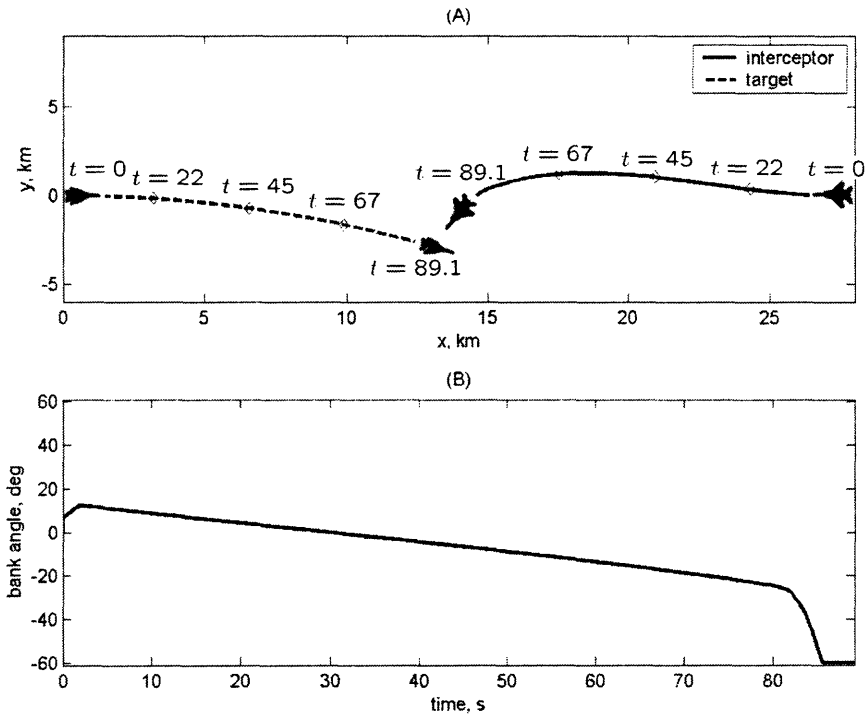


Figure 5-6: For this scenario, the detection range of the target's radar is set to zero, $R_d(\sigma = 1m^2) = 0km$, and the target is turning at a constant rate of $0.3°/s$. Figure (A) shows the trajectories of both aircraft, and Figure (B) is the bank angle history of the interceptor. The interceptor initially makes a small turn away from the target followed by a very gradual turn towards the target. At the end of the trajectory, a sharp turn is applied to achieve the final desired relative heading and position.

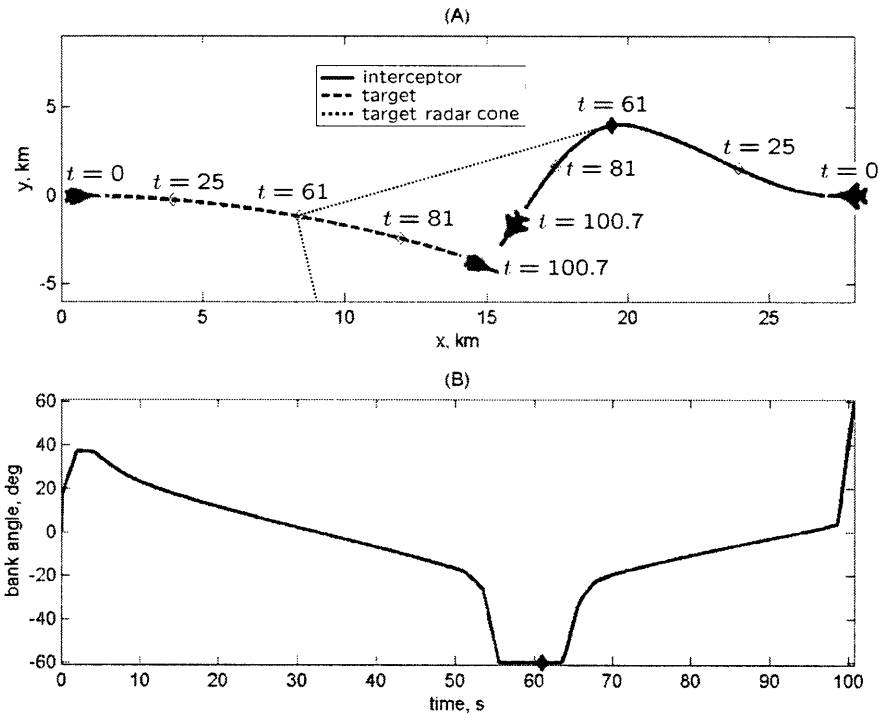


Figure 5-7: In this case, the detection range of the target's radar is $12.2km$ when the interceptor presents a radar cross section of $1m^2$, and the target is turning at a constant rate of $0.3°/s$. Figure (A) shows the trajectories of the aircraft, and Figure (B) is the bank angle history of the interceptor. The interceptor begins the maneuver with a turn away from the target to fly to the point, shown with a solid black diamond, at which it will cross the target's radar cone. Just before reaching this point, maximum negative bank angle is applied to return the interceptor to a heading which will allow completion of the maneuver. The interceptor travels along a nearly straight path and then finally applies maximum positive bank angle to obtain the desired endstate.

Table 5.2: Results for constant speed trajectories including radar avoidance constraints. The term $R_d(\sigma = 1m^2)$ refers to the detection range of the target's radar when the interceptor's radar cross section is $1m^2$.

	path length (km)	trajectory time (s)	computation time (s)
$R_d(\sigma = 1m^2) = 0km$	13.6	89.1	15.1
$R_d(\sigma = 1m^2) = 12.2km$	15.4	100.7	52.6

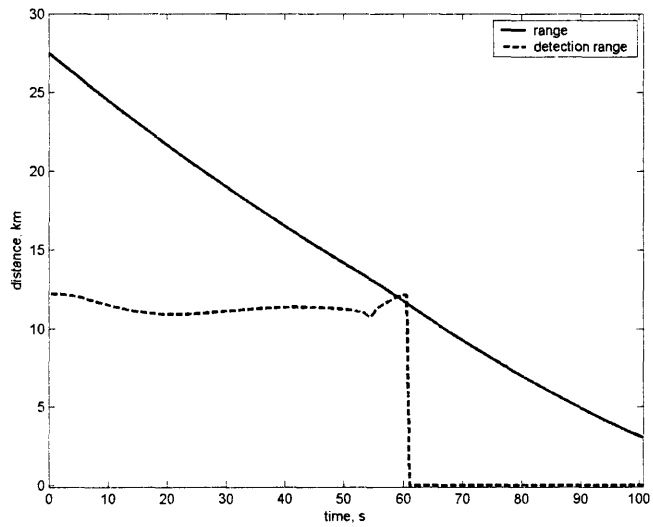


Figure 5-8: Limitation of radar avoidance constraint for the second simulation, $R_d(\sigma = 1m^2) = 12.2km$. The interceptor is detected by the target's radar seconds before crossing the target's radar cone.

Because the radar detection constraint is only applied at the nodes, for nearly every simulation the interceptor is detected by the target's radar for a brief moment before crossing the radar cone, as shown in Figure 5-8.

There are a few variations to the problem formulation which might prevent radar detection while the interceptor crosses the target's radar cone. The density of the nodes around this area could be increased, using the previously calculated trajectory as an initial guess in forming a new solution. Also, the optimization problem could be solved, increasing the target's radar power to more than its actual value. Finally, the sigmoid equation could be altered with the term b in order to shift the radar cone

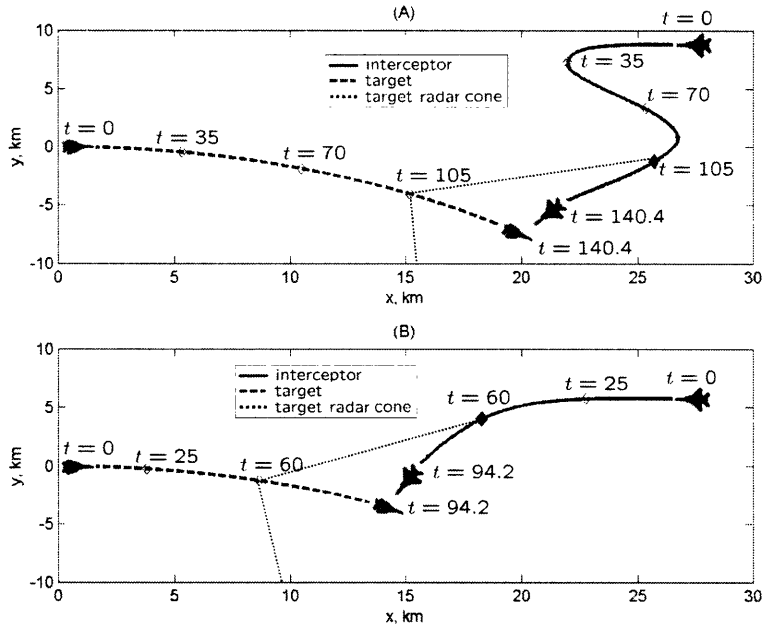


Figure 5-9: This figure shows two locally optimal trajectories where each was produced using a different initial guess for the NLP variables. Trajectories (A) and (B) are completed in $140.4s$ and $94.2s$ respectively. The solid black diamond is the point at which the interceptor crosses the target's radar cone.

$$\frac{\nu\sigma^{\frac{1}{4}}(k)}{1+e^{-a\zeta(k)}} \Rightarrow \frac{\nu\sigma^{\frac{1}{4}}(k)}{1+e^{-a(\zeta(k)+b)}} \tag{5.74}$$

A problem with casting the optimal control problem as a nonlinear program is the existence of multiple local minima. Of $1,140$ simulations run using MATLAB's Optimization Toolbox, which varied with initial position of the interceptor with respect to the target and target radar strength, 99.74% appear to be solutions that could be globally optimal. For the three cases which were clearly local minima, the simulation was performed again, changing the guess of the initial variables at the nodes by multiplying by 1.1, and a better solution was obtained. One example is shown in Figure 5-9.

Another issue with nonlinear programming is the time duration for the optimization software to converge to a solution. Of the $1,140$ simulations, the average computation time to reach a locally optimal solution was 24.25 seconds with a standard

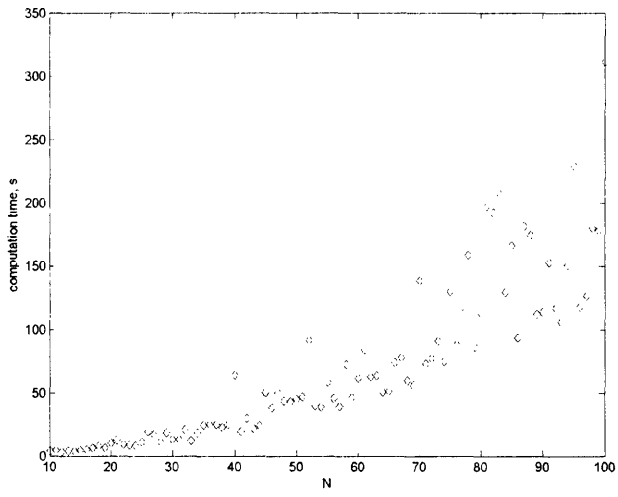


Figure 5-10: Impact of N on computation time with the same constants as the scenario in Figure 5-7

deviation of 19.04 seconds. In compiled implementation and on a faster processor, these computation times may be acceptable for real-time implementation. However, the maximum computation time was over 6 minutes. This uncertainty in reaching a locally optimal solution clearly makes online implementation difficult.

Generally, the computation time will increase as the number of variables increases for the NLP. An important parameter, which the user can select, is N. This number represents the number of nodes for which the problem is discretized. As previously stated, the number of variables for the NLP is $(N+1)*m+(N+1)*n+1$ where m and n are the number of states and controls at each node respectively. Choosing N is a balancing process. With N too small, large portions of the trajectory ignore state constraints such as radar detection avoidance. With larger values of N, the computation time generally increases for the NLP optimization software as shown in Figure 5-10. For the previous simulations N was set to 50, in an attempt to balance the computation time with the amount of time that the interceptor might be detected by the target's radar. In the following results sections, N is set to 30 in order to speed up the optimization software for aircraft models with greater values of m, n, and the number of constraints.

