New Development on Sense and Avoid Strategies for Unmanned Aerial Vehicles

Yu Fu

A Thesis
in
The Department
of
Mechanical and Industrial Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of
Master of Applied Science (Mechanical Engineering) at
Concordia University
Montréal, Québec, Canada

February 2016

© Yu Fu, 2016

CONCORDIA UNIVERSITY
School of Graduate Studies

This is to certify that the thesis prepared

By: **Yu Fu**

Entitled: **New Development on Sense and Avoid Strategies for Unmanned Aerial Vehicles**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science (Mechanical Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

_____ Chair
Dr. Chun-Yi Su

_____ External Examiner
Dr. Zhenhua Zhu

_____ Examiner
Dr. Subhash Rakheja

_____ Supervisor
Dr. Youmin Zhang

Approved by _____
Martin D. Pugh, Chair
Department of Mechanical and Industrial Engineering

_____ 2016

_____
Amir Asif, Dean
Faculty of Engineering and Computer Science

Abstract

New Development on Sense and Avoid Strategies for Unmanned Aerial Vehicles

Yu Fu

Unmanned Aerial Vehicles (UAVs) can carry out more complex civilian and military applications with less cost and more flexibility in comparison of manned aircraft. Mid-air collision thus becomes profoundly important considering the safe operation of air transportation systems, when UAVs are increasingly used more with various applications and share the same airspace with manned air vehicles. To ensure safe flights, UAVs have to configure Sense and Avoid (S&A) systems performing necessary maneuvers to avoid collisions. After analyzing the manner of S&A system, avoidance strategies based on a subset of possible collision scenarios are proposed in this thesis. 1) To avoid a face-to-face intruder, a feasible trajectory is generated by differential geometric guidance, where the constraints of UAV dynamics are considered. 2) The Biogeography Based Optimization (BBO) approach is exploited to generate an optimal trajectory to avoid multiple intruders' threats in the landing phase. 3) By formulating the collision avoidance problem within a Markov Decision Process (MDP) framework, a desired trajectory is produced to avoid multiple intruders in the 2D plane. 4) MDP optimization method is extended to address the problem of optimal 3D conflict resolution involving multiple aircraft. 5) Considering that the safety of UAVs is directly related to the dynamic constraints, the differential flatness technique is developed to smoothen the optimal trajectory. 6) Energy based controller is designed such that the UAV is capable of following the generated trajectory.

Acknowledgments

Firstly, I would like to express my sincere gratitude to my supervisor, Dr. Youmin Zhang, for professional supervision and his scientific assistance, constant help, advice, guidance and good mood throughout the years.

I would like to express my sincere thanks to Dr. Xiang Yu, for his enthusiasm for research and challenging projects. His continuous support and very helpful inputs eased the hardship that faced me throughout flight simulation and paper writing. I am grateful that I had the opportunity to work with and learn so much from him.

I would also like to acknowledge and thank Puthy Soupin, Phil Cole at the Marinvent Corporation, who were helpful in giving their experience and advice that were useful in achieving cooperated project. I also wish to thank all members of lab for generously sharing their years of experience.

Last but not least, I would like to thank my loving and caring family. My deepest gratitude goes to my beloved parents for their endless love, prayers and encouragement. It would not have been possible for me to reach where I am without their support, patience and care. I would also like to express my appreciation to my sister and friends who were always supporting and encouraging me with their best wishes.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Unmanned Aerial Vehicles (UAVs) become promising in the civilian and military applications due to the less cost and more flexibility in comparison of manned aircraft. However, the growing diversity of flight makes UAVs vulnerable to mid-air collisions. To guarantee the safety for manned aircraft, human pilots cooperated with Detect and Avoid (D&A) (Hottman, Hansen, & Berry, 2009) systems are responsible of detecting intruders in airspace and performing necessary maneuvers to avoid collisions. Unlike manned aircraft with human pilots involved, UAVs have to equip with Sense and Avoid (S&A) systems to guarantee the flight safety (Yu & Zhang, 2015). Thus, Sense and Avoid systems play an important role in merging UAVs into the National Airspace System (NAS).

As illustrated in Fig. 1.1, a S&A scheme is composed by four units, which are sensing, conflict detection, collision avoidance, and flight controller, respectively. The sensing technologies to provide traffic information of the surrounding environment for UAVs rely on existing infrastructure, such as cooperative sensors and non-cooperative sensors. Based on the sensed data, the conflict detection function extracts useful information and projects the states into the future to determine whether a potential conflict will occur or not. The conflict is an event in which two or more aircraft break the defined minimal separation criterion. The criterion of the en-route horizontal separation in civilian air traffic is 5 Nautical Miles and of the vertical separation is 1000 ft. When a potential collision is predicted to take place, the parameters of the conflict (position of predicted collision and time of possible conflicts) are transferred to the collision avoidance unit. This unit is applied to generate trajectories in order to resolve multiple collisions without incurring too much cost. According

to the re-planned trajectory and UAV dynamics, the flight controller sends outputs the control signals to drive the engine and the control surfaces correspondingly so that avoidance maneuvers can be accomplished physically.

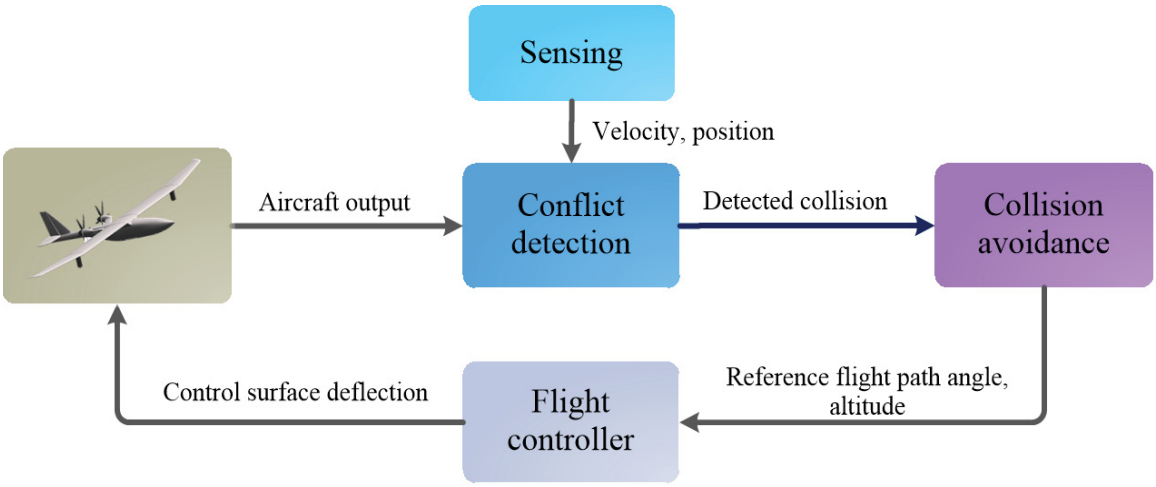


Figure 1.1: The illustrative scheme of S&A functionality in a UAV.

1.2 Challenges

A large amount of collision avoidance approaches have been proposed with application to UAVs in the literature. However, there are several challenges in the existing literature:

- Collision avoidance systems rely on sensor measurements to detect intruders. However, most sensor have measurement noises which result in uncertainties of detected positions and the estimated velocity of threat aircraft. Because collision avoidance systems are safety-critical, taking into account state uncertainty can be very important.
- One of the most practical factors in real flights is that the dynamic constraints have to be respected. In practice, there are allowable bounds related to the control inputs and system states (e.g. control surface deflections, pitch rate). To ensure a safe flight, the produced trajectory should follow the rules without violating these limits.
- Traditional heuristic algorithms are easily trapped into a local minimum after updating the solution. The generated trajectory may not be the optimal one due to the constraints of the algorithms.
- An investigation of Air Safety Foundation (ASF) reveals that 45% of collisions occur in the traffic pattern, and more than 67% of these collisions occur during approach

and landing when aircraft are on final or over the runway (Marsh, 2015). Moreover, many civilian UAV applications are performed at low altitude airspace, increasing the possibility of conflicting with other aircraft in a landing period. To ensure a safe and successful landing, it is desirable that the S&A systems perform necessary maneuvers to avoid potential collisions with intruder aircraft/UAVs.

1.3 Contributions

The contributions of this thesis are summarized below:

- In Chapter 4, the contributions include: 1) At the stage of collision detection, Closest Point of Approach (CPA) is exploited to predict the collision using the geometry of relative kinematics between the UAV and intruder. The probability of collision is introduced to the conflict detection approach, considering the sensor measurement noises; 2) A Differential Geometric Guidance (DGG) based conflict resolution algorithm is proposed to calculate the reference of heading angle to avoid face-to-face intruder; 3) The flatness based approach is used to re-plan routes with consideration of the physical constraints of the UAV; 4) The developed S&A algorithm is validated through extensive tests in the Flight Simulator developed by Marinvent Corporation.
- In Chapter 5, the contributions include: 1) Considering the safety requirement for landing, which is identified as the most complex and challenging period for UAV flight and the S&A mission, Biogeography-Based Optimization (BBO) approach is employed to produce an optimal solution for avoiding multiple collisions; 2) With respect to the path planner, the constraints of UAV are adopted in the differential flatness algorithm to smoothen the planned trajectory such that the generated trajectory is feasible for the UAV; 3) Integrating passivity based control technique with singular perturbation ideas, an energy-based controller is designed to ensure that the UAV can follow the planned path to complete the collision avoidance.
- In Chapter 6, the contributions include: 1) Markov Decision Process (MDP) based methods are presented to model multiple threats avoidance problem in 2D or 3D environment which are appropriate for real-time applications; 2) Taking the constraints of UAV into consideration when planning a path, the differential flatness-based algorithm smooths the planned path; 3) Considering the requirements of stabilization of the lateral dynamics and the longitudinal dynamics, the decoupled controller including a lateral controller and a longitudinal controller is proposed to ensure the attitude tracking of UAV.

1.4 Organization of the Thesis

This thesis consists of seven chapters including the introduction chapter. The remainder of this thesis is structured as follows:

- Chapter 2 provides a literature review of S&A technologies in the sequence of fundamental functions of S&A in sensor techniques, collision detection, and collision avoidance.
- The UAV model, the problem and objective function for collision avoidance are stated in Chapter 3.
- Chapter 4 presents a conflict detection and path planning algorithm for UAV that is developed for S&A applications based on DGG and flatness methods.
- Chapter 5 proposes a BBO based algorithm for UAV avoiding multiple intruders threats in the landing phase.
- Chapter 6 investigates the avoidance strategy for different conflict scenarios in which UAV should change direction in two-dimensional or three-dimensional plane. The collision avoidance problems are formulated by a MDP framework to produce desired trajectories in real-time.
- Concluding remarks are drawn in Chapter 7.