5.3.2 Interceptor Variable Speed Model Results

For the simulations in this section, the interceptor is able to vary its speed throughout the trajectory, and both aircraft are flying at a constant altitude of $6.1km$. The interceptor is limited to flight within the subsonic region in order that the drag coefficient can be expressed solely in terms of angle of attack. Also, the interceptor can only perform a 2-g turn which is equivalent to a maximum bank angle of 60°. With larger values of maximum bank angle, the accuracy of the collocation method for approximating the aircraft dynamics decreased. The combination of speed limits and the limit on bank angle produces a curve of maximum turn rates shown in Figure 5-11.

Table 5.3: Constants for variable speed aircraft model simulations using direct collocation

N	30	g	$9.8m/s^2$
ϕ_{min}	$-60°$	ϕ_{max}	$60°$
v_{min}	$122m/s$	v_{max}	$213.5m/s$
α_{min}	$0°$	α_{max}	$30°$
δ_{min}	0	δ_{max}	1
$\Psi(t_0)$	$180°$	$\Psi_{tgt}(t_0)$	$0°$
$v(t_0)$	$152.5m/s$	v_{tgt}	$152.5m/s$
$x_{rel}(t_0)$	$27.5km$	$y_{rel}(t_0)$	$0km$
R_s	$3.05km$	RCA	$45°$
h	$6.1km$	ρ_s	$0.9048kg/m^3$
p	0.1	$mass$	$9,299kg$
T_{max}	$91.13kN$		

The objective is the same as the previous section where the interceptor flies the trajectory in minimum time with a small weight on bank angle

$$\min_{\phi(k),\alpha(k),\delta(k)} J = t_f + p\sum_{k=1}^{N}\phi^2(k) \tag{5.75}$$

Figures 5-12 and 5-14 show two examples of the beam intercept and the corresponding bank angle history. For the first scenario, the interceptor performs the maneuver ignoring radar detection constraints. In the second scenario, the intercep-

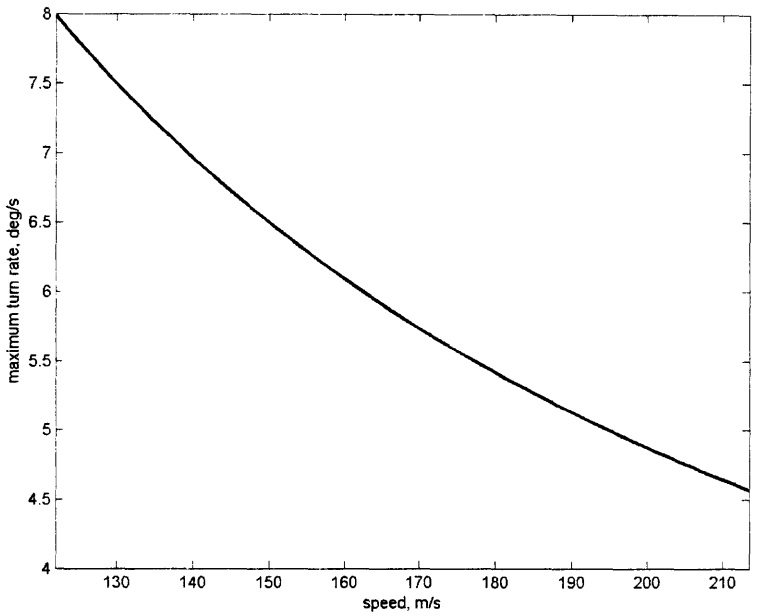


Figure 5-11: Interceptor maximum turn rate for variable speed, constant altitude model

tor attempts to avoid the target's radar, which has a detection range of $12.2km$ when the interceptor presents a radar cross section of $1m^2$. In both cases, the target is turning slowly at a constant rate of $0.3°/s$. The path length, trajectory time, and computation time for both scenarios are shown in Table 5.4.

Table 5.4: Results for variable speed trajectories shown in Figures 5-12 and 5-14. The term $R_d(\sigma = 1m^2)$ refers to the detection range of the target's radar when the interceptor's radar cross section is $1m^2$.

	path length (km)	trajectory time (s)	computation time (s)
$R_d(\sigma = 1m^2) = 0km$	15.6	76.7	175.5
$R_d(\sigma = 1m^2) = 12.2km$	16.8	81.0	299.5

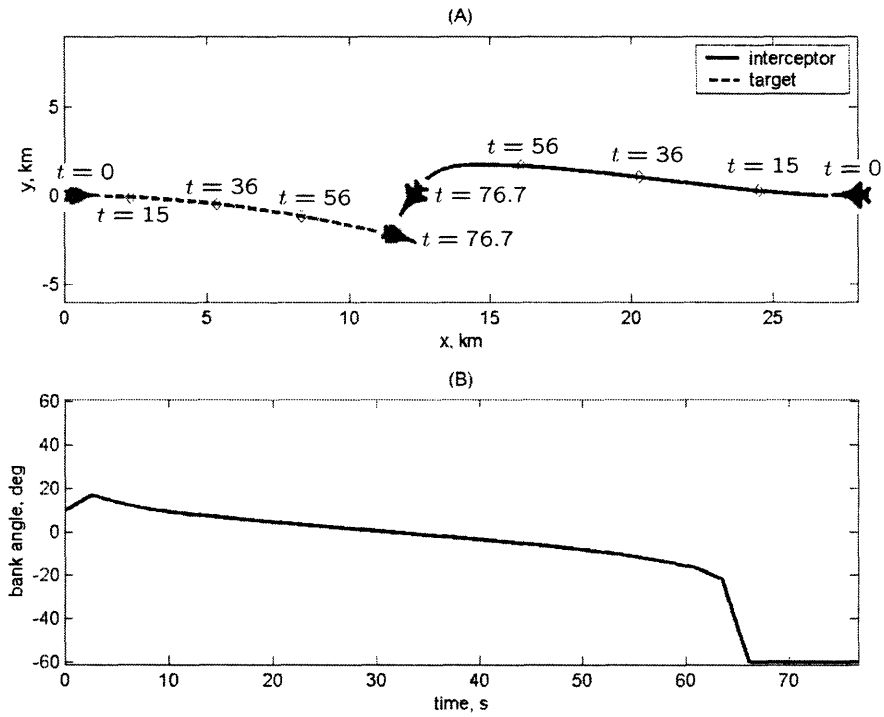


Figure 5-12: In this scenario, the interceptor performs the beam intercept without considering the target's radar. The trajectory, shown in Figure (A), consists of a slow turn away from the target, a slow turn towards the target, and a sharp turn at maximum bank angle to complete the maneuver. The interceptor's bank angle history is shown in Figure (B).

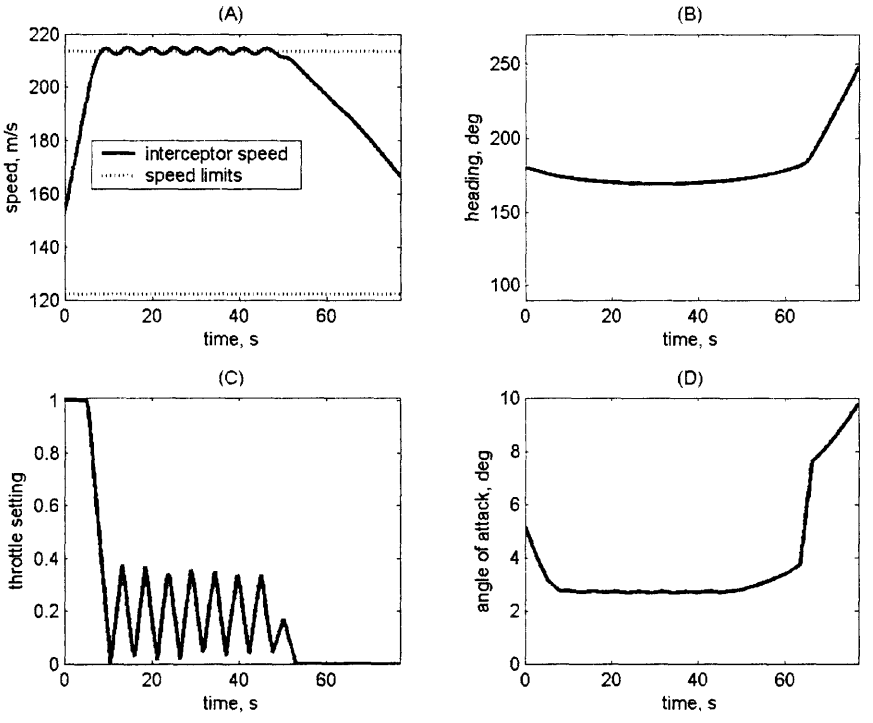


Figure 5-13: Additional control and state histories for the trajectory in Figure 5-12. The interceptor initially applies full throttle to reach its maximum speed. Although the maximum speed limit is met at the nodes used in the NLP, the constraint is violated slightly between the nodes. For the final 30 seconds of the trajectory, the interceptor slows its speed in order to obtain a faster turn rate.

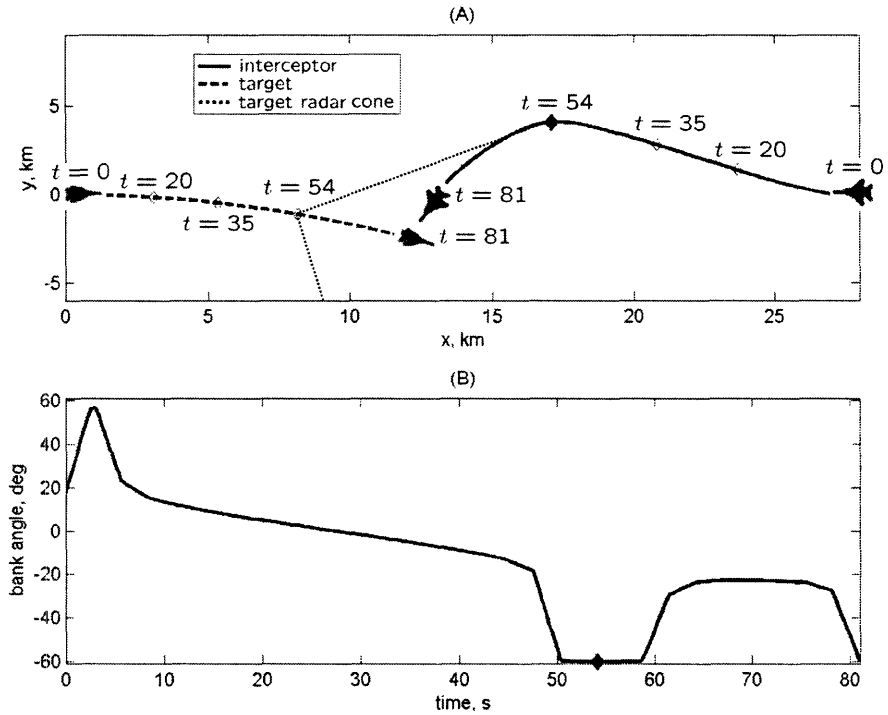


Figure 5-14: At the beginning of the trajectory, the interceptor changes its heading in order to cross the target's radar cone at the point marked by the solid diamond. At this range, along with the corresponding radar cross section, the target is unable to detect the interceptor. Prior to reaching the black diamond, the interceptor applies maximum negative bank angle to wrap around the target's radar cone. Near the end of the trajectory, the interceptor applies maximum bank angle to obtain the desired endstate.

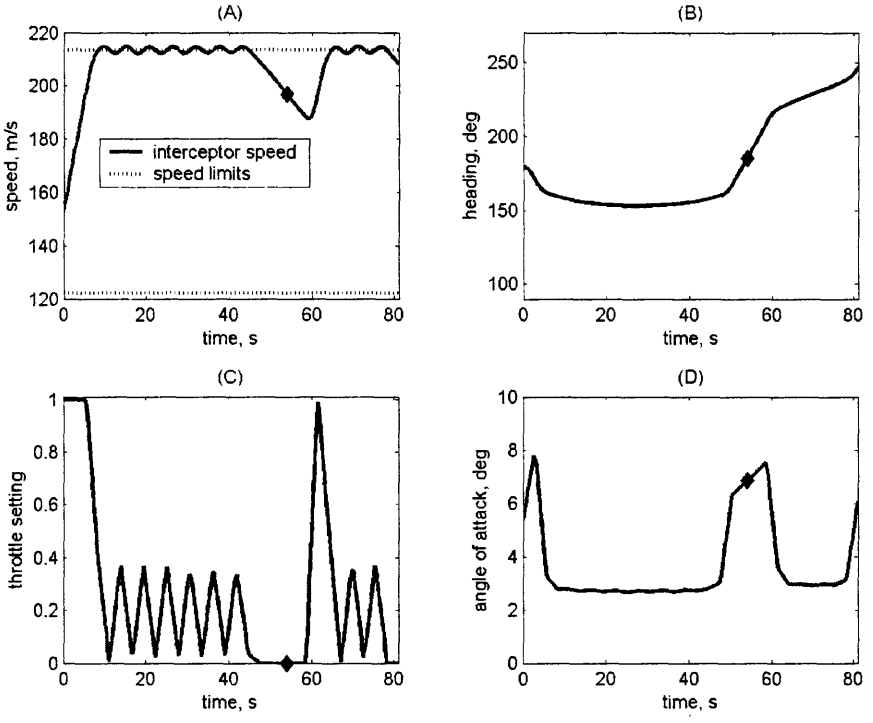


Figure 5-15: Additional control and state histories for the trajectory in Figure 5-14. The interceptor begins the maneuver, applying full throttle to increase its speed to the maximum limit. Before crossing the target's radar cone which is shown with the black diamond, the interceptor decreases its speed to obtain a faster turn rate. Soon afterwards, the interceptor increases its speed again to the limit. During the final seconds of the maneuver, the throttle is dropped to zero, decreasing the speed slightly.

Figures 5-13 and 5-15 show that for both trajectories the maximum speed constraint is violated. This problem occurs because the state constraint involving speed is only applied at the nodes. A more dense discretization with additional nodes placed in the vicinity of where the violation occurs could fix the problem. However, for these simulations, the maximum speed is set conservatively so that the aircraft is still within the subsonic region if the constraint is violated by a small amount.

Another potential problem is the rapid variations in throttle command. This could be addressed by adding additional inequality constraints to limit the rate of change between nodes, which might also indirectly fix the maximum speed violations.

As stated in the previous section, a shortcoming to creating trajectories with direct collocation is the possibility of converging to a local minimum. The maneuvers in Figures 5-12 and 5-14 appear to be candidates for the globally optimal solution. However, in 5.2% of 522 conducted simulations, where the boundary conditions and radar strength were varied, the converged solution is definitely not a global minimum. A better solution can often be obtained by simply varying the initial guess for the NLP as shown in Figure 5-16.

Also, the computation time to reach a converged solution is still a problem. Of the 522 simulations, the average computation time was 208 seconds with a standard deviation of 147 seconds. The maximum time to reach a locally optimal solution was over 41 minutes. With computation times that are potentially this long, real-time implementation seems questionable.

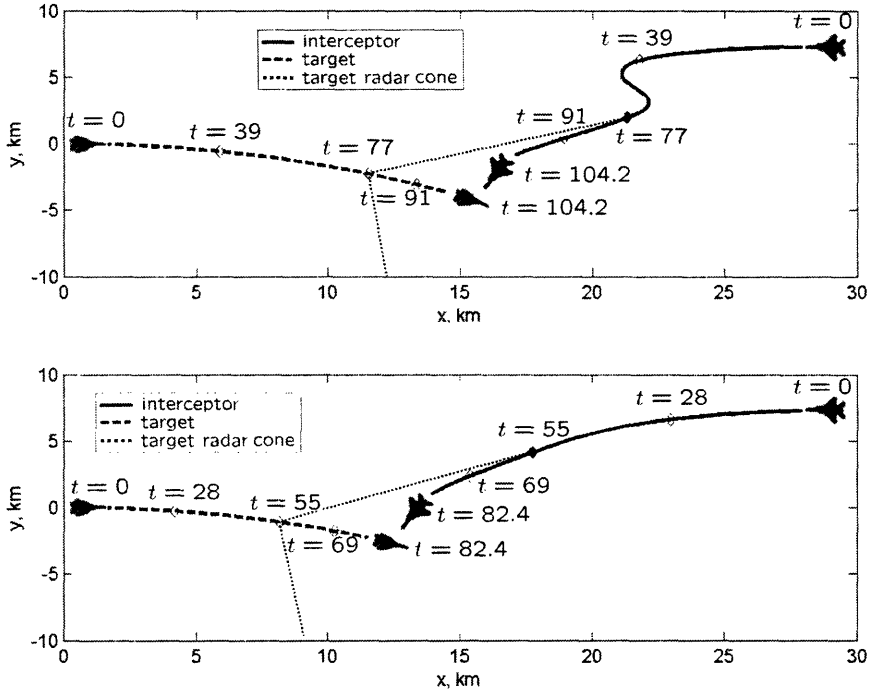


Figure 5-16: For the above trajectories, the same constants are used as in Table 5.4, but $x_{rel}(t_0) = 28.6km$ and $y_{rel}(t_0) = 7.3km$. The target's radar power is set to detect an aircraft at $12.2km$ when the radar cross section is $1m^2$. Figure (A) shows a trajectory that obeys all of the NLP constraints, but is clearly just a locally optimal solution. Figure (B) is a new trajectory with the same initial conditions as Figure (A), but the initial guess for the NLP variables was perturbed by 110%. This new trajectory resembles what could be the optimal solution.

5.3.3 Interceptor Variable Speed and Altitude Model Results

For this scenario, the interceptor must perform a beam intercept in three dimensions. The trajectory's initial conditions are shown in Table 5.5. In addition to finishing the trajectory with the desired relative heading and position in the (x,y) plane, the interceptor must finish the maneuver with $610m$ of altitude separation above the target. Throughout the duration of the maneuver, the target varies its heading with a constant turn rate and maintains a constant speed. For the first scenario presented

Table 5.5: Constants for direct collocation simulations with variable speed and altitude aircraft model

N	30	g	$9.8m/s^2$
ϕ_{min}	$-60°$	ϕ_{max}	$60°$
v_{min}	$122m/s$	v_{max}	$213.5m/s$
α_{min}	$0°$	α_{max}	$30°$
δ_{min}	0	δ_{max}	1
γ_{min}	$-20°$	γ_{max}	$20°$
$\Psi(t_0)$	$180°$	$\Psi_{tgt}(t_0)$	$0°$
$v(t_0)$	$152.5m/s$	v_{tgt}	$152.5m/s$
$x_{rel}(t_0)$	$27.5km$	$y_{rel}(t_0)$	$0km$
R_s	$3.05km$	RCA	$45°$
ρ_s	$0.9048kg/m^3$	$h(t_0)$	$6.1km$
$h(t_f)$	$6.405km$	h_{tgt}	$5.795km$
RCA	$45°$	p	0.1
$mass$	$9,299kg$	T_{max}	$91.13kN$

in this section, the target is turning at a rate of $0.3°/s$, and the interceptor performs the trajectory without considering the radar of the target. In the second scenario, the target is flying along a straight path, and the interceptor must avoid the target's radar which can detect an aircraft at $9.2km$ when the radar cross section is $1m^2$.

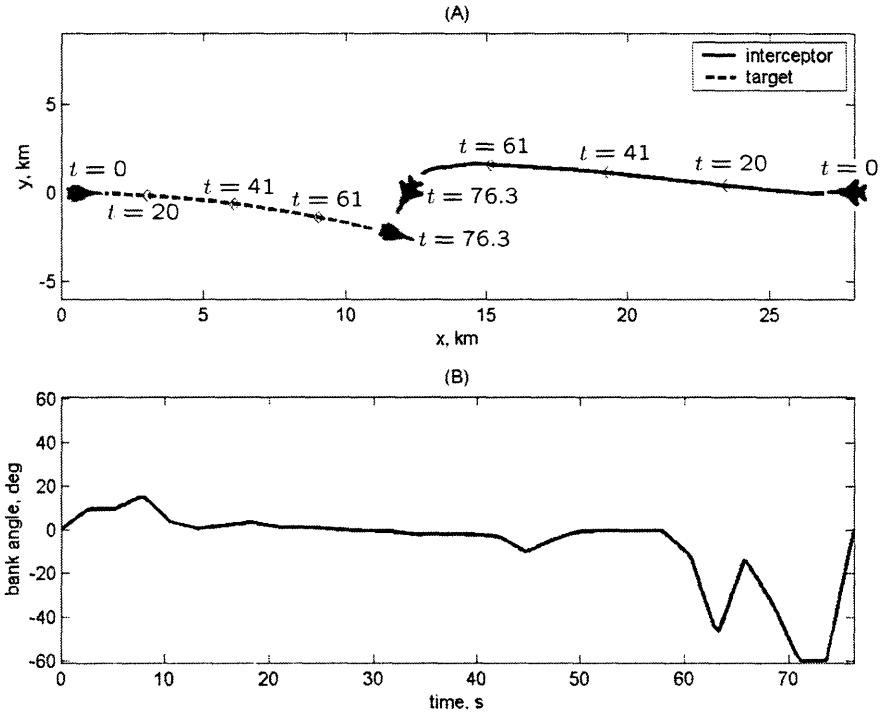


Figure 5-17: Figure (A) shows a beam intercept trajectory in the (x,y) plane where the target is turning at a constant rate of $0.3°/s$. The interceptor begins with a slight turn away from the target and travels along an approximately straight line for the majority of the maneuver. The trajectory is completed with a final turn towards the target. Figure (B) is the interceptor's bank angle history.

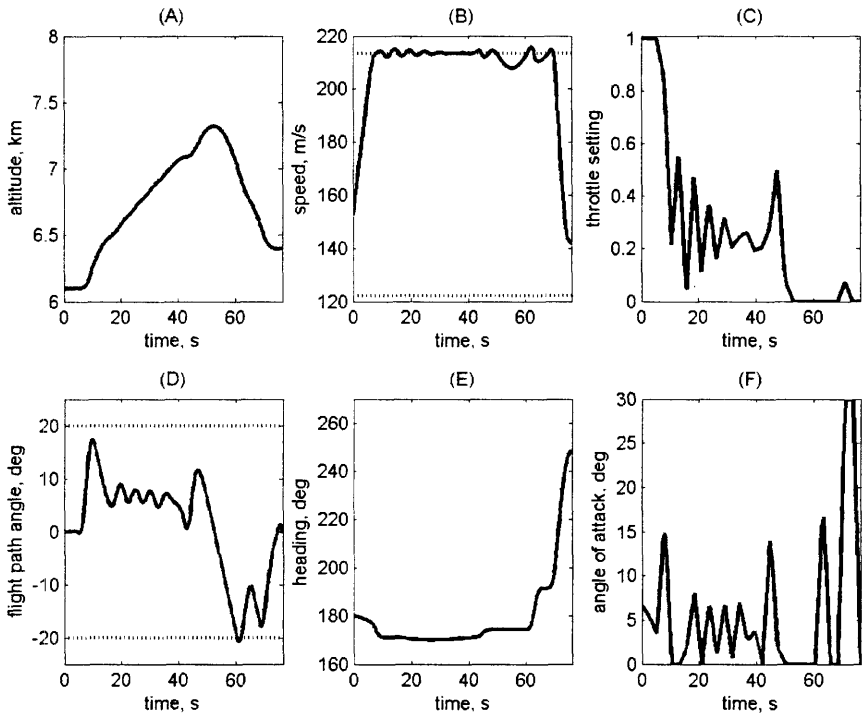


Figure 5-18: This figure shows the additional states and controls of the trajectory found in Figure 5-17. Initially, the interceptor climbs to an altitude significantly higher than the desired final value. It seems as if this excess climb would only slow the completion of the maneuver. This lack of intuitiveness in the trajectory might indicate that the solution is not close to the global optimum.