Chapter 2

Literature Review

2.1 Survey on Sensors

The sensing system provides traffic information about surrounding environment for unmanned aircraft system. Depending on the different transmitted ways, the existing sensor technologies can be basically divided into two categories: cooperative and non-cooperative methods.

2.1.1 Cooperative Sensors

For cooperative sensors, it is required that UAVs in the airspace equip the cooperative sensors so as to exchange information between the cooperative UAVs like position, attitude, velocity and trajectory. The (TCAS) is known as the primary cooperative collision avoidance system and has been used by a variety of airspace users (Billingsley, Kochenderfer, & Chryssanthacopoulos, 2012). TCAS transmits information with a transponder during the detection period. Equipped with a TCAS, an aircraft is capable of communicating with other aircraft which also adopted TCAS. Considering the weight restrictions, small UAV may not directly deploy TCAS. The communication between manned and unmanned aircraft becomes a challenging issue.

Automatic Dependent Surveillance-Broadcast (ADS-B) considered as an element of the US Next Generation Air Transportation System (Valovage, 2007) is capable of broadcasting the identification, position, velocity, and intent of the aircraft with a data-link to a transceiver. Ground stations and other aircraft within a 241 km radius can receive these data-links. ADS-B has several advantages, including that 1) it can provide more accurate and reliable traffic information, which is updated on a near-constant basis; 2) it allows for

rapid and effective search and rescue operations; 3) flexible software structure deployed in ADS-B ensures an ability for adaptation and incorporation of future technologies.

2.1.2 Non-Cooperative Sensors

For UAVs without communication sensors, non-cooperative sensors are used to detect airborne objects. These technologies include radar, Laser/Light Detection and Ranging (LIDAR), Electro-Optical (EO) system, acoustic system, and Infrared (IR) sensor (Marshall, Barnhart, Shappee, & Most, 2015). Compared to TCAS and ADS-B, the non-cooperative technologies can be developed to detect ground obstacles.

Using multiple radar pulses, Synthetic Aperture Radar (SAR) generates an image of the object applied for detecting motion and calculating velocity, location, and size of ground targets (Lentilhac, 2010). Equipped with a light weight (230 g) and X-band (10.5 GHz), a Doppler radar system is developed to detect and identify a UAV (Moses, Rutherford, & Valavanis, 2011). A L-band SAR with a range bandwidth of 80 MHz and effective distance of 16 km is applied for UAV (Rosen et al., 2007). The General Atomics Predator B radar is capable of covering a total of 220° in azimuth and 30° in elevation (Owen, Duffy, & Edwards, 2014).

LIDAR measures distance by illuminating a target with a laser and analyzing the reflected light. Geyer, Dey, and Singh (2009) designed a LIDAR system to reduce the false positive rate and measure the range of the intruders in Robotics Institute. A LIDAR system equipped with the Ibeo Lux laser scanner (Lin, Hyyppä, & Jaakkola, 2011) is deployed to estimate tree height and refine digital terrain model. The detection range of LIDAR systems is variable from 200 m to 3 km. The advantages of LIDAR include high resolution and it is appropriate for different atmospheric conditions.

EO systems, as passive technologies, are developed to detect intruders by virtue of natural visible light. Global Hawk equips with an EO-based system to detect and avoid intruders (McCalmont, Utt, Deschenes, & Taylor, 2005). Taking account of altitude and velocity of head-on collisions, Minwalla, Thomas, Ellis, Hornsey, and Jennings (2012) designed a prototype optical instrument. The benefit of EO systems lies in low cost, mass, and power requirements. In J. Kim et al. (2010), the EO system which consists of a transmission system, gimbals controller, and image acquisition system has ability to track images and take 3D measurement of a target as well as acquire high quality images.

Following the character of signal emitted from a propeller-driven aircraft, acoustic sensors are developed for S&A capability. By exploiting the sound of engines, the Passive

Acoustic Non-Cooperative Collision Alert System (PANCAS) is designed to detect aircraft (Milkie, 2007). Acoustic arrays comprised 4 microphones (Finn & Franklin, 2011) are capable of locating and tracking aircraft. The main advantage of acoustic sensors is that detecting all frequencies and angles within a wide-open range.

Infrared sensors detect potential aircraft by measuring the infrared light radiated from objects. Equipped with low-resolution infrared cameras, Osborne III, Bar-Shalom, Willett, and Baker (2011) presented a Passive Collision Warning System (PCWS). Medium wave infrared sensor on-board the aircraft is developed to detect runway and obstacles during approach for landing (Satish, Sudesh, & Shantha Kumar, 2014). Without the requirement of light, infrared sensors is appropriate for night-time usage.

2.1.3 Summary

The traffic environment information around aircraft is collected by sensors. Table. 2.1 (Yu & Zhang, 2015) briefly described the advantages and disadvantages of the relative sensing devices. The cooperative sensors provide the ability to sense the environment and to communicate with aircrafts equipped with the same type of sensors by establishing a communication link. The cooperative sensors can be able to accurately provide the information such as the range, the azimuth and the elevation, but the cost is high. For the non-cooperative sensors, it is not required the coordination with other similarly equipped aircraft. The advantages of non-cooperative sensors are easily configured, the low cost, and the ability to detect moving and stationary obstacles. However, the non-cooperative sensing techniques usually operate a much shorted distance than cooperative sensors. Distant objects can not be correctly detected. Based on the review of cooperative and non-cooperative sensors, the detection ranges of proposed sensors are compared in Fig. 2.1.

Table 2.1: The characteristics of sensor technologies

Sensor	Information Provided	Size	Weight	Power	Cost
TCAS	Range, Altitude	×	×	×	×
ADS-B	Position, Altitude, Velocity	×	×	×	×
Radar	Range, Bearing	×	×	×	×
EO System	Azimuth, Elevation	√	√	√	√
Acoustic System	Azimuth, Elevation	√	√	√	√
Infrared Sensor	Azimuth, Elevation	√	√	√	√
LIDAR	Range	×	×	×	×

*Note: √: applicable; ×: not applicable

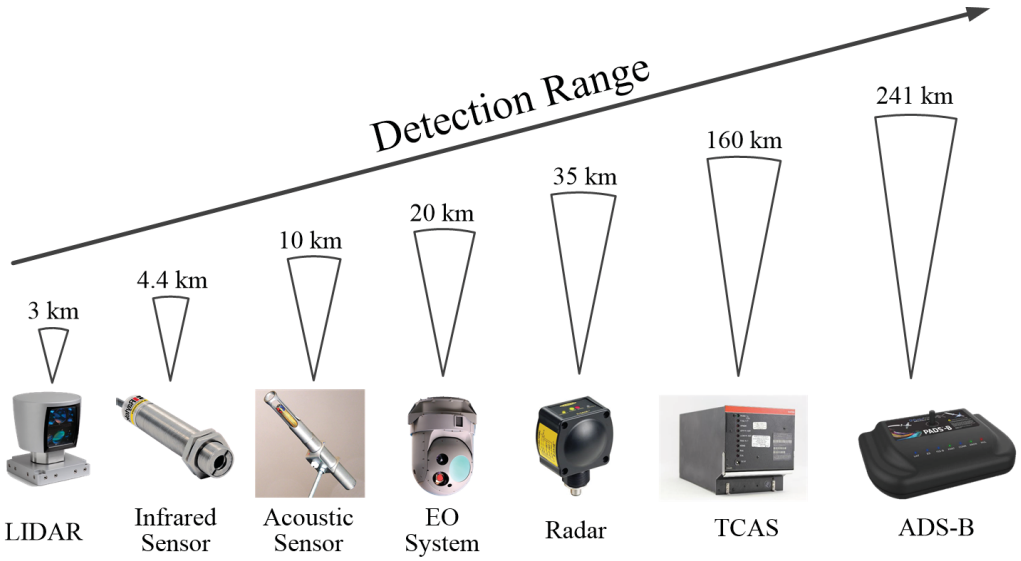


Figure 2.1: Illustration of typical sensing devices.

2.2 Survey on Collision Detection Approaches

According to the data sources from sensing devices, the data process has been developed to estimate motion and project the states into the future in order to predict whether a conflict will occur. The dimension of sensing environment can be categorized into 2D horizontal plane, 2D vertical plane, and 3D state information, respectively. Burgess, Altman,

and Wood (1994) proposed a resolution advisory through estimating the 2D horizontal miss distance at closest approach. Using a radar altimeter to generate 2D vertical data, a basic Ground Proximity Warning System (GPWS) on-board an aircraft provides distinctive aural warning to a flight crew when the aircraft is in close proximity to ground terrain (Breen, 1999). A 3D geometric approach is adopted in (Carbone, Ciniglio, Corraro, & Luongo, 2006) to determine the potential conflicts for non-cooperative aircraft.

The extrapolation method as the main component of collision detection is developed to project the current states of UAV and encounter into the near future. Three extrapolation methods have been identified as nominal method, worst case, and probabilistic. The three methods are shown in Fig. 2.2.

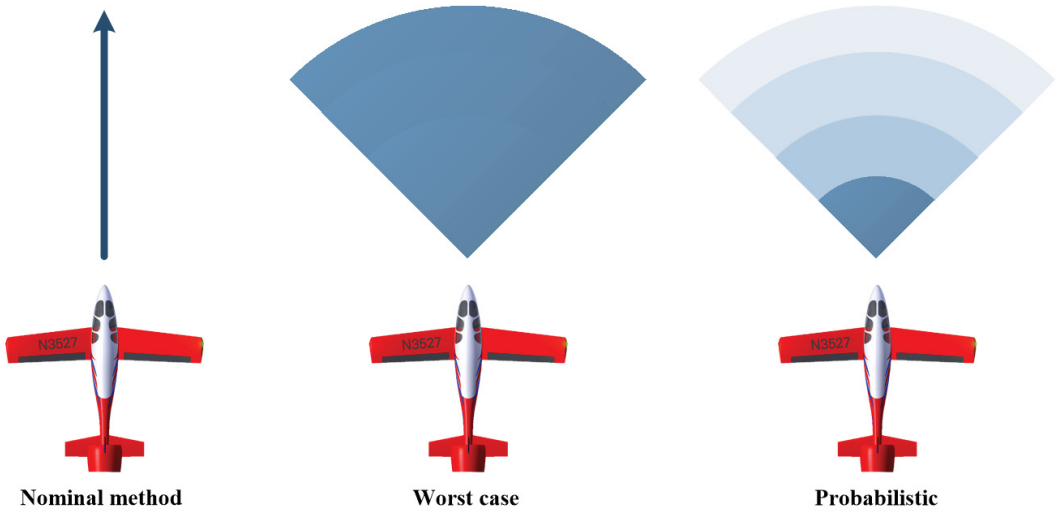


Figure 2.2: Collision detection method.