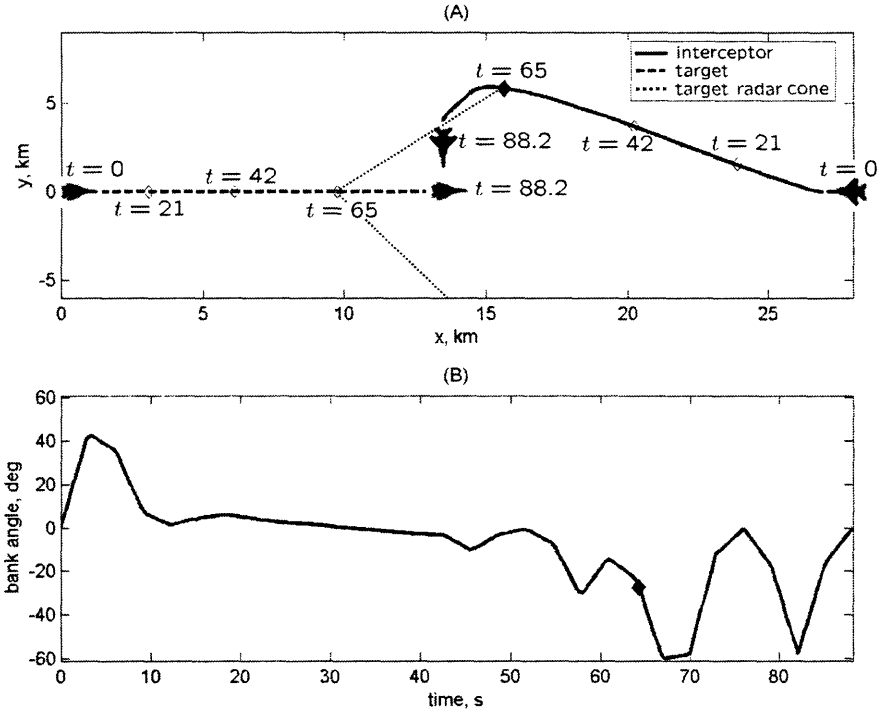


Figure 5-19: Figure (A) shows the trajectory in the (x,y) plane and Figure (B) is the bank angle history of the interceptor. The interceptor begins the maneuver with a turn away from the target to fly to a point along the radar cone, marked by a black diamond, where it will not be detected. Upon reaching this point, a sharp turn is applied to point the interceptor's velocity vector more towards the target. Then, the aircraft flies along a nearly straight path and finishes the trajectory with a sharp turn.

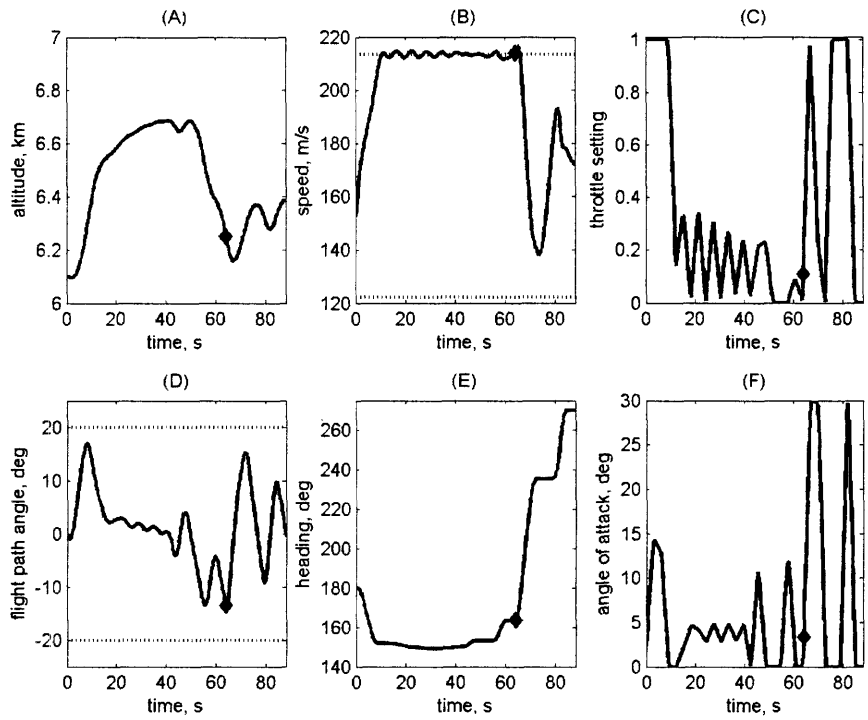


Figure 5-20: Figures (B) and (E) show that the interceptor lowers its speed when a large turn is initiated. The altitude history shown in Figure (A) is less intuitive and may be the result of the solution having converged to a local minimum.

Table 5.6: Results for variable speed and altitude trajectories shown in Figures 5-17 and 5-19. The term $R_d(\sigma = 1m^2)$ refers to the detection range of the target's radar when the interceptor's radar cross section is $1m^2$.

	path length (km)	trajectory time (s)	computation time (s)
$\dot{\Psi}_{tgt} = -0.3°/s, R_d(\sigma = 1m^2) = 0km$	15.74	76.3	712.9
$\dot{\Psi}_{tgt} = \ 0.0°/s, R_d(\sigma = 1m^2) = 9.2km$	17.55	88.2	659.1

Optimal solutions for this aircraft model where significantly more difficult to obtain than with the constant altitude models, as is shown by the computation times in Table 5.6. There is some uncertainty as to whether the results shown above are close to the globally optimal solution, because the altitude history shown in both scenarios lacks intuition. For many simulations, the optimization software converged to a peculiar solution where the interceptor flies a circuitous route. This occurred with larger values of the target's radar strength. In some cases, providing the optimization software with a better initial guess seemed to improve the converged solution. For instance, an initial guess that is dynamically feasible and meets the final endstate constraints usually helped convergence for the problem with added radar constraints.

5.3.4 Discussion of Results

The constant speed and altitude aircraft model simulations had the fastest convergence times and also had the highest rate of converging to what appears to be the globally optimal solution. For a real-time implementation, a combination of the different models could be used. Initially, a trajectory would be produced for the constant speed aircraft model. Once the computer has reached a solution, additional trajectories could be produced using the models with greater degrees-of-freedom. With this approach, the interceptor might not obtain the optimal trajectory, but the chances of obtaining a feasible trajectory are increased.

Chapter 6

Trajectory Interpolation

In this chapter, the trajectory interpolation method of [12] is applied for producing maneuvers which resemble a beam intercept. The idea behind this approach is as follows. Initially, a set of trajectories for a particular type of maneuver are produced offline using nonlinear programming software. Each trajectory in the set accomplishes a similar goal and closely resembles the others. The trajectories are unique in that they have different parameters such as boundary conditions encompassing a variety of values. For example, a set of beam intercepts might be formulated with each trajectory containing a different initial range between the target and the interceptor. The second step of the method is the online calculation of trajectories. The method interpolates between the previously calculated trajectories to find a trajectory for the desired parameters.

Trajectory interpolation is essentially a relaxation of a nonlinear parametric programming problem where the objective function is removed [12]. Feasible trajectories are synthesized as an analytic continuation problem from a compact parametric representation of a particular set of trajectories. In creating new trajectories, the focus is on maintaining feasibility. Optimality is not explicitly considered; however, in certain cases the interpolated trajectories are close to the optimal solution [13].

The set of feasible trajectories is described by

$$h(p, \alpha) = 0 \tag{6.1}$$

$$g(p) \leq 0 \tag{6.2}$$

where p is the set of analytic variables which define a trajectory. In this thesis, the values of p are b-spline coefficients. α is a vector which allows the user to chose specific values for the parameters which correspond to a desired trajectory. The equality constraints typically describe equations encompassing aircraft dynamics and boundary conditions. The inequality constraints impose limits on the states and controls of the vehicle.

Given two trajectories of similar form, v_1 and v_2 where $v = [p; \alpha]$, the method interpolates from one to the other, creating a similar set of trajectories. Beginning at trajectory v_1, the method interpolates towards v_2 with a series of steps. The direction of the step is formed by creating $u(s)$ where

$$u(s) = v_2 - v_1 \tag{6.3}$$

and projecting this vector back onto the set of feasible trajectories defined by equations 6.1 and 6.2. The projection is defined as

$$\hat{u} = Z^+ u = Z(Z^T Z)^{-1} Z^T (v_2 - v_1) \tag{6.4}$$

where Z is a basis for the nullspace of $D_v[h; g_{J_0}]$. $D_v[h; g_{J_0}]$ is a matrix consisting of the partial derivatives of each equality constraint and active inequality constraint, which are denoted with the set J_0, with respect to each of the variables in v. Along the path towards v_2, the method will reach the desired trajectory v_d which contains the desired parameters in α. The method is shown in Figure 6-1 and succinctly described with the following Pseudo-code found in [12].

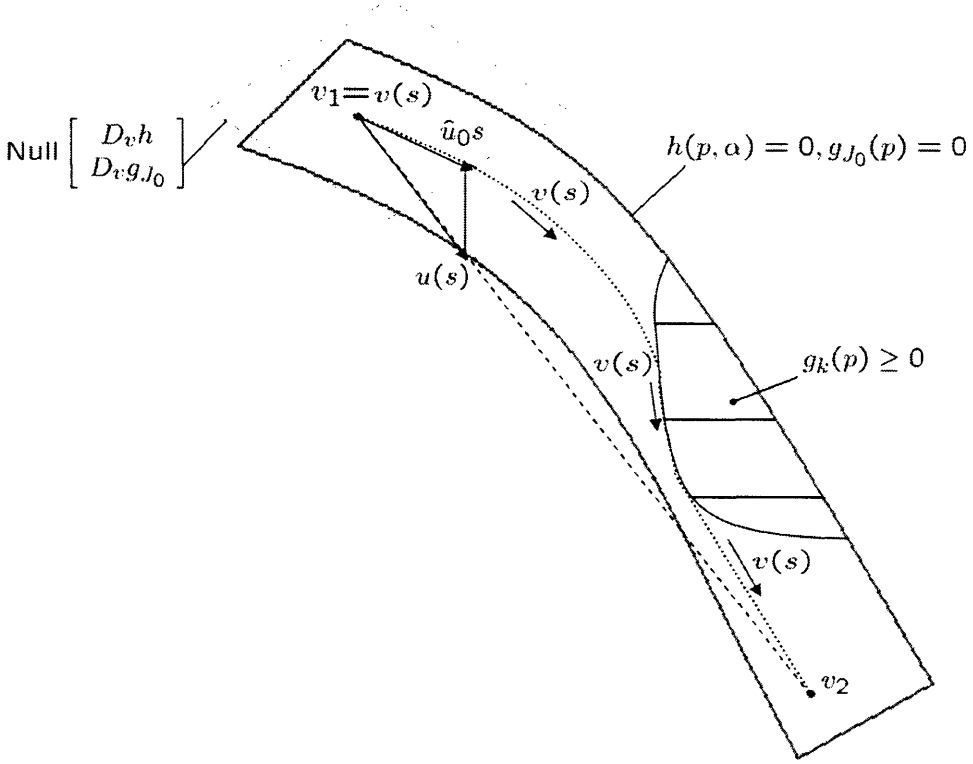


Figure 6-1: This figure found in [12] describes the trajectory interpolation method. The algorithm begins at trajectory v_1 and steps towards trajectory v_2. At each step feasibility is maintained by projecting the direction vector $u(s)$ back onto the hyperplane of equality constraints $h(p, \alpha)$ and active inequality constraints $g_{J_0}(p) = 0$.

Pseudo-code for Interpolation Algorithm

- Obtain feasible maneuvers $v_1 = [p_1; \alpha_1]$ and $v_2 = [p_2; \alpha_2]$ of the same type but satisfying numerically different boundary conditions.
- Integrate $\dot{v}(s) = \hat{u}(s, v)$ with independent variable s along $[0, \alpha_{goal} - \alpha_1]$ with initial condition $v(0) = v_1$ and $\hat{u}(s, v)$ at any point s given by the following logic:

1: $u(s) = v_2(s) - v(s)$
2: evaluate $D_v h(v(s))$ — local equality derivatives
3: $Z =$ basis of Null$(D_v h)$ — tangent space basis
4: $\hat{u}_0 = \text{Proj}_Z u$ — project difference using equation 6.4
5: $\hat{u}_0 \leftarrow \hat{u}_0 \cdot (d\alpha/ds)^{-1}$ — normalize arc length
6: $J_1 = \{i | g_i(p) \geq 0\}, \ \forall i \in [1 \ldots M]$ — test for active inequalities where M is the number of inequality constraints
7: **if** $J_1 = \emptyset$ — none active
8: $\quad \hat{u} = \hat{u}_0$
9: **else**
10: $\quad J_2 = \{j \in J_1 | D_v g_j \cdot \hat{u}_0 \geq 0\}$ — test 1st-order inequality behavior
11: $\quad$ **if** $J_2 = \emptyset$ — none active to 1st-order
12: $\quad\quad \hat{u} = \hat{u}_0$
13: $\quad$ **else**
14: $\quad\quad Z' =$ basis of Null $([D_v h; D_v g_{J_2}])$ — follow 1st-order active inequalities
15: $\quad\quad \hat{u} = \text{Proj}_{Z'} u$ — re-project difference
16: $\quad\quad \hat{u} \leftarrow \hat{u} \cdot (d\alpha/ds)^{-1}$ — normalize arc length
17: $\quad$ **end**
18: **end**

6.1 Aircraft Kinematic Model

The interceptor is modelled as a constant speed aircraft, and the target is assumed to fly with a constant turn rate throughout the trajectory. Radar avoidance constraints

are not added to the problem. The system dynamics are described in a relative frame using the same constant speed aircraft model, found in Section 5.1.

$$\dot{x} = v\cos\Psi - v_{tgt}\cos\Psi_{tgt} \tag{6.5}$$

$$\dot{y} = v\sin\Psi - v_{tgt}\sin\Psi_{tgt} \tag{6.6}$$

$$\dot{\Psi} = \frac{\tan(\phi)g}{v} \tag{6.7}$$

$$\dot{\Psi}_{tgt} = d \tag{6.8}$$

Ψ_{tgt} is found through simply integrating equation (6.8), and the states (x, y, and Ψ) are discretized with a B-spline basis set.

$$x(\tau) = \sum_{i=1}^{n_x} c_{i,x} B_{i,k_x}(\tau, i : i + k_x) \tag{6.9}$$

$$y(\tau) = \sum_{j=1}^{n_y} c_{j,y} B_{j,k_y}(\tau, j : j + k_y) \tag{6.10}$$

$$\Psi(\tau) = \sum_{l=1}^{n_\Psi} c_{l,\Psi} B_{l,k_\Psi}(\tau, l : l + k_\Psi) \tag{6.11}$$

where $\tau = 1/T$, T is the time to complete the trajectory, $k_x = k_y = k_\Psi = 6$ describe the order of the splines, and $n_x = n_y = n_\Psi = 15$ are the number of spline coefficients for each of the variables. The knot sequence for x, y, and Ψ are defined as $S_x = S_y = S_\Psi = \{0^6, \frac{1}{10}, \frac{2}{10}, \ldots, \frac{9}{10}, 1^6\}$ where 0^6 and 1^6 represent six knots at 0 and 1. The b-splines share the same order and knot sequence as those found in [12]. The trajectory can now be described with the following set of variables
$p \equiv [\{c_{i,x}\}_{i=1}^{n_x}, \{c_{j,y}\}_{j=1}^{n_y}, \{c_{l,\Psi}\}_{l=1}^{n_\Psi}, T]$.

The model dynamics are sampled over the set
$S_e = \{\tau_1, \ldots, \tau_n\} = \{0, \frac{1}{30}, \frac{1}{15}, \frac{2}{15}, \ldots, \frac{14}{15}, \frac{29}{30}, \frac{59}{60}\}$, and the equality constraints are de-

fined with the following equations

$$h(p,\alpha) = \begin{pmatrix} \frac{1}{T}x'(\tau_1) - v\cos(\Psi(\tau_1)) + v_{tgt}\cos(\dot{\Psi}_{tgt}\tau_1 T) \\ \vdots \\ \frac{1}{T}x'(\tau_n) - v\cos(\Psi(\tau_n)) + v_{tgt}\cos(\dot{\Psi}_{tgt}\tau_n T) \\ \frac{1}{T}y'(\tau_1) - v\sin(\Psi(\tau_1)) + v_{tgt}\sin(\dot{\Psi}_{tgt}\tau_1 T) \\ \vdots \\ \frac{1}{T}y'(\tau_n) - v\sin(\Psi(\tau_n)) + v_{tgt}\sin(\dot{\Psi}_{tgt}\tau_n T) \end{pmatrix} \tag{6.12}$$

Rather than including the dynamics for $\dot{\Psi}$ in the equality constraints, a limit on $\dot{\Psi}$ is included in the inequality constraints which are described later.

The boundary conditions are defined as

$$h(p,\alpha) = \begin{pmatrix} x(0) - \bar{x}(t_0) \\ y(0) - \bar{y}(t_0) \\ \Psi(0) - \bar{\Psi}(t_0) \\ x(1) - R_s\cos(\dot{\Psi}_{tgt}T + 90^\circ) \\ y(1) - R_s\sin(\dot{\Psi}_{tgt}T + 90^\circ) \\ \Psi(1) - \dot{\Psi}_{tgt}T + 90^\circ \end{pmatrix} \tag{6.13}$$

where $\bar{x}(t_0)$, $\bar{y}(t_0)$, and $\bar{\Psi}(t_0)$ are constants describing the states at the initial time, and R_s is the desired final separation distance between the interceptor and target. The last three constraints dictate the final desired relative position and heading for completing the beam intercept.

The purpose of the inequality constraints is to limit the rate of change of the interceptor's heading. These constraints are sampled over the interval $S_{ineq} = \{s_1, \ldots, s_m\} =$

$\{0, \frac{1}{20}, \ldots, 1\}$ and are defined with

$$g(p) = \begin{pmatrix} \frac{\Psi'(s_1)}{T} - \frac{\tan(\phi_{max})g}{v} \\ \vdots \\ \frac{\Psi'(s_m)}{T} - \frac{\tan(\phi_{max})g}{v} \\ -\frac{\Psi'(s_1)}{T} + \frac{\tan(-\phi_{max})g}{v} \\ \vdots \\ \frac{\Psi'(s_m)}{T} - \frac{\tan(\phi_{max})g}{v} \end{pmatrix} \tag{6.14}$$

6.2 Interpolation Methods

Two methods are applied for entering the parameters into the equality constraints. In both cases, a trajectory must be produced for the desired values of initial relative y position, target speed, and target turn rate. All other variables including the initial target heading, interceptor's speed, initial relative x position, and initial interceptor heading are constant. This method works under the assumption that the interceptor has sufficient time to fly to a state with the above constants before starting the beam intercept.

6.2.1 Single Interpolation (SI)

The first method involves setting the initial relative y position as α and the target speed and target turn rate as a function of α.

$$\alpha = \bar{y}(t_0) \tag{6.15}$$

$$v_{tgt}(\alpha) = a_1 + b_1\alpha + c_1\alpha^2 \tag{6.16}$$

$$\dot{\Psi}_{tgt}(\alpha) = a_2 + b_2\alpha + c_2\alpha^2 \tag{6.17}$$

The coefficients for $v_{tgt}(\alpha)$ and $\dot{\Psi}_{tgt}(\alpha)$ are found using the equations

$$v_{tgt_{v_1}} = a_1 + b_1\alpha_{v_1} + c_1\alpha_{v_1}^2 \tag{6.18}$$

$$v_{tgt_{v_2}} = a_1 + b_1\alpha_{v_2} + c_1\alpha_{v_2}^2 \tag{6.19}$$

$$v_{tgt_{v_d}} = a_1 + b_1\alpha_{v_d} + c_1\alpha_{v_d}^2 \tag{6.20}$$

$$\dot{\Psi}_{tgt_{v_1}} = a_2 + b_2\alpha_{v_1} + c_2\alpha_{v_1}^2 \tag{6.21}$$

$$\dot{\Psi}_{tgt_{v_2}} = a_2 + b_2\alpha_{v_2} + c_2\alpha_{v_2}^2 \tag{6.22}$$

$$\dot{\Psi}_{tgt_{v_d}} = a_2 + b_2\alpha_{v_d} + c_2\alpha_{v_d}^2 \tag{6.23}$$

where $(\alpha_{v_1}, v_{tgt_{v_1}}, \dot{\Psi}_{tgt_{v_1}})$, $(\alpha_{v_2}, v_{tgt_{v_2}}, \dot{\Psi}_{tgt_{v_2}})$, and $(\alpha_{v_d}, v_{tgt_{v_d}}, \dot{\Psi}_{tgt_{v_d}})$ pertain to the trajectories v_1, v_2, and v_d respectively. Now, the algorithm can create new trajectories with varying initial relative y, target speed, and target turn rate by interpolating over one variable. v_1 and v_2 are selected from the candidate trajectory pairs

$$\begin{aligned}
&v_a(\bar{y}_{min}(t_0), v_{tgt_{min}}, \dot{\Psi}_{tgt_{max}}), \quad v_b(\bar{y}_{max}(t_0), v_{tgt_{max}}, \dot{\Psi}_{tgt_{min}}) \\
&v_c(\bar{y}_{min}(t_0), v_{tgt_{min}}, \dot{\Psi}_{tgt_{min}}), \quad v_e(\bar{y}_{max}(t_0), v_{tgt_{max}}, \dot{\Psi}_{tgt_{max}}) \\
&v_f(\bar{y}_{min}(t_0), v_{tgt_{max}}, \dot{\Psi}_{tgt_{max}}), \quad v_g(\bar{y}_{max}(t_0), v_{tgt_{min}}, \dot{\Psi}_{tgt_{min}}) \\
&v_h(\bar{y}_{min}(t_0), v_{tgt_{max}}, \dot{\Psi}_{tgt_{min}}), \quad v_i(\bar{y}_{max}(t_0), v_{tgt_{min}}, \dot{\Psi}_{tgt_{max}})
\end{aligned}$$

which have the smallest value of $|c_1| + |c_2|$ found in equations 6.16 and 6.17, because the method worked best when the functions $v_{tgt}(\alpha)$ and $\dot{\Psi}_{tgt}(\alpha)$ were nearly linear. The trajectories $v_a, v_b, \ldots, v_i$ are produced offline using NLP optimization software to solve for the set of p, subject to the constraints in equations (6.12-6.14) with an objective of minimizing the maneuver time.

6.2.2 Multiple Interpolation (MI)

The second method also uses $v_a, v_b, \ldots, v_i$ but interpolates over each variable in order that a desired trajectory can be reached by performing at most seven interpolations. If we are to represent the sphere of all engagement trajectories by $\bar{y}(t_0)$, v_{tgt} and $\dot{\Psi}_{tgt}$, we can define a cube with tuning boundary parameters. Now assume that the library contains an explicit trajectory solution for each such extreme condition, which

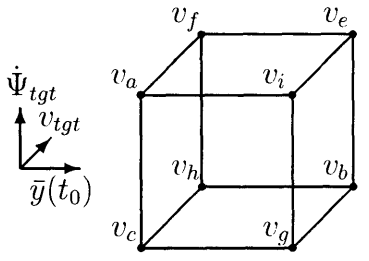


Figure 6-2: Parameter space for trajectory interpolation

is shown as the points on the cube in Figure 6-2. The method is described in Figure 6-3.