In the nominal method, the future trajectory is directly generated through current states of UAV without directly considering the possibility of any uncertainty or change (Kuchar & Yang, 2000). In situations where the trajectory of aircraft can be predictable, the nominal method is accurate and straightforward to be deployed. Nominal projection is based on the assumption that aircraft will behave as expectation.

The other method is the worst case projection which assumes that an aircraft will perform in any range of maneuvers. A conflict will be predicted when any one of these maneuvers could cause a conflict. The worst case projection can produce a swath of potential trajectories to be monitored whether a conflict will take place. Wollkind, Valasek, and Ioerger (2004) deployed non-cooperative game theoretic angle to solve for solutions that work in worst case scenarios.

In the probabilistic method, uncertainties are modeled to describe potential variations in the future trajectory of the aircraft. To construct this model, all possible trajectories are generated and each trajectory is evaluated by the probability function. The probabilistic method is the most general one as a trade-off between the nominal method and the worst case prediction. For the probabilistic conflict detection, UAV is modeled as a point mass with uncertain states that are not allowed to enter a known conflict zone. Wang, Hussein, and Erwin (2011) computed the conflict probability for complex aircraft actions in cluttered environment, using upper bound calculation and assumption of Gaussian distributed aircraft states.

2.3 Survey on Collision Avoidance Approaches

With the information of intruders returned by sensors and collision detection result, a conflict resolution must be applied to resolve the conflict. The function of the conflict resolution stage can be expressed either as an advisory system, which provides the appropriate commands that are required to avoid the threats, or as a feedback system with the ability to monitor if the conflicts have been resolved or not. Many studies have been carried out to solve the collision avoidance problem in airspace to improve the performance for UAV, the categories of the existing collision avoidance approaches developed for UAV are illustrated in Fig. 2.3. The details of each class are discussed in the following.

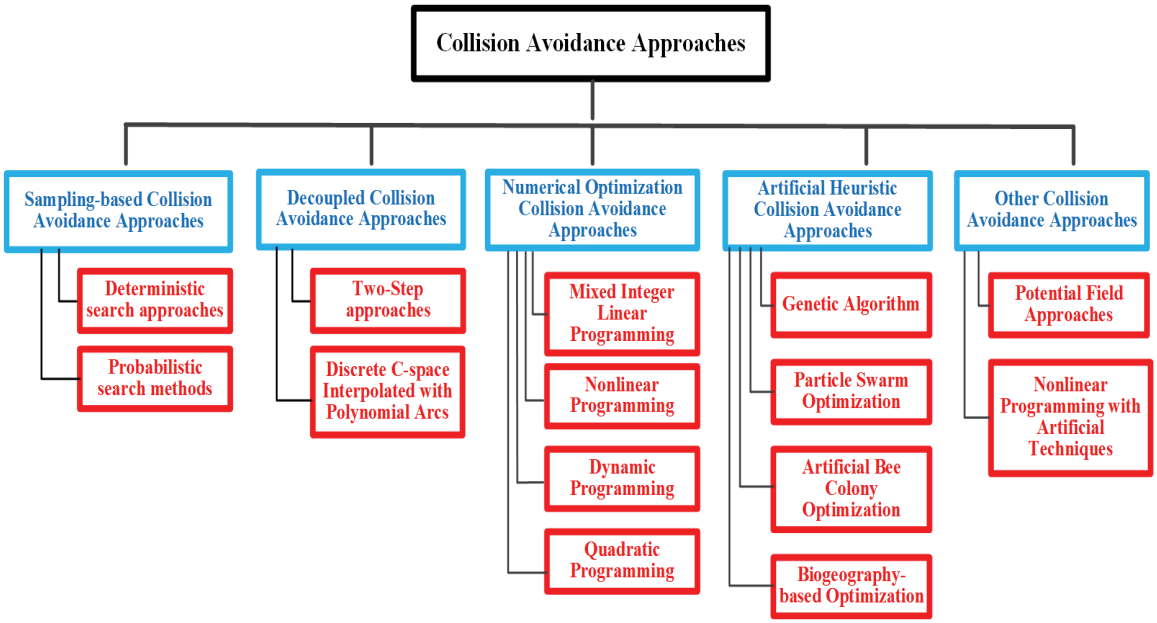


Figure 2.3: Categories of UAV collision avoidance approaches.

2.3.1 Sampling-Based Collision Avoidance Approaches

Following the scheme of sampling-based approaches, the collision avoidance is exploited to find a feasible trajectory within a limited quantity of candidates from a continuous state space. Based on the scheme of the search algorithms, this sort of methods can be divided into two parts: deterministic and probabilistic methods, respectively.

Deterministic Search Approaches

To search the minimum-time path, the overall state space of the UAV is discretized into a finite set of points. Using a virtual moving target, the optimal path from an initial position and orientation to a final position and orientation for an airplane with a bounded turning rate in the presence of a constant wind was presented in (McGee & Hedrick, 2007). Modeling UAV as a particle moving at a constant air-relative speed and with symmetric bounds on turn rate, a minimum-time path in the horizontal plane is derived using trochoidal paths and straight segments (Techy & Woolsey, 2009). Yokoyama and Ochi (2009) described two types of path planning with the ability to calculate the paths between two waypoints under constant wind conditions for skid-to-turn UAVs. One type is a rigorous optimization algorithm based on the Euler-Lagrange formulation and the minimum principle with analytical integration of the path. The other is a fast algorithm describing the path by two

circular arcs connected by a line segment or another circular arc in the air mass frame, which is similar to the Dubins path. Formulating the general aircraft visual reconnaissance problem as static ground targets in terrain, the path planning problem can be reduced as the Polygon-Visiting Dubins Traveling Salesman Problem (PVDTSP) and a shortest planar closed tour can therefore be derived by visiting at least one point in each of a set of polygons (Obermeyer, Oberlin, & Darbha, 2012). To solve the scenarios where targets are close together and polygons may overlap significantly, a Finite One-in-Set Traveling Salesmen Problem (FOTSP) based roadmap is constructed to approximate a PVDTSP. With respect to the above cited references, the impacts of winds and constraints of UAV are taken into consideration at the unit of path planning. However, the problem of avoiding intruders is not solved. Due to the roadmap that is only dependent on a priori knowledge, this type of approach may not be appropriate for path planning in the event of encountering aircraft. Based on the on-board camera, UAV can maintain a coverage map, thus UAV path can be planned in real time. Following the proposed framework, a priori knowledge is no longer required for the detailed knowledge of the workspace (Paull, Thibault, Nagaty, Seto, & Li, 2014).

Probabilistic Search Approaches

Several probability approaches have been devoted and successfully applied to path planning. The probabilistic roadmap planner consists of two phases: preprocessing and query processing. In preprocessing phase, the configuration space of UAV is randomly sampled into a one-dimensional graph and a feasible path is derived by connecting these configurations to other nearby configurations. The query phase connects the original and goal configurations to the roadmap. Extending the tree by a pre-defined motion primitive set, Rapidly-exploring Random Tree (RRT) as a randomized data structure is developed to produce a collision-free path, which can satisfy the specified approach direction and arrival time constraints (Frazzoli, Dahleh, Feron, et al., 2001). In Yu, Beard, and Byrne (2010), a randomized, incremental roadmap building algorithm is capable of dealing with the system's dynamics in the presence of moving obstacles. This roadmap approach takes into account the dynamic constraints on the vehicle's motion. To cope with uncertain dynamic obstacles, a RRT based path planner integrated with the chance constraint framework is proposed in (Kothari & Postlethwaite, 2013) accounting for uncertainty. Cooperated with a number of heuristics, the proposed path planning improves the computational speed. Considering the partial detection in the form of a task difficulty map, a Gaussian mixture model

based mode goodness ratio heuristic (Lin & Goodrich, 2014) is proposed to search for effective paths through the parameter space at different levels of resolution.

2.3.2 Decoupled Collision Avoidance Approaches

The philosophy of this approach can be separated into two units. The first unit is aimed at constructing a discrete trajectory in the configuration space, which is solved by the relevant algorithms, such as probabilistic roadmap, Voronoi approach. Based on the resulting path, the feasible trajectory is then generated taking into account the dynamical constraints of UAV as the second unit. The advantage of decoupled path planning methods is that the computational efficiency can be guaranteed. The disadvantage is that the completeness and optimality of the proposed path is difficult to be proven.

Two-Step Approaches

In the first step, an optimal trajectory is obtained under the specific constraints. The second step is deployed to ensure the safe flight of UAV in airspace. A global discrete search strategy for real-time generations of four-dimensional trajectories is deployed for a single UAV (Yang & Zhao, 2004). The first step is to represent the obstacles and potential conflicts by several basic shapes and their combination, and define mathematical conditions that ensure the safe flight. Then successor point is generated by the A* search scheme which can avoid all specified obstacles and stay within limits of dynamic vehicle motion constraints. In Vachtsevanos, Tang, Drozeski, and Gutierrez (2005), two steps exploited in the path planning are presented for a rotary-wing UAV. At the initial step, A* search algorithm is deployed for the search of the optimal path from the original to the terminal point under the constraints of distance, hazard, and UAV maneuvering. Then, the planned trajectory is further smoothed by a route filter. Based on the probabilistic risk in the surrounding area with respect to the UAV's current position, the algorithm proposed in (Y. Kim, Gu, & Postlethwaite, 2008) produces a series of waypoints out of the threat area and converges to a given target point finally. The proposed approach to path planning that can resolve the transition issue is also suitable for fully autonomous UAV applications. The problem when one can switch back to the first phase is addressed as well. In Karelahti, Virtanen, and Öström (2008), two steps are involved in the path planning. Using an integration inverse method, a near-optimal aircraft trajectory is evaluated at the first step with a sophisticated five-degree-of-freedom performance model. At the second phase, the parameters affecting the optimization and inverse simulation are adjusted correspondingly. Taking into account

the fuel or time constraints as actual optimization constraints, the initial path is produced (Ogren & Winstrand, 2008). Risk estimates of individual path segments in the planning are subsequently considered in second phase of path planning, which aim at maximizing an estimate of the overall mission success rate. The avoidance problem modeled by the branch-and-bound approach is exploited by two steps. Based on the decision between passing an obstacle clockwise or counterclockwise, a geometric branching strategy is developed in the first phase. At the second phase, the resulting subproblems are solved by constructing simple solutions on each chosen side. However, the optimal trajectories produced by (Eele & Richards, 2009) are constrained to avoid fixed obstacles. The concept of path planning with obstacle avoidance in (Dai & Cochran, 2010) is to firstly utilize the Voronoi graph to generate an initial path, and secondly refine the initial path by a simplified parameter-identification procedure in cluttered environment.