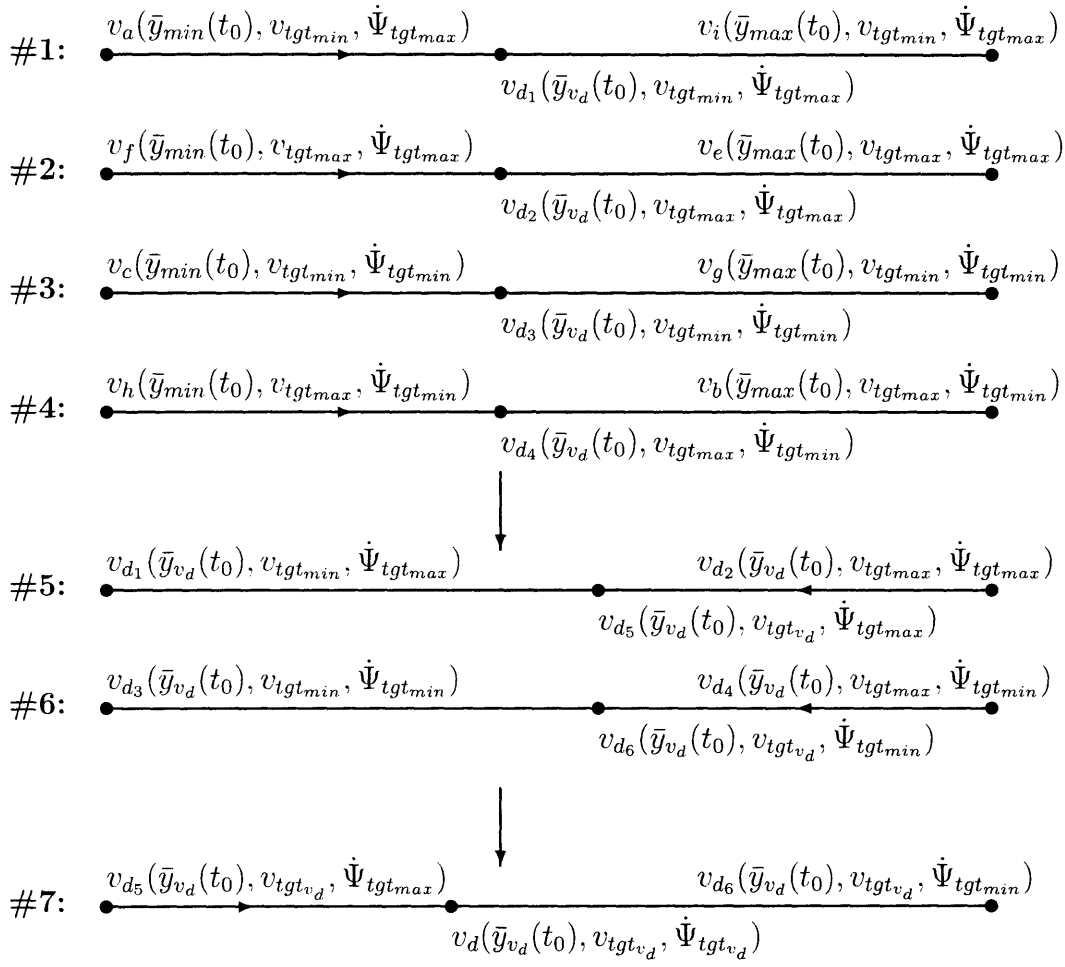


Figure 6-3: The desired trajectory $v_d(\bar{y}_{v_d}(t_0), v_{tgt_{v_d}}, \dot{\Psi}_{tgt_{v_d}})$ is obtained by performing three sets of interpolations. The first set ($\#1 - \#4$) consists of interpolating between the eight given maneuvers to obtain four trajectories which all have $\bar{y}_{v_d}(t_0)$. The second set ($\#5 - \#6$) uses the four new trajectories to create two trajectories which share $\bar{y}_{v_d}(t_0)$ and $v_{tgt_{v_d}}$. The final set ($\#7$) interpolates the previous two trajectories to obtain $\dot{\Psi}_{tgt_{v_d}}$.

6.3 Results

This section is a comparison between the two trajectory interpolation methods and direct collocation. With the single interpolation (SI) method, the desired trajectory is obtained with one interpolation, whereas the multiple interpolation (MI) method could require as many as seven interpolations. The eight trajectories ($v_a, v_b, v_c, v_e, v_f, v_g, v_h$, and v_i) used in this section are defined with the constants

Table 6.1: Constants for trajectory interpolation

$\bar{y}_{min}(t_0)$	$0km$	$\bar{y}_{max}(t_0)$	$3.05km$
$v_{tgt_{min}}$	$152.5m/s$	$v_{tgt_{max}}$	$183m/s$
$\dot{\Psi}_{tgt_{min}}$	$-0.3°/s$	$\dot{\Psi}_{tgt_{max}}$	$0°/s$
$\bar{x}(t_0)$	$27.1km$	v	$152.5m/s$
$\Psi(t_o)$	π	$\Psi_{tgt}(t_o)$	0
ϕ_{max}	$\pi/3$	R_s	$3.1km$

These trajectories vary with initial relative y, target speed and target turn rate in order that trajectory interpolation can be used to obtain a maneuver between $[\bar{y}_{min}(t_0), \bar{y}_{max}(t_0)]$,$[v_{tgt_{min}}, v_{tgt_{max}}]$, and $[\dot{\Psi}_{tgt_{min}}, \dot{\Psi}_{tgt_{max}}]$.

For the first simulation, a trajectory must be found with $\bar{y}_{v_d}(t_0) = 1.53km$, $v_{tgt_{v_d}} = 167.8m/s$, and $\dot{\Psi}_{tgt_{v_d}} = -0.15°/s$. These parameters are the average of the minimum and maximum values for the given trajectories. For the SI method, the trajectories $v_c(\bar{y}_{min}(t_0), v_{tgt_{min}}, \dot{\Psi}_{tgt_{min}})$ and $v_e(\bar{y}_{max}(t_0), v_{tgt_{max}}, \dot{\Psi}_{tgt_{max}})$ are chosen for v_1 and v_2 respectively. Figure 6-4 shows v_1 and v_2 along with a few interpolated trajectories produced in obtaining the desired trajectory v_d.

The trajectories from all three methods are shown in Figure 6-5. The solutions from the two trajectory interpolation methods are nearly the same, while the direct collocation approach achieved a solution that is slightly faster, as shown in Table 6.3.

Figure 6-6 is a comparison in optimality between the direct collocation method and the SI method. Fitting the trajectory v_1 with B-splines resulted in a small time increase in comparison to the direct collocation method. For the desired trajectory v_d the optimal solution produced by direct collocation is more than a second faster.

The SI interpolation approach worked well for various scenarios when c_1 and c_2

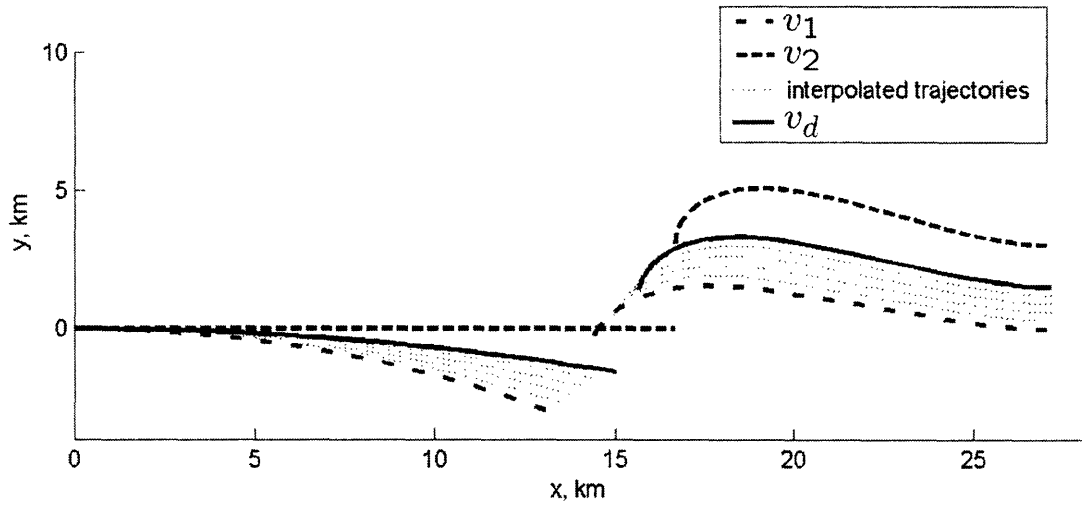


Figure 6-4: The SI method interpolates between $v_1(\bar{y}_{min}(t_0), v_{tgt_{min}}, \dot{\Psi}_{tgt_{min}})$ and $v_2(\bar{y}_{max}(t_0), v_{tgt_{max}}, \dot{\Psi}_{tgt_{max}})$ to produce the desired trajectory $v_d(\bar{y}_{v_d}(t_0), v_{tgt_{v_d}}, \dot{\Psi}_{tgt_{v_d}})$.

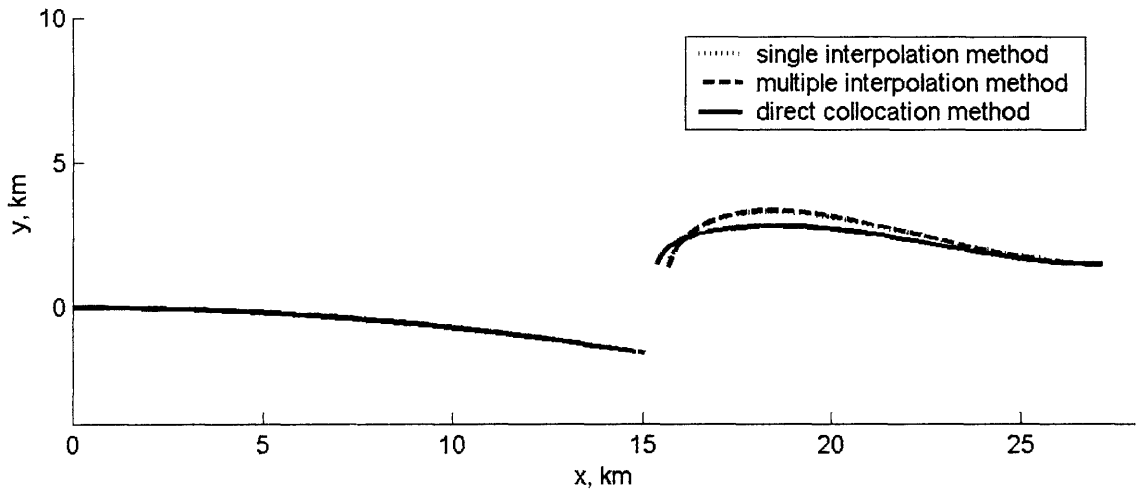


Figure 6-5: Comparison of trajectory interpolation methods and direct collocation for Simulation #1

Table 6.2: Simulation #1

	path length (km)	trajectory time (s)	computation time (s)
single interpolation method	12.60	82.6	89.7
multiple interpolation method	12.63	82.8	435.1
direct collocation	12.41	81.4	16.2

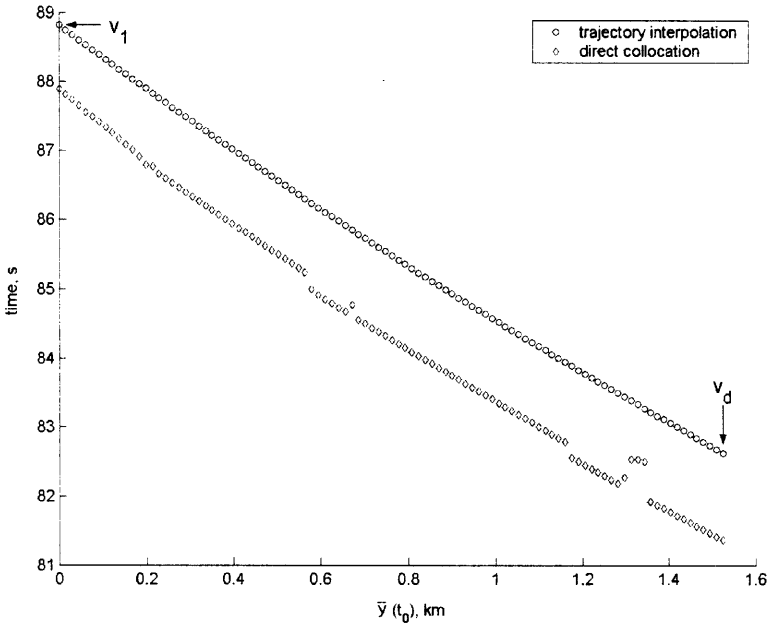


Figure 6-6: Optimality difference between SI method and direct collocation for Simulation #1. The variations in optimality from a smooth curve for the direct collocation method are the result of the method having converged to a local minimum.

are small enough so that the equations describing $v_{tgt}(\alpha)$ and $\dot{\Psi}_{tgt}(\alpha)$ are sufficiently linear to fit within the area described in Figure 6-7. The following simulation is an example where this was not the case.