Discrete C-Space Interpolated with Polynomial Arcs

The philosophy of this approach is that firstly a discrete planner is carried out to generate an ordered set of waypoints, fitted with a polynomial arcs based spline, and then the path produced by circular arc segments and straight segments is without violating the acceleration constraints (Chamseddine, Zhang, Rabbath, & Theilliol, 2012). In Chamseddine, Zhang, Rabbath, Join, and Theilliol (2012), the proposed algorithms of path planning and re-planning are appropriate in the presence of faults, which are validated by flight tests of a quadrotor UAV.

2.3.3 Numerical Optimization Approaches

A collision avoidance problem is modeled as a numerical optimization problem using mathematical programming methods, which aims at deriving a collision-free trajectory from the initial point to the destination. The numerical optimization approaches to the collision avoidance can be briefly divided into five categories: Mixed Integer Linear Programming (MILP), Nonlinear Programming (NP), Dynamic Programming (DP), Quadratic Programming (QP).

MILP Based Method

MILP represents the problem of collision avoidance as an optimization problem with a series of constraints. The objective function is to minimize the time required for traveling through waypoints. MILP approach integrated with indirect branch-and-bound optimization reformulates the collision avoidance problem in a linearized form, and a commercial

software is carried out to solve the MILP problem (Schrijver, 1998). MILP can extend continuous linear programming to include binary or integer decision variables to encode logical constraints and discrete decisions together with UAV dynamics. In Raghunathan, Gopal, Subramanian, Biegler, and Samad (2004), the UAV collision-free trajectory generation is formulated as a 3D optimization problem under certain conditions in the Euclidean space, characterized by a set of decision variables, a set of constraints and the objective function. MILP is used within a receding horizon framework to form aircraft trajectories with a minimum fuel cost to mitigate persistent contrail formation (Campbell, Bragg, & Neogi, 2013). In Radmanesh, Kumar, Nemati, and Sarim (2015), MILP is used for obtaining optimal trajectory from a global perspective for a group of vehicles simultaneously. In Borrelli, Subramanian, Raghunathan, and Biegler (2006), the tasks of avoidance of obstacles and waypoint navigation are incorporated as constraints in the MILP problem with the cost function that incorporates the time and energy required for completing the task. Taking into account the quadrotor dynamics, current states, and current control inputs, the future trajectory of the vehicle is estimated to avoid conflicts (Israelsen et al., 2014). With the increase of obstacles and dynamic constraints, the computation complexity is remarkably increasing.

NP Based Method

The process of NP is to produce trajectories to maintain the safety distance between each aircraft while minimizing the specified objective function. Modeling the aircraft as nonlinear point-mass with cylindrical protection zone, the three-dimensional coordinated maneuvers for resolution of conflicts are presented by NP in (Jain & Tsiotras, 2008). In Geiger et al. (2008), the optimal control problem of finding trajectories that minimize a certain objective function while maintaining the safe separation of the UAV are solved by NP. The trajectory optimization with the constraints of system states and control inputs is formulated as a NP problem (Williams, 2009). The automatic calculation of a suitable nonuniform grid is proposed to solve the nonlinear programming problem. The path generated by direct collocation with nonlinear programming can produce maximal surveillance time of a moving or constant ground target. Moreover, the generated path takes into account the aircraft performance or mission constraints. Representing state equations and interpolation by local Hermite interpolating polynomials and the Legendre-Gauss-Lobatto points, NP is capable of solving the path optimization problem (Ketema & Zhao, 2010). To plan a feasible trajectory for MAV in winds, the controllability of a desired target point and the reachable set

from a given trajectory point are solved by NP (Tisdale, Kim, & Hedrick, 2009). Gradient-based nonlinear programming method is presented in (Patel & Goulart, 2011) to derive avoidance trajectories for aircraft with computation times sufficiently fast for the real-time system. Raghunathan et al. (2004) proved the capability of receding-horizon path planning to perform autonomous search and localization in a cooperative setting.

DP Based Method

The principle of DP is to divide the optimization problem into smaller sub-problems until a simple case is reached that can be easily solved. The solution of the DP based path planning is to calculate a shortest path from the original point to a target point over a group of connected nodes without entering non-fly zone. The stochastic dynamic programming is used in (Jorris & Cobb, 2008) to resolve conflicts and to control data gathering actions that can improve decision-making. Following the principle of DP, an autonomous trajectory planner in (Gu, Snider, Dolan, & Lee, 2013) is proposed to generate a minimum-time trajectory with safety terminal and state constraints.

QP Based Method

The philosophy of QP is to produce a trajectory for cooperative attack whilst the system dynamics and sensor performance are taken into consideration. Yokoyama (2012) applies the QP to generate paths for avoiding turbulent regions in real time using information obtained by an on-board Doppler LIDAR. QP provides a complete solution to the obstacle-free-path-generation problem for a UAV in the real-time implementation (Jung & Tsiotras, 2013). To form a communication network to multiple targets with given radio communication capacities, QP based path planning algorithm is proposed for automatic initialization and optimal reconfiguration of the communication topology in case of failures or several radio paths.

2.3.4 Artificial Heuristic Approaches

By mimicking the natural evolution process, Artificial Heuristic approaches have been employed to optimize a trajectory bypassing all intruders. The relevant methods include Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Artificial Bee Colony Optimization (ABCO), and Biogeography-Based Optimization (BBO), respectively.

GA Based Approach

Initiating from the process of natural revolution, GA is a heuristic search that can be potentially used to address a path planning optimization problem. The advantages of GA is that

there is no need to compute the gradient of the cost or the constraint functions. The fitness of each individual solution is evaluated by objective function (Winter, Periaux, Galan, & Cuesta, 1996). A new population of trajectories is consequently generated using a series of genetics based operations. A modified GA-based offline/online path planner is used to calculate a curved path line in a 3D rough terrain environment (Nikolos & Tsourvelouds, 2009). The authors in (Pehlivanoglu, 2012) exploit GA to optimize the path of UAV flying over a given terrain. An optimization algorithm called multi-frequency vibrational genetic algorithm is used to solve the path planning problems of autonomous UAVs. In Ergezer and Leblebicioglu (2013), GA is applied to maximize the collected information from the desired regions, while avoiding flying over forbidden regions and reaching the destination.

PSO Based Approach

PSO is a population-based algorithm that mimics the social behaviors of animals. The principle of PSO can be described as following. The method applies the approach of problem solving in groups. Each solution consists of set of parameters and represents a point in multi-dimensional space. The solution is called particle and the group of particles (population) is called swarm. Each particle has a corresponding velocity and moves around in the search-space. During each iteration, the particles are updated following the rule that is the best known position in the search-space as well as the entire swarm's best known position. The global search is performed by the PSO algorithm at the beginning and local search. The collision-free trajectory is formulated as a hybrid optimization problem taking into account the constraints of UAV dynamics, environment and real-time requirements. Considering the nonlinear model and performance constraints, feasible trajectories are produced by PSO (Karimi & Pourtakdoust, 2013) to avoid random threats arised sporadically in the terrain. Path planning problem for UAV in different known and static environment is solved by phase angle-encoded and quantum-behaved PSO (Fu, Ding, & Zhou, 2012). In Fu, Ding, Zhou, and Hu (2013), UAV path planning in the presence of various threats is solved by a hybrid differential evolution with quantum-behaved PSO. The advantages of the improved PSO include faster convergence speed and stronger robustness.

ABCO Based Approach

ABCO algorithm is an optimization algorithm under the inspiration of collective behavior on honey bees (Karaboga & Basturk, 2007). ABCO conducts both global search and local search in each iteration, and as a result the probability of finding the optimal parameters

is significantly increased, which can efficiently avoid local optimum to a large extent. Improved by a balance-evolution strategy, a novel ABCO (Li, Gong, & Yang, 2014) is applied to obtain an optimal flight route with the threats and constraints in the combat field well considered. Xu, Duan, and Liu (2010) proposed a ABCO-based path planner to generate a feasible flight path under complex combat environment for Uninhabited Combat Air Vehicles (UCAVs).

BBO Based Approach

BBO is a novel population-based optimization algorithm (Simon, 2008). The basic idea of BBO is influenced by the geographical distribution of species over time and space. Similarly to other artificial heuristic algorithms, the search agent in BBO is called habitat. Each habitat is assigned a vector of habitants, representing as a possible solution for the proposed problem. Global optimum is searched by two steps: migration and mutation. Solutions are maintained from one iteration to the next, and the solutions can also be improved by migration. An optimal or near-optimal flight path is generated by a novel Chaotic Predator-Prey Biogeography-Based Optimization (CPPBBO) approach to ensure UCAV avoid hostile threats and safely reach an intended target with minimum fuel cost. The chaos theory and the concept of predator-prey are integrated in the migration operation to enhance the global convergence of the BBO algorithm (Zhu & Duan, 2014).

2.3.5 Other Methods

Potential Field Approaches

Potential Field based path planning is a virtual electric field approach. The obstacle-charges in the workspace of the UAV exert a repulsive effect on the UAV while the desired target-charge position has an attractive effect on the UAV center position. As a result, it is popularly applied for path planning of collision avoidance. An analytically tractable potential field model of free space is presented in (Chuang & Ahuja, 1998). Object paths going through narrow areas of free space without obstacle can be guaranteed subsequently. Considering the situation where the workspace of a robot cannot be segmented into geometrical subregions, the Gamma-Harmonic Potential Field (GHPF) approach is adaptable to avoid zero-probability (definite threat) regions and the convergence to the goal are provided finally (Masoud et al., 2012). Provided that positions of the UAVs are detected autonomously, collision avoidance between the UAVs and other geographical obstacles are addressed by potential fields in (Campbell et al., 2013).