For the second simulation, the desired trajectory is defined with the parameters $\bar{y}_{v_d}(t_0) = 0.03km$, $v_{tgt_{v_d}} = 165m/s$, and $\dot{\Psi}_{tgt_{v_d}} = -0.11°/s$. The SI method uses the trajectories $v_a(\bar{y}_{min}(t_0), v_{tgt_{min}}, \dot{\Psi}_{tgt_{max}})$ and $v_b(\bar{y}_{max}(t_0), v_{tgt_{max}}, \dot{\Psi}_{tgt_{min}})$ as v_1 and v_2 respectively.

As shown in Figure 6-8, the MI method produces a solution that is slightly slower than the direct collocation method. However, the SI method produces a trajectory that does not appear to be close to an optimal solution.

The SI method is able to maintain feasibility in producing a maneuver with the given parameters. However, the circuitous trajectory is the result of the equation for $v_{tgt}(\alpha)$, which is plotted in Figure 6-9. In order to reach $(\alpha_{v_d}, v_{tgt_{v_d}})$, the method must interpolate along a curve which has a large value of $|c_1|$. If the interpolation were

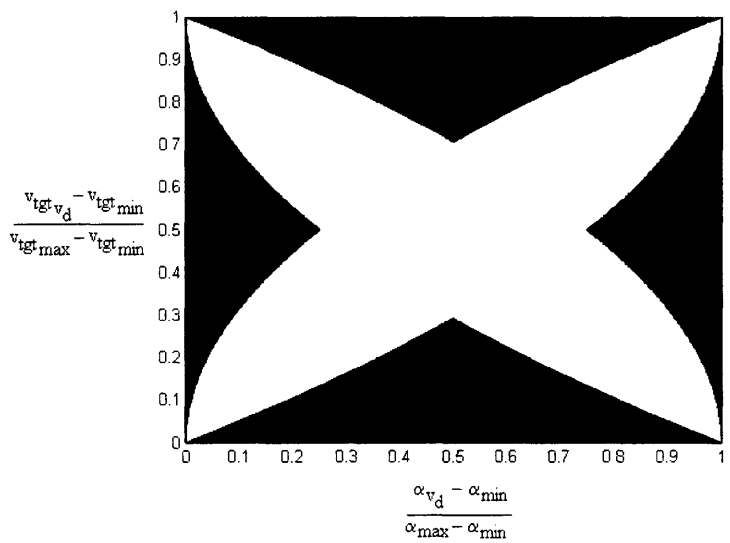


Figure 6-7: The white region of the above contour plot contains the values of $[\alpha_{v_d}, v_{tgt_{v_d}}]$ for which $v_{tgt_{min}} \leq v_{tgt}(\alpha) \leq v_{tgt_{max}}$.

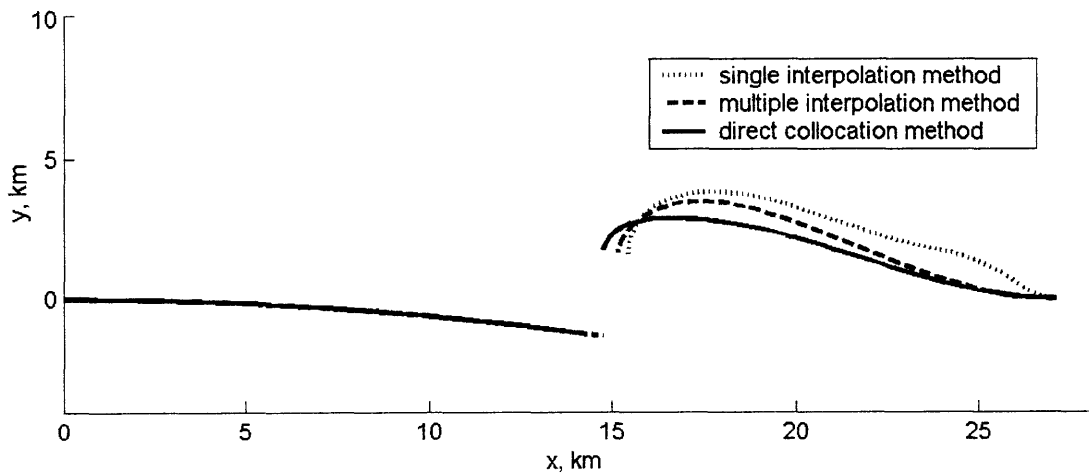


Figure 6-8: Comparison of trajectory interpolation methods and direct collocation for Simulation #2

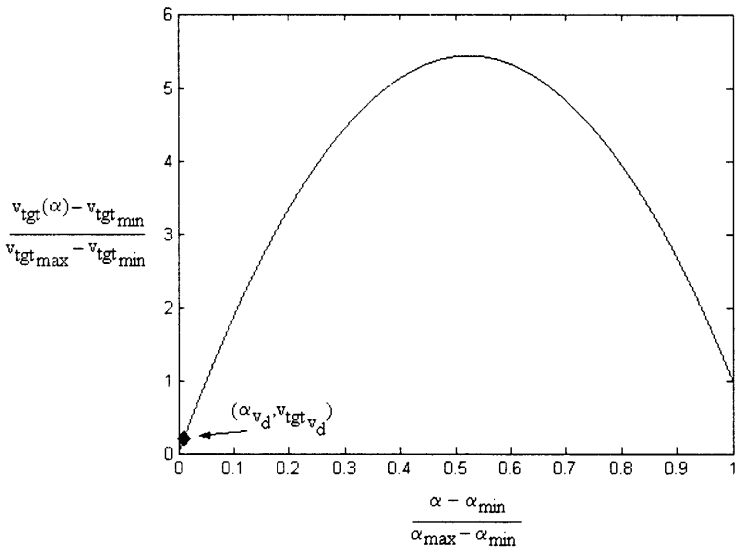


Figure 6-9: Curve describing problem with single interpolation approach

Table 6.3: Simulation #2

	path length (km)	trajectory time (s)	computation time (s)
single interpolation method	13.8	90.7	205.9
multiple interpolation method	13.6	89.1	189.7
direct collocation	13.3	87.2	12.8

continued along the path for $v_{tgt}(\alpha)$ towards v_2, the target's speed would eventually reach a value of at least $1km/s$. At this speed, the interceptor would be unable to perform the desired maneuver. The wobbly trajectory of Figure 6-8 is a precursor of interpolating towards infeasible trajectories.

6.3.1 Discussion of Results

Fortunately, in certain regions where one method struggles, the other excels. For the second simulation, the MI method obtained a solution faster than the SI method despite having to perform seven interpolations. The MI approach is fastest when the desired trajectory has parameters which are close to the given trajectories'.

As shown in the first simulation, the MI method consumes significantly more time,

over 400 seconds, when the desired trajectory has parameters which are furthest away from the given trajectories'. This is exactly where the SI method works well, because the equations for $v_{tgt}(\alpha)$ and $\dot{\Psi}_{tgt}(\alpha)$ are more linear.

In every simulation performed, the computation time for the direct collocation method was significantly faster than the interpolation methods'. Both the SI and MI methods could be improved by increasing the number of available trajectories to be used for interpolation. A denser set of trajectories would increase the region where $v_{tgt_{min}} \leq v_{tgt}(\alpha) \leq v_{tgt_{max}}$ and $\dot{\Psi}_{tgt_{min}} \leq \dot{\Psi}_{tgt}(\alpha) \leq \dot{\Psi}_{tgt_{max}}$. Also, more trajectories would decrease the computation time for the MI method, because the desired parameters might occur closer to a given trajectory's.

Another issue worth considering is the optimality gap between the direct collocation and trajectory interpolation methods. Because there is no guarantee that the converged solutions for $v_a, v_b, \ldots, v_i$ are global optima, better solutions for these trajectories might reduce the maneuver times for the interpolated trajectories.

Chapter 7

Conclusions

This research presents three different approaches for creating trajectories which resemble the fighter pilot maneuver frequently referred to as a beam intercept. The tactic is typically used to visually identify a suspected aircraft, and an ideal trajectory is completed in minimum time while avoiding detection from the other aircraft.

This chapter includes a summary of the thesis and ideas for future work.

7.1 Summary

In Chapter 2, a description of the five phases of air-to-air combat is presented. Technology and tactics have changed throughout history, but these phases are still relevant. However, the importance of each phase possibly has shifted. The maneuver phase or dogfight in recent conflicts did not occur nearly as often as in WWI and WWII. The detection and closing phase remain extremely important. An early detection provides the fighter pilot with an opportunity to make the first decisions. Despite radar technology that can detect certain aircraft hundreds of miles away and missiles which can reach far beyond a pilot's visual range, the closing phase is still essential. A fighter pilot might be forced to merge with an electronically confirmed, enemy aircraft because of the rules of engagement.

Presented with this scenario, the pilot must be able to perform a maneuver to visually identify the aircraft. Four different methods are discussed, and the beam

intercept is usually the maneuver of choice due to its quick execution and stealthy approach.

In Chapter 3, five different trajectory generation methods are considered for creating maneuvers which resemble a beam intercept. Ideally, an analytic solution could be produced. This would allow the problem to be solved in real-time; however, it is unlikely that such a solution exists given the complexity of the radar avoidance constraints. A shooting method approach is also considered but not applied in this thesis due to the difficulty of including path constraints. The three applied methods in this thesis are mixed integer linear programming (MILP), direct collocation, and trajectory interpolation.

In Chapters 4-6, the three chosen methods are applied. The MILP approach has the advantage that a solution can be produced in real-time. However, the complexity of the radar avoidance constraint is limited. Additionally, the aircraft model must be cast in linear form, which might limit the accuracy of an aircraft model with more degrees of freedom such as being able to change altitude.

The direct collocation approach is able to handle the limitations that MILP faces. A more complex radar avoidance constraint and aircraft model are included in this formulation. However, because the problem is solved as a nonlinear program the major advantages with MILP, such as nearly guaranteed real-time convergence, are lost. In many cases, the optimization software is able to converge quickly to what appears to be a global optimum. However, it is the few, faulty cases which make an onboard implementation difficult.

The trajectory interpolation approach is the third method applied. With this formulation, the goal is not to find an optimal trajectory. Rather, the method interpolates between given trajectories with varying boundary conditions and dynamics, obtaining new, feasible maneuvers. The time to reach a solution can be estimated, and the ability of the method to reach feasible maneuvers can be tested offline.

7.2 Future Work

Direct collocation

For the direct collocation simulations, the initial guess for the nonlinear program can affect greatly the converged solution. This was especially true for the variable speed and altitude aircraft model. In these simulations, the method often struggled to reach a feasible solution. Even when the method converged, it is questionable whether or not the produced trajectory is a global optimum.

A potential solution to this problem is to use a genetic algorithm approach to produce initial guesses. Used alone, genetic algorithms are not competitive with gradient methods for producing an optimal solution [3]. However, in [37] the authors found that combining genetic algorithms with direct collocation can improve the chances of converging to the global optimum. The idea is that the genetic algorithm, while not typically able to find a locally optimal solution, is able to search globally the set of feasible solutions. The best solution of the genetic algorithm is then applied as the initial guess for the nonlinear program. This combined approach may yield better solutions for beam intercept trajectories.

Trajectory interpolation

The trajectory interpolation approach could also be explored further by adding radar avoidance constraints and a more complicated aircraft model. With additional constraints and variables, it would be interesting to note if the method can still obtain feasible trajectories.

Another aspect of trajectory interpolation to be explored is the computation time in forming new trajectories. In the presented results, the method was usually significantly slower than direct collocation. Increasing the number of trajectories calculated offline, in order that the method does not have to interpolate as far, is one potential solution. Defining how many trajectories is appropriate for a given scenario could be an interesting topic of research.

Intelligent target

For all of the approaches presented, the trajectory of the target aircraft was limited to trim flight in steady level or a constant turn rate. Future research could include a more sophisticated target which would take precautions to avoid a beam intercept. For example, the target could respond defensively or offensively when the interceptor violates the radar range constraint. It could also intelligently vary its heading so that its radar cone does not produce constant blind spots.