NP with Artificial Techniques

Since single type of path planning method has its own advantages and disadvantages, the combined approaches have received a growing interest around the problem of path planning. With the help of artificial techniques (Yu, Cao, & Zhang, 2011), the NP based path planning presents an improvement in a real-time calculation aspect. In Subbarao and Shippey (2009), the initial guesses were obtained by GA, while the minimum-time trajectory is generated by NP. With the advantage of offering accurate computation of gradients, NP is applied to approximate the objective functions and system dynamics (Geiger & Horn, 2012). Since there is no need of the formulation of analytical gradients, the method is more easily extended to complex applications with different objective functions and constraints.

2.3.6 Summary

Trajectory planning to avoid threats is inspired by mission constraints, vehicles characteristics and mission environment. A large amount of path planning approaches have been developed to generate feasible maneuvers while UAV is still following its mission. Sampling-based algorithms rely on a collision-detection module, providing information about the feasibility of candidate trajectories, and connect a set of points sampled from the obstacle-free space in order to build a road-map of feasible trajectories. The road-map is then used to construct the solution to the collision avoidance problem. Decoupled trajectory planning algorithm computes a discrete path through the configuration space as the first step. The resulting path is then used as the basis for the generation of a trajectory that is feasible for dynamics-constrained UAV. Numerical optimization trajectory planning find an optimum collision-free trajectory to a goal point based on the predefined cost functions. Artificial heuristic methods deal with a set of solutions in each of iteration and finally derive an optimum collision-free trajectory of UAV. According to (Goerzen, Kong, & Mettler, 2010), the problem of trajectory planning is considered as the optimization process combining a kinematic model with a set of cost metrics. An optimal resolution strategy is then solved with the lowest cost.

Chapter 3

Problem Formulation

The specific problem to be formulated for collision avoidance is how a UAV is being navigated from a start position to a destination, while bypassing intruders.

3.1 UAV Dynamics

As shown in Fig. 1.1, the beginning unit of S&A system is to model the flight dynamics. Developing and testing a set of differential equations able to describe aircraft-dynamic behaviors under a given set of commands would provide a useful tool to verify design choices in simulation.

In order to clarify the meaning of the differential equations presented further, the definition of the principal reference systems exploited in the model will be briefly recalled (De Filippis, 2012):

- **Body Frame**: this frame has origin in the aircraft center of gravity, x axis is aligned with body fuselage axis, y axis is perpendicular to the previous one and directed along right wing, z axis is completing the orthogonal frame and directed along wing top-bottom direction.
- **Wind Frame**: this frame has origin in the aircraft center of gravity, x axis is aligned with relative-wind speed, y axis is perpendicular to the previous one and directed on the right, z axis is completing the orthogonal frame and directed downward.
- **Local-Vertical Frame**: this frame has origin in the aircraft center of gravity, x axis is in the north direction, y axis is perpendicular to the previous one and in the east direction, z axis is completing the orthogonal frame, aligned with gravitational force and directed downward.

- **Ground Frame**: this frame has origin in a fixed point on the Earth, x axis is coincident with longitude axis, y axis is perpendicular to the previous one and coincident with latitude axis, z axis is completing the orthogonal frame, aligned with gravitational force in the origin and directed upward.

In this paper, equations of motion are written in body frame assuming flat earth and neglecting revolution on its orbit and rotation along its axis. These assumptions allow to neglect some terms in the equations of forces and moments. Then the following systems are obtained.

The decoupled lateral and longitudinal equations are deployed to model the UAV dynamics based on Euler-Lagrange (EL) formalism (Akmeliawati & Mareels, 2010). Lateral dynamics is proposed to achieve the stability of motion at horizontal plane with two control input δ_r and δ_a. Other inputs about longitudinal dynamics are set to the equilibrium point of UAV at 80 m/s airspeed. Combining with roll and yaw motions, the lateral dynamics can be described by the function of roll angle ϕ, yaw angle ψ, roll rate $\dot{\phi}$, and yaw rate $\dot{\psi}$:

$$I_x\ddot{\phi} + \frac{\partial F}{\partial \dot{\phi}} = M_{\phi,\delta_a}\delta_a + M_{\phi,\delta_r}\delta_r, \tag{1}$$

$$I_z\ddot{\psi} + \frac{\partial F}{\partial \dot{\psi}} = M_{\psi,\delta_a}\delta_a + M_{\psi,\delta_r}\delta_r, \tag{2}$$

where $R_l = \begin{bmatrix} I_x & 0 \\ 0 & I_z \end{bmatrix}$ represents the aircraft inertial with respect to x and z directions. The limits of the aileron deflection δ_a and the rudder deflection δ_r satisfy:

$$\begin{cases} -25^\circ < \delta_a < 25^\circ \\ -30^\circ < \delta_r < 30^\circ \end{cases}.$$

$M_{L,u} = \begin{bmatrix} M_{\phi,\delta_a} & M_{\phi,\delta_r} \\ M_{\psi,\delta_a} & M_{\psi,\delta_r} \end{bmatrix}$ is the input matrix contributed by the drag, lift and side force. The parameters of $M_{L,u}$ are extracted from the model described in (Magni, Bennani, & Terlow, 1997). Assuming θ is fixed to 0.0284 rad, the components of $M_{L,u}$ are expressed as:

$$M_{\phi,\delta_a} = -630.63V_A^2\cos(\theta)\cos(\psi),$$

$$M_{\phi,\delta_r} = 231.231V_A^2(\cos(\theta)\cos(\psi) - 634.41378V_A^2(\cos(\phi)\sin(\theta)\cos(\psi) + \sin(\phi)\sin(\psi)),$$

$$M_{\psi,\delta_a} = 630.63V_A^2\sin(\theta),$$

$$M_{\psi,\delta_r} = -231.231V_A^2\sin(\theta) - 634.41378V_A^2\cos(\phi)\cos(\theta).$$

Based on Eq. (1), Eq. (2) and the aerodynamic coefficients given in (Magni et al., 1997), the parameters of partial differential of the Rayleigh terms $\frac{\partial F}{\partial \dot{q}_l} = [\frac{\partial F}{\partial \dot{\phi}} \quad \frac{\partial F}{\partial \dot{\psi}}]^T$ satisfy:

$$\begin{aligned}\frac{\partial F}{\partial \dot{\phi}} &= -1051.05\cos\theta\cos\psi(-1.4\beta + 6.6(-11\dot{\phi} + 11\dot{\psi}\sin\theta + 5\dot{\psi}\cos\phi\cos\theta - 5\dot{\theta}\sin\phi) \\ &/V_A)V_A^2 - 1051.05(\sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi)(-0.4461 - 2.1505\alpha - 92.4424(\dot{\theta}\cos\phi \\ &+ \dot{\psi}\sin\phi\cos\theta)/V_A + 0.11(6.0723\alpha + 1.0656 + 24.6016(\dot{\theta}\cos\phi + \dot{\psi}\sin\phi\cos\theta)/V_A) \\ &\cos(\alpha) + 0.11(0.13 + 0.07(5.5\alpha + 0.654)^2)\sin\alpha)V_A^2 - 1051.05(\cos\phi\sin\theta\cos\psi + \sin\phi \\ &\sin\psi)((1 - 3.8197\alpha)\beta + 6.6(1.7\dot{\phi} - 1.7\dot{\psi}\sin\theta - 11.5\dot{\psi}\cos\phi\cos\theta + 11.5\dot{\theta}\sin\phi)/V_A - \\ &0.176\beta)V_A^2,\end{aligned} \tag{3}$$

and

$$\begin{aligned}\frac{\partial F}{\partial \dot{\psi}} &= -1051.05\cos\theta\cos\psi(-1.4\beta + 6.6(-11\dot{\phi} + 11\dot{\psi}\sin\theta + 5\dot{\psi}\cos\phi\cos\theta - 5\dot{\theta}\sin\phi) \\ &/V_A)V_A^2 - 1051.05\sin\phi\cos\theta(-0.4461 - 2.1505\alpha - 92.4424(\dot{\theta}\cos\phi + \dot{\psi}\sin\phi\cos\theta)/V_A \\ &+ 0.11(6.0723\alpha + 1.0656 + 24.6016(\dot{\theta}\cos\phi + \dot{\psi}\sin\phi\cos\theta)/V_A)\cos(\alpha) + 0.11(0.13 + \\ &0.07(5.5\alpha + 0.654)^2)\sin\alpha)V_A^2 - 1051.05\cos\phi\cos\theta((1 - 3.8197\alpha)\beta + 6.6(1.7\dot{\phi} - 1.7 \\ &\dot{\psi}\sin\theta - 11.5\dot{\psi}\cos\phi\cos\theta + 11.5\dot{\theta}\sin\phi)/V_A - 0.176\beta)V_A^2,\end{aligned} \tag{4}$$

where β is the sideslip angle, α is the angle of attack and V_A is the airspeed, respectively.

For the vertical plane, the longitudinal dynamics is constructed with fast dynamics related to the pitch motion and slow dynamics along $x-z$ axis. Considering the pitch motion influenced by pitch angle θ and pitch rate $\dot{\theta}$, the fast dynamics is described as:

$$I_y\ddot{\theta} + \frac{\partial F}{\partial \dot{\theta}} = M_{\theta,\delta_e}\delta_e + M_{\theta,\delta_{th}}\delta_{th}, \tag{5}$$

where I_y is the aircraft inertial, $M_{\theta,\delta_e}\delta_e - \frac{\partial F}{\partial \dot{\theta}}$ describes the pitch moment relied on the elevator, and $M_{\theta,\delta_{th}}\delta_{th}$ is the moment derived from the thrust. And the slow dynamics with respect to the longitudinal motion is modeled by x, z positions:

$$m\ddot{x} + \frac{\partial F}{\partial \dot{x}} = M_{x,\delta_e}\delta_e + M_{x,\delta_{th}}\delta_{th}, \tag{6}$$

$$m\ddot{z} + \frac{\partial F}{\partial \dot{z}} + mg = M_{z,\delta_e}\delta_e + M_{z,\delta_{th}}\delta_{th}, \tag{7}$$

where m is the aircraft mass. As indicated in Fig. 3.1, (x, z) is the position of UAV with respect to the center of gravity in the inertial axis, where the positive x-direction points east and positive z-direction points upside. The elevator deflection δ_e and the throttle angle δ_{th} are bounded by:

$$\begin{cases} -25^\circ < \delta_e < 10^\circ \\ -0.5^\circ < \delta_{th} < 10^\circ \end{cases}.$$

The parameters of control input matrix $M_{X,u} = \begin{bmatrix} M_{x,\delta_e} & M_{x,\delta_{th}} \\ M_{z,\delta_e} & M_{z,\delta_{th}} \\ M_{\theta,\delta_e} & M_{\theta,\delta_{th}} \end{bmatrix}$ satisfy:

$$M_{x,\delta_e} = -121.52\dot{z}V \quad M_{x,\delta_{th}} = 19.62m\cos(\theta),$$

$$M_{z,\delta_e} = 121.52\dot{x}V \quad M_{z,\delta_{th}} = 19.62m\sin(\theta),$$

$$M_{\theta,\delta_e} = -2925.47V^2 \quad M_{\theta,\delta_{th}} = 37.278m.$$

And the details of the Rayleigh terms $\frac{\partial F}{\partial \dot{X}}$ and $\frac{\partial F}{\partial \dot{\theta}}$ are indicated as:

$$\frac{\partial F}{\partial \dot{x}} = (25.47\dot{x}V + 337.21\dot{x}V\alpha^2 + 80.20\dot{x}V\alpha + 967.02\dot{z}V\alpha + 169.70\dot{z}V) + 3917.80\dot{z}\dot{\theta}, \tag{8}$$

$$\frac{\partial F}{\partial \dot{z}} = (-967.02\dot{x}V\alpha - 169.70\dot{x}V + 25.47\dot{z}V + 337.21\dot{z}V\alpha^2 + 80.20\dot{z}V\alpha) - 3917.80\dot{x}\dot{\theta}, \tag{9}$$

$$\frac{\partial F}{\partial \dot{\theta}} = (1539.73V^2\alpha + 345.69V^2 - 244.82V^2\alpha^3 - 58.22V^2\alpha^2) + 94317.23V\dot{\theta}, \tag{10}$$

where $\dot{x}$ and $\dot{z}$ are the velocity along x-axis and z-axis, $V = \sqrt{\dot{x}^2 + \dot{z}^2}$, α = θ - γ, γ = $\arctan(\dot{z}/\dot{x})$.

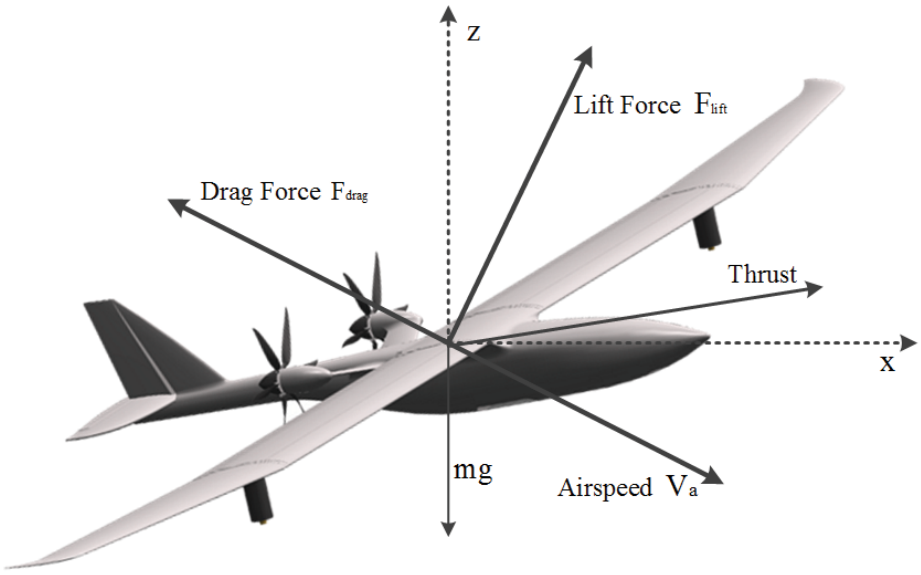


Figure 3.1: A model of forces in UAV inertial frame.

3.2 Formulation

Tracing back to Fig. 1.1, collision avoidance maneuvers rely on the parameters from the conflict detection process. It determines that the UAV trajectory has to be re-planned or not for avoiding collisions. The miss distance following the closest point approach (Park, Oh, & Tahk, 2008) is investigated to define the conflict in this study. If the predicted miss distance between the UAV and the intruder is smaller than the minimum separation distance within a specific block of time, the collision will occur. It is also assumed that intruders follow the prescribed trajectories without any avoidance maneuver. The intruder velocity can therefore be calculated. The relative motion of the UAV and one intruder is indicated in Fig. 3.2.

Following the procedure of the closest approach distance, the miss distance vector $\vec{D}_m$ can be derived as:

$$\vec{D}_m = \vec{V}_r \times (\vec{D} \times \hat{V}_r), \tag{11}$$

where $\vec{D}$ is the distance vector locating Intruder with respect to UAV, and the unit vector $\hat{V}_r$ lies in the direction of the relative velocity vector of UAV with respect to the Intruder:

$$\hat{V}_r = \frac{\vec{V}_r}{\|\vec{V}_r\|}. \tag{12}$$

$\vec{V}_r$ is the relative velocity of UAV with respect to the Intruder, which can be derived as:

$$\vec{V}_r = \vec{V}_B - \vec{V}_A, \tag{13}$$

where $\vec{V}_r$ is located at the vector direction of the time to the closest point τ. Due to the relation between $\vec{D}_m$ and $\vec{D}$, $\vec{D}_m$ can be derived as:

$$\vec{D}_m = \vec{D} + (\vec{V}_B - \vec{V}_A)\tau. \tag{14}$$

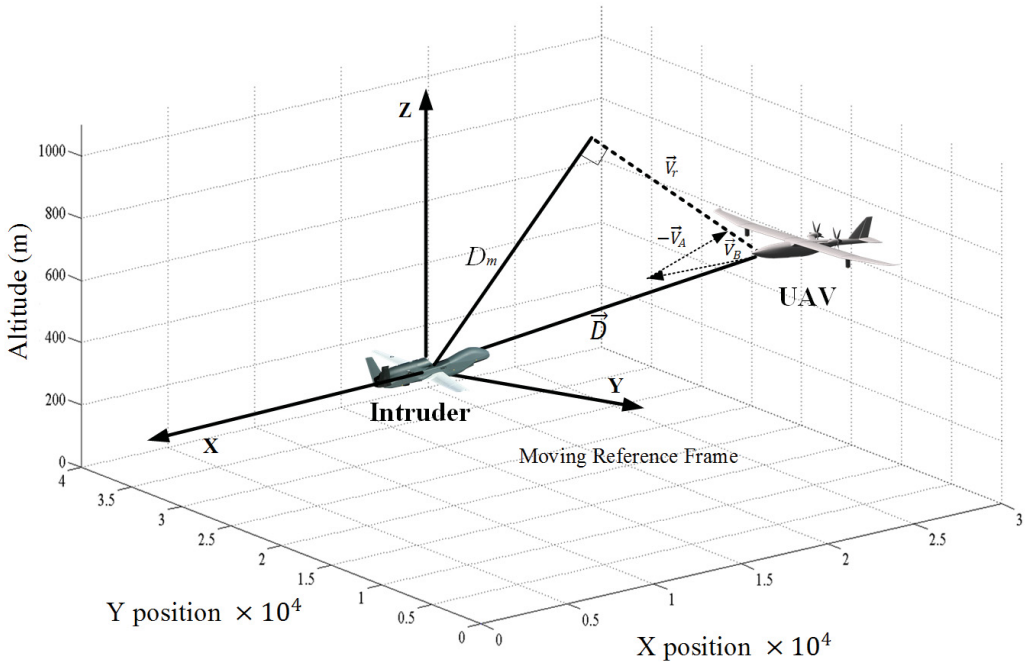


Figure 3.2: The relative motion of a UAV and an Intruder.

As can be seen in Fig. 3.2, the miss distance vector $\vec{D}_m$ and the relative velocity vector $\vec{V}_r$ are orthogonal:

$$\vec{D}_m \cdot (\vec{V}_B - \vec{V}_A) = 0. \tag{15}$$

Combining Eqs. (14) and (15), τ can be derived as:

$$\tau = -\frac{\vec{D} \cdot (\vec{V}_B - \vec{V}_A)}{(\vec{V}_B - \vec{V}_A) \cdot (\vec{V}_B - \vec{V}_A)}. \tag{16}$$

As uncertainties exist in sensing, the conflict detection approach should take into account

the sensor measurement noises. In this paper, the position, velocity, and attitude variables are modeled by Gaussian distributions (Krozel, Peters, & Hunter, 1997). A conflict is defined by evaluating the probability P that at time instant τ the derived miss distance $\|D_m\|$ is less than or equal to the required minimum separation distance D_{safe}:

$$\begin{aligned} P(conflict) &= P(\|D_m\| \leq D_{safe}) \\ &= \frac{1}{\sigma_r\sqrt{2\pi}} \int_{-D_{safe}}^{D_{safe}} e^{-\frac{(x-\overline{D}_m)^2}{2\sigma_r}} dx, \end{aligned} \tag{17}$$

where σ_r is the standard deviation of the sensor measurement noises, $\overline{D}_m$ is the mean of the miss distance $\vec{D}_m$.

If the following two conditions:

(1) τ is positive, and

(2) $P(conflict)$ is greater than P_{Na} (P_{Na} is the probability of no action),

are simultaneously satisfied, a collision will occur between the UAV and Intruder. After conflicts between intruders are determined, the conflict resolution is to ensure the safe flight in airspace. The collision avoidance for a UAV is initialized by modeling the flying environment. Considering the airspace as a grid system where each grid point is regarded as a waypoint, the environment of accommodating UAV position and predicted collisions is represented as S. All predicted collisions in S are shown as the set Col, which is denoted as follows:

$$Col = Col_1, Col_2, ..., Col_m.$$

As shown in Fig. 3.3, the trajectory is to connect segments from a starting point to the destination, which can be described as:

$$L = \vec{L}_1 + \vec{L}_2 + ... + \vec{L}_n, \tag{18}$$

where $\vec{L}_i$ is the vector of ith segment to link with two waypoints. Therefore, the set L can be rewritten as:

$$L = P_1 + P_2 + ... + P_i, \tag{19}$$

where P_i is the discrete point of the orderly grid.

To evaluate the performance of collision avoidance, the cost function F of each step is

given in the following equation:

$$F = \begin{cases} d(P_i, P_{i+1}), & (20) \\ d(P_i, P_{i+1}) \times J_t. & (21) \end{cases}$$

If the vector $\overrightarrow{P_iP_{i+1}}$ is far away from the safety zone of Col_m, the cost is derived as Eq. (20). If vector $\overrightarrow{P_iP_{i+1}}$ crosses the safety zone of Col_m, the cost is instead described as Eq. (21). J_t is the weighting factor for threats cost, $d(P_i, P_{i+1})$ is Euclidean distance between two waypoints:

$$d(P_i, P_{i+1}) = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2}. \quad (22)$$

(x_i, y_i, z_i) is the position of the UAV, $(x_{i+1}, y_{i+1}, z_{i+1})$ describes the UAV's next position.