Multiple Aircraft

Radar blind spots can also be avoided by flying in formation. It is unlikely that the target aircraft will always fly alone. An ideal trajectory would include constraints considering the radar of all present aircraft. Additionally, the method should include which aircraft to pursue first.

The interceptor will most likely fly with support aircraft as well. The problem could include additional friendly aircraft which would be able to assist in performing visual identifications. The MILP approach can easily handle more aircraft. However, this might be difficult to include with the direct collocation and trajectory interpolation approaches.

Additional constraints

Another potential problem is no fly zone areas. For instance, an optimal trajectory should avoid areas densely populated with surface-to-air missile sites. If these constraints are included, the relative frame formulation used for the direct collocation and trajectory interpolation approaches may be inappropriate. An inertial reference frame would facilitate the formulation of these constraints.

Appendix A

Collocation Example

The following equations are an example of the collocation method described in section 5.1.1. The system dynamics are the same as section 5.2.1

$$\dot{x}_{rel} = v\cos\Psi - v_{tgt}\cos\Psi_{tgt} \tag{A.1}$$

$$\dot{y}_{rel} = v\sin\Psi - v_{tgt}\sin\Psi_{tgt} \tag{A.2}$$

$$\dot{\Psi} = \frac{\tan(\phi)g}{v} \tag{A.3}$$

$$\dot{\Psi}_{tgt} = d \tag{A.4}$$

The problem is discretized with $N+1$ nodes so that the NLP is formulated with the variables $x_{rel}(k)$, $y_{rel}(k)$, $\Psi(k)$, $\phi(k)$, and t_f where $\forall k \in [0 \ldots N]$.

The defects are calculated using the equations from section 5.1.1

$$x_c = (x_1 + x_2)/2 + T(f_1 - f_2)/8 \tag{A.5}$$

$$x_c' = -3(x_1 - x_2)/2T - (f_1 + f_2)/4 \tag{A.6}$$

$$\Delta = f_c - x_c' \tag{A.7}$$

where $T = \frac{t_f}{N}$.

The first step in forming the defects for this problem is to calculate the interpolated values of the states and controls which are expressed in the dynamics. In this case, they are Ψ and ϕ

$$\Psi_c(k) = \frac{\Psi(k)+\Psi(k+1)}{2} + T\left(\frac{\tan\phi(k)g}{v} - \frac{\tan\phi(k+1)g}{v}\right)/8 \quad \text{(A.8)}$$

$$\phi_c(k) = \frac{\phi(k)+\phi(k+1)}{2} \quad \text{(A.9)}$$

$$\forall \quad k \in [0 \ldots N-1]$$

The next step is to calculate the actual defects for each of the dynamics equations. Collocation is not required for the target's heading because equation (A.4) can be integrated easily.

$$f_{c,\dot{x}_{rel}}(k) = v\cos\Psi_c(k) - v_{tgt}\cos\Psi_{tgt} \quad \text{(A.10)}$$

$$\begin{aligned} x'_{c,\dot{x}_{rel}}(k) &= -3(x_{rel}(k) - x_{rel}(k+1))/2T + \\ &\quad -[(v\cos\Psi(k) - v_{tgt}\cos\Psi_{tgt}) + \\ &\quad (v\cos\Psi(k+1) - v_{tgt}\cos\Psi_{tgt})]/4 \end{aligned} \quad \text{(A.11)}$$

$$\Delta_{k,1} = f_{c,\dot{x}_{rel}}(k) - x'_{c,\dot{x}_{rel}}(k) \quad \text{(A.12)}$$

$$\forall \quad k \in [0 \ldots N-1]$$

$$f_{c,\dot{y}_{rel}}(k) = v\sin\Psi_c(k) - v_{tgt}\sin\Psi_{tgt} \quad \text{(A.13)}$$

$$\begin{aligned} x'_{c,\dot{y}_{rel}}(k) &= -3(y_{rel}(k) - y_{rel}(k+1))/2T + \\ &\quad -[(v\sin\Psi(k) - v_{tgt}\sin\Psi_{tgt}) + \\ &\quad (v\sin\Psi(k+1) - v_{tgt}\sin\Psi_{tgt})]/4 \end{aligned} \quad \text{(A.14)}$$

$$\Delta_{k,2} = f_{c,\dot{y}_{rel}}(k) - x'_{c,\dot{y}_{rel}}(k) \quad \text{(A.15)}$$

$$\forall \quad k \in [0 \ldots N-1]$$

$$f_{c.\dot{\Psi}(k)} = \frac{\tan\phi_c(k)g}{v} \tag{A.16}$$

$$x'_{c.\dot{\Psi}}(k) = -3(\Psi(k) - \Psi(k+1))/2T + \\ -\left[\left(\frac{\tan\phi(k)g}{v}\right) + \left(\frac{\tan\phi(k+1)g}{v}\right)\right]/4 \tag{A.17}$$

$$\Delta_{k,3} = f_{c.\dot{\Psi}}(k) - x'_{c.\dot{\Psi}}(k) \tag{A.18}$$

$$\forall \quad k \in [0 \ldots N-1]$$

Now, the dynamics of the system are met approximately when equations A.12, A.15, and A.18 are equal to zero.

[This page intentionally left blank.]

Bibliography

[1] Bellingham, John, Arthur Richards, and Jonathan P. How, "Receding Horizon Control of Autonomous Aerial Vehicles", *American Control Conference*, Anchorage, AK, May 2002.

[2] Ben-Asher, Joseph, Z. "Optimal Trajectories for an Unmanned Air-Vehicle in the Horizontal Plane". Journal of Aircraft, Vol.32, No.3, 1995, pp. 677-680.

[3] Betts, John T., "Survey of Numerical Methods for Trajectory Optimization", Journal of Guidance, Control, and Dynamics, Vol.21, No.2, 1998, pp. 193-207.

[4] Bocvarov, Spiro, Frederick H. Lutze, Eugene M. Cliff, "Time-Optimal Reorientation Maneuvers for a Combat Aircraft", Journal of Guidance, Control, and Dynamics, Vol.16, No.2, 1993, pp. 234-240.

[5] Brown, David R., "Unmanned Combat Aerial Vehicles: Evolution or Potential Revolution?", Air Command and Staff College, Maxwell Air Force Base, AL, 1998.

[6] Bryson, Arthur E., *Dynamic Optimization*, Addison Wesley Longman, Menlo Park, CA, 1999.

[7] Bryson, A. E., and Denham, W. F., "A Steepest-Ascent Method for Solving Optimum Programming Problems", Journal of Applied Mechanics, Vol.29, 1962.

[8] Clements, John C., "Minimum-Time Turn Trajectories to Fly-to-Points", Optimal Control Applications & Methods, Vol.11, 1990, pp. 39-50.

[9] Clements, John C., "Real-Time Optimal Control of Aircraft Turn Trajectories", Journal of Guidance, Vol.16, No.2, 1992, pp. 385-387.

[10] Cohen, Eliot A., Emery M. Kiraly, Robert Kelley, et al. *Gulf War Air Power Survey*, Vol. 2-4, U.S. Government Printing Office, Washington DC, 1993.

[11] Cry, Joe, "Radar Acronyms", URL: http://www.radomes.org/museum/scripts/jargon.cgi [cited 22 March 2005].

[12] Dever, Chris, Bernard Mettler, Eric Feron, Jovan Popović, and Marc McConley, "Trajectory Interpolation for Parameterized Maneuvering and Flexible Motion Planning of Autonomous Vehicles", *Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit*, Providence, RI, August 2004.

[13] Dever, Christopher, "Parametrized Maneuvers for Autonomous Vehicles", Doctoral Thesis, Massachusetts Institute of Technology, Department of Mechanical Engineering, CSDL-T-1498, 2004.

[14] Erzberger, Heinz, and Homer Q. Lee, "Optimum Horizontal Guidance Techniques for Aircraft", Journal of Aircraft, Vol.8, No.2, 1971, pp. 95-101.

[15] Grimm, Werner, and Markus Hans, "Time-Optimal Turn to a Heading: an Improved Analytic Solution", Journal of Guidance, Control, and Dynamics, Vol.21, No.6, 1998, pp. 940-947.

[16] Hargraves, C.R., and S.W. Paris,"Direct Trajectory Optimization Using Nonlinear Programming and Collocation", AIAA Journal of Guidance, Vol.10, No.4, 1987, pp. 338-342.

[17] Hermann, Albert L., and Bruce A. Conway, "Direct Optimization Using Collocation Based on High-Order Gauss-Lobatto Quadrature Rules", Journal of Guidance, Control, and Dynamics, Vol.19, No.3, 1996, pp. 592-599

[18] Horie, Kazuhiro, and Bruce A. Conway, "Optimization for Fighter Aircraft Vertical-Plane Maneuvering Using Postall Flight", Journal of Aircraft, Vol.37, No.6, 2000, pp. 1017-1021.

[19] Lambeth, B., *The Transformation of American Air Power*, Cornell University Press, Ithica, NY, 2000.

[20] Lewis, William, "UCAV the Next Generation Air-Superiority Fighter?" Master Dissertation. Air University, School of Advanced Airpower Studies, June 2002.

[21] Nolan, Robert, "The Pilotless Air Force? A Look at Replacing Human Operators with Advanced Technology", Air Command and Staff College, Maxwell Air Force Base, AL, 1997.

[22] Norsell, Martin, "Aircaft Trajectory Optimization with Tactical Constraints", Doctoral Thesis, KTH Aeronautical and Vehicle Engineering, Stockholm, Sweden, 2004.

[23] Norsell, Martin, "Radar Cross Section Constraints in Flight Path Optimization", *Proceedings of 41st Aerospace Sciences Meeting and Exhibit*, Reno, NV, January 2003.

[24] Repp, D., "History of 641^{st} AC&W Squadron", URL: http:// www.pinetreeline.org/other/other19/other19r.html [cited 22 March 2005].

[25] Richards, Arthur, and Jonathan P. How, "Aircraft Trajectory Planning with Collision Avoidance Using Mixed Integer Linear Programming", *American Control Conference*, Anchorage, AK, May, 2002.

[26] Richards, Arthur, Yoshiaki Kuwata, and Jonathan How, "Experimental Demonstrations of Real-time MILP Control", *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Providence, RI, August, 2004.

[27] Ringertz, Ulf, "Flight Testing an Optimal Trajectory for the Saab J35 Draken", Journal of Aircraft, Vol.37, No.1, 1999, pp. 187-189.

[28] Ringertz, Ulf, "Optimal Trajectory for a Minimum Fuel Turn", Journal of Aircraft, Vol.37, No.5, 2000, pp. 932-934.

[29] Schouwenaars, Tom, Bart De Moor, Eric Feron, and Jonathan How, "Mixed Integer Programming for Multi-Vehicle Path Planning", *European Control Conference* Porto, Portugal, September 2001.

[30] Seywald, Hans, Eugene M. Cliff, and Klaus H. Well, "Range Optimization for a Supersonic Aircraft", AIAA-91-2712-CP, 1991.

[31] Shapira, Ilana, and Joseph Z. Ben-Asher, "Near-Optimal Horizontal Trajectories for Autonomous Air Vehicles", Journal of Guidance, Control, and Dynamics, Vol.20, No.4, 1997, pp. 735-741.

[32] Shaw, Robert, *Fighter Combat.* Naval Institute Press, Annapolis, MD, 1985.

[33] Spick, Mike, *Fighter Pilot Tactics: The Techniques of Daylight Air Combat*, Stein and Day, New York, NY, 1983.

[34] Stimson, George, W. *Introduction to Airborne Radar*, Second Edition, SciTech Publishing, Mendham, NJ, 1998.

[35] Walden, Rainer, "Time-Optimal Turn to a Heading: an Analytic Solution", Journal of Guidance, Vol.17, No.4, 1993, pp. 873-875.

[36] *Xpress-MP Essentials User's Guide*, Dash Optimization, 2002.

[37] Yokoyama, Nobuhiro, and Shinji Suzuki, "Modified Genetic Algorithm for Constrained Trajectory Optimization", Journal of Guidance, Control, and Dynamics, Vol. 28, No.1, 2005, pp. 139-144.

[38] Zarchan, Paul, *Tactical and Strategic Missile Guidance*, Second Edition, American Institute of Aeronautics and Astronautics, Washington, DC, 1994.