For the sake of UAV safely evading threats, dynamic constraints to prevent the UAV from developing into an irreversible state should be considered. With respect to the studied UAV, the limit of the pitch angle and the pitch rate are bounded by

$$\begin{cases} -25^\circ < \theta < 30^\circ \\ -5^\circ/s < \dot{\theta} < 5^\circ/s \end{cases},$$

the roll angle and the roll rate should be within following ranges

$$\begin{cases} -23^\circ < \phi < 23^\circ \\ -3^\circ/s < \dot{\phi} < 3^\circ/s \end{cases}.$$

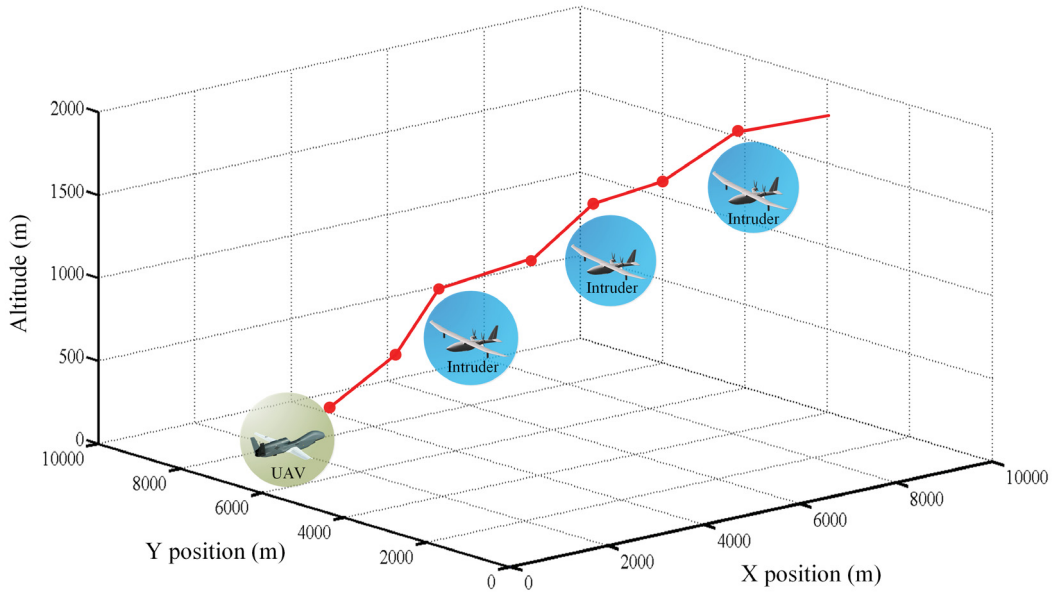


Figure 3.3: Grid-based trajectory representation.

3.3 Summary

CPA is required to project the states of UAV and intruders into the future in order to predict whether a conflict will occur. When a conflict is predicted to occur, the action implemented to avoid collisions is considered necessary. The objective of the collision avoidance is to produce a feasible trajectory with the minimum cost under the dynamic constraints of the UAV.

Chapter 4

Sense and Collision Avoidance of Unmanned Aerial Vehicles Using Differential Geometric Guidance and Flatness Approach

4.1 Differential Geometric Guidance (DGG) Based Collision Resolution

Geometric methods use the geometric properties of aircraft trajectories and utilize position and velocity vectors of all or some of the aircraft involved in the encounter. In order to predict a conflict, geometric methods compare velocity vectors of aircraft with those of obstacles. Geometric methods provide information about the geometry of conflict to the guidance algorithm, so conflict resolution strategies can be derived. Geometrical approaches consider in many cases the whole trajectory of the airplane while it avoids observed solution. Various approaches optimize given objective function while to search for the proper evasion trajectory.

An algorithm based on DGG is proposed to generate a reasonable reference of heading angle for a UAV to avoid a face-to-face conflict. The conflict resolution algorithm should guarantee that $\|D_m\|$ is greater than or equal to the minimum separation distance D_{safe}. The work herein only considers the following condition:

$$\|D_m\| = D_{safe}. \tag{23}$$

As illustrated in Fig. 4.1, when distance between two aircraft equals to R (detection distance), according to right-of-way rules, the reference heading angle of UAV-B ψ_r is:

$$\psi_r = \psi_i + \psi, \tag{24}$$

where ψ_i is the initial heading angle of UAV-B. ψ denotes the angle between the sight line and tangent line on the minimum separation circle from the Aircraft-A's position. ψ can be therefore derived as:

$$\frac{sin\psi}{1 + cos\psi} = \frac{D_{safe}}{R}. \tag{25}$$

It should be emphasized that when UAV-B reaches the point along the new flight path that is out of safety bounds of Aircraft-A, a new maneuver is performed to subtract 2ψ from current heading angle. Finally, a heading change of ψ is performed to drive UAV-B back to its original flight path.

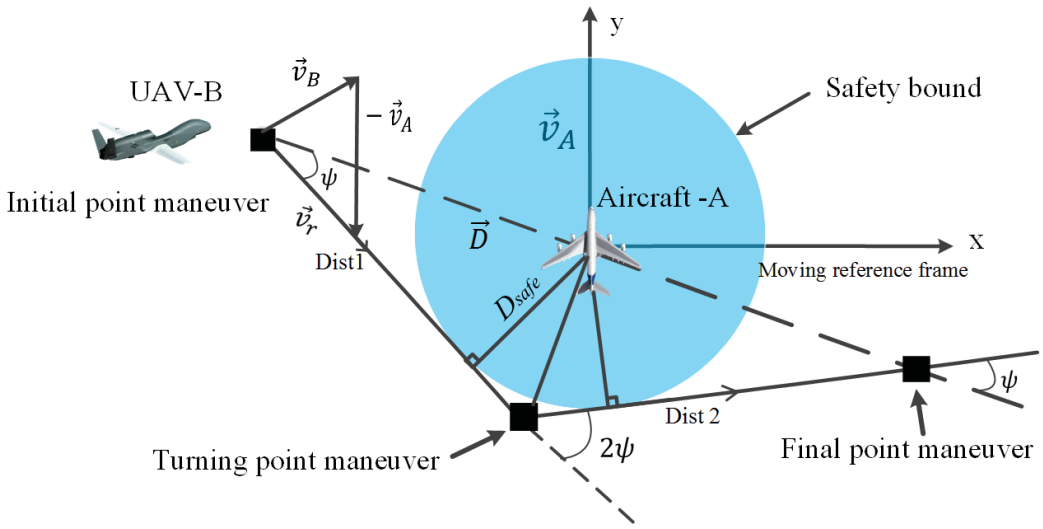


Figure 4.1: Geometry of collision resolution.

However, there exists a potential collision hazard if ψ is a same value even when the speed of intruder is different from that of UAV. In response to this practical issue, ψ is replaced by

$$\psi^* = \rho\psi, \tag{26}$$

where ρ is an influence factor of relative velocity to determine the feasible reference heading angle.

4.2 Path Planning

If the demand of heading angle is known, the UAV-B is required to avoid collision by following the planned path, while the physical constraints have to be respected as well. A flatness-based path planner is proposed for such a purpose, which is considered as an appropriate method for real-time requirement. Differential flatness theory was introduced in the late 1980s by Michel Fliess and co-researchers. Since then, they keep on being developed and on providing efficient solutions to advanced control and state estimation problems. A basic property of differentially flat systems is that through a change of variables they can be transformed to an equivalent description, which is the linear canonical form.

4.2.1 Differential Flatness Based Approach

A flatness-based nonlinear system is defined as following:

$$\begin{cases} \dot{x} = f(x,u) \\ y = h(x) \end{cases}, \tag{27}$$

where $x \in R^n$ and $u \in R^m$ represent system states and control inputs, respectively. The system is flat only if flat output $F \in R^m$ fulfills the requirements such as

$$\begin{cases} x = \Xi_1(F, \dot{F}, , F^{(n-1)}) \\ y = \Xi_2(F, \dot{F}, , F^{(n-1)}) \\ u = \Xi_3(F, \dot{F}, , F^{(n-1)}) \end{cases}, \tag{28}$$

where Ξ_1, Ξ_2, Ξ_3 are three smooth mappings and $F^{(i)}$ is the ith derivative of F. Since collision resolution method concerns heading angle only, the system is flat when flat output $F_i = \psi$. Let F_1^* be the reference trajectory for the flat output F_1, then the heading angle with respect to reference trajectory is $F_1^* = \psi^*$.

The parameterization of the heading angle ψ in functional of the flat output F (Chamseddine, Zhang, Rabbath, & Theilliol, 2012) is important in the path planning problem: the heading angle to be applied during a mission can be expressed in the desired path. Hence, the profile of the path can be tuned to keep the heading angle below the flight envelope constraints.

4.2.2 Trajectory Planning under UAV Constraints

Collision avoidance may require aggressive maneuvers. However, the constraints of actuators may interfere with the system stability in this situation. To address this problem, it is necessary to limit the range of the maximal heading angle rate and minimal settling time during the flight.

In this work, second-order system is deployed to define the reference trajectory. The natural frequency is approximated as a function of the settling time. The settling time is then tuned so that system constraints are respected. The second-order system based reference trajectory allows to calculate in a straightforward manner.

A reference trajectory is defined as:

$$\ddot{F}_1^*(t) = -2\xi\omega_n\dot{F}_1^*(t) - \omega_n^2 F_1^*(t) + \omega_n^2 r_1(t), \tag{29}$$

where ξ is the damping ratio, ω_n is the natural frequency, $r_1(t)$ is the reference along ψ direction. Moreover, assuming a critically damped system $\xi = 1$ with zero initial conditions, the solution of reference model can be derived as:

$$F_1^*(t) = [1 - (1 + \omega_n t)e^{-\omega_n t}]\psi. \tag{30}$$

According to (Chamseddine, Zhang, Rabbath, Join, & Theilliol, 2012), the relationship between ω_n and t_s can be approximated by:

$$\omega_n \approx \frac{5.83}{t_s}, \tag{31}$$

where t_s is the settling time of the desired heading angle. Thus, Eq. (30) can be rewritten as

$$F_1^*(t) = [1 - (1 + \frac{5.83}{t_s}t)e^{-\frac{5.83}{t_s}t}]\psi. \tag{32}$$

Since $F_1^* = \psi^*$, one can obtain:

$$\psi^*(t) = [1 - (1 + \frac{5.83}{t_s}t)e^{-\frac{5.83}{t_s}t}]\psi. \tag{33}$$

To determine the settling time of the heading angle, the time derivative of the heading angle with respect to time is derived as:

$$\dot{\psi}^* = (\frac{5.83}{t_s})^2 t\psi e^{-\frac{5.83}{t_s}t}. \tag{34}$$

Secondly, time derivative of the heading angle rate with respect to time:

$$\ddot{\psi}^* = (\frac{5.83}{t_s})^2 \psi e^{-\frac{5.83}{t_s}t}(1 - \frac{5.83}{t_s}t). \tag{35}$$

By setting Eq. (35) to zero, one can obtain the maximal heading angle rate when $t = \frac{t_s}{5.83}$. Substituting this value into Eq. (34), the extreme value is obtained as:

$$\dot{\psi}^*(t_s) = \frac{5.83}{t_s}\psi e^{-1}. \tag{36}$$

To make sure that $|\dot{\psi}^*| \leq \omega^*_{max}$ (ω^*_{max} is the maximal heading rate), t_s has to satisfy:

$$t_s \geq |\frac{5.83\psi}{e\omega^*_{max}}|. \tag{37}$$

It is worthwhile mentioning that the minimal t_s is equal to $|5.83\psi/e\omega^*_{max}|$ to make the trajectory feasible.

4.3 Experimental Results

4.3.1 Experimental Platform

The simulation platform of this S&A study is illustrated in Fig. 4.2. The Marinvent Flight Simulator is used to mimic the Aircraft-A, whilst the developed S&A for UAV-B is implemented in the laptop as shown in Fig. 4.2. The communication between the flight simulator and the laptop is through a Local Area Network (LAN). The intruder flying in the Flight Simulator follows the initial trajectory, which is set up by X-Plane without avoidance maneuver. The laptop is configured with a 2.0 GHz Intel Core i7-4510U processor and 8 GB RAM.

4.3.2 Testing Scenarios

The original trajectories without conflict resolution are shown in Fig. 4.3. It is obvious that the safety bounds of UAV-B is intersected with that of Aircraft-A if no S&A actions are taken. In simulation studies, four scenarios are included. They are:

(1) **Scenario 1:** The initial ground speed of both Aircraft-A and UAV-B is 150 m/s and the maximal heading angle rate of UAV-B is $2^\circ/s$. The initial heading angles of UAV-B and Aircraft-A are 161° and 352°, respectively.

(2) **Scenario 2:** Aircraft-A speeds up when UAV-B starts to avoid collisions. The rest of initial conditions are the same as in Scenario 1.

(3) **Scenario 3:** Aircraft-A flies at a maximum speed 180 m/s. The rest of initial conditions are the same as in Scenario 1.

(4) **Scenario 4:** Aircraft-A speeds down when UAV-B starts to avoid collisions. The rest of initial conditions are the same as in Scenario 1.

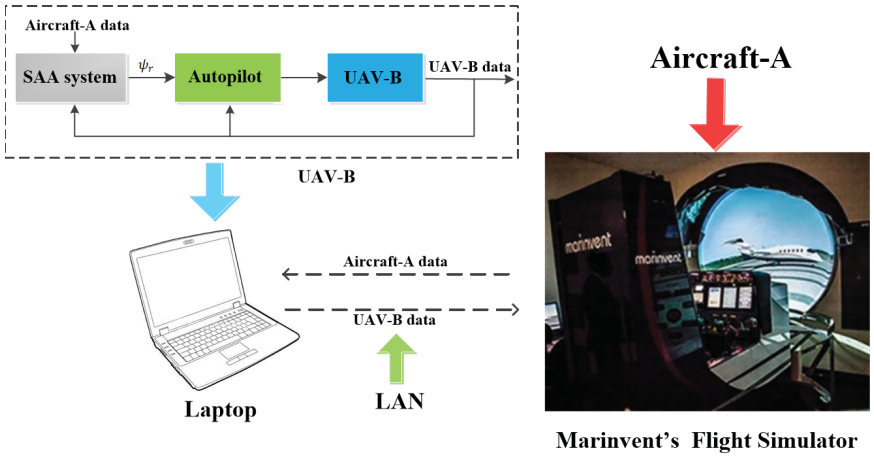


Figure 4.2: The platform for validating S&A scheme.

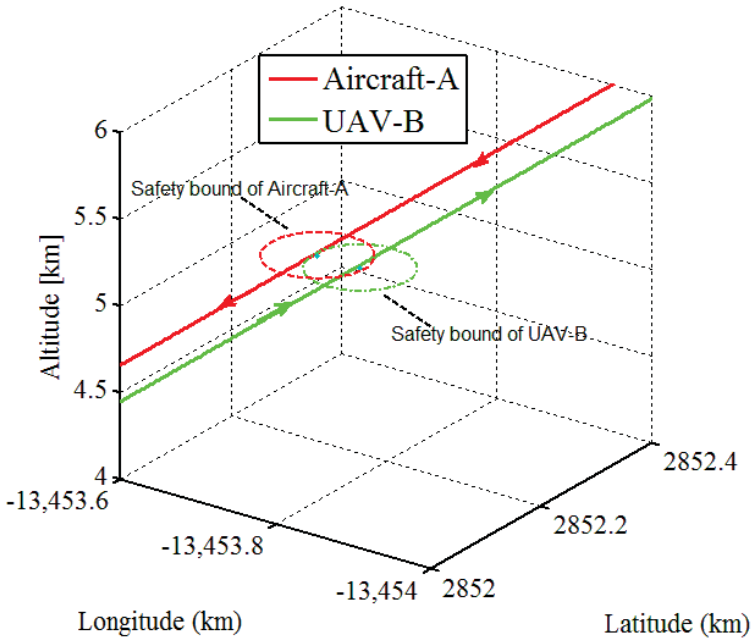


Figure 4.3: Original trajectories without S&A.

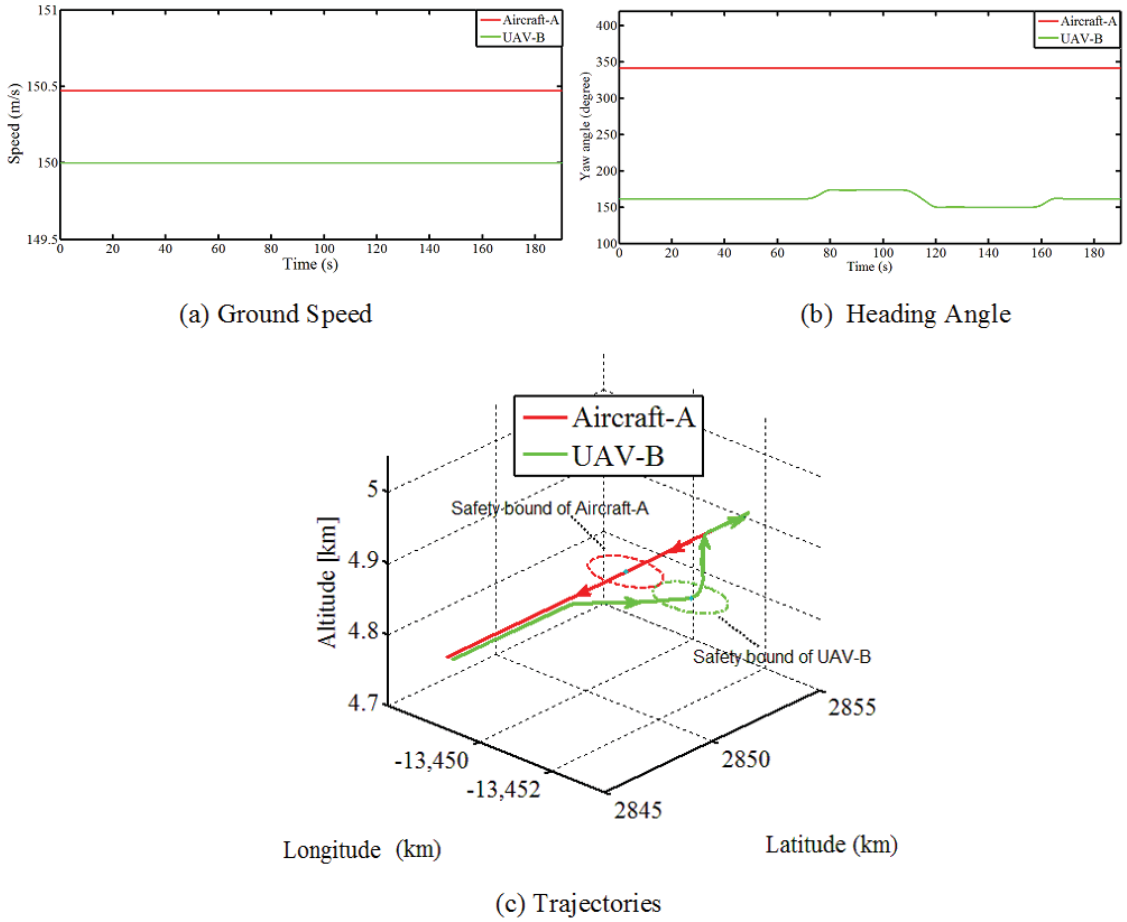
4.3.3 Experimental Results and Evaluations

(1) *Results of Scenario 1*

As shown in Fig. 4.4(a), Aircraft-A and UAV-B fly in the same speed. It is illustrated in Fig. 4.4(b) that the heading angle starts to change at $70s$. In addition, a second change of heading angle takes place after $100s$ at the turning point where UAV-B is out of safety bounds of Aircraft-A at the minimal relative distance. After the last turning, UAV-B is back to the initial trajectory as well as heading angle recovering to the initial value.

Fig. 4.4(c) shows the path generated by the proposed method in three dimensions. The red line represents the trajectory of the Aircraft-A, and the arrow illustrates the direction the Aircraft-A heads to. The green lines show the trajectory of UAV-B, where the initial direction is opposite to that of Aircraft-A. As can be seen in Fig. 4.4(c), when UAV-B arrives at the turning point, UAV-B is out of safety bound of Aircraft-A, which means that UAV-B avoids the collision successfully. The generated trajectory of UAV-B is smooth thanks to flatness-based path planning approach.

As indicated in Table 4.1, the minimum distances between Aircraft-A and UAV-B are always greater than the minimum safe separation of $1852m$.



(a) Ground Speed (b) Heading Angle

(c) Trajectories

Figure 4.4: Results in Scenario 1.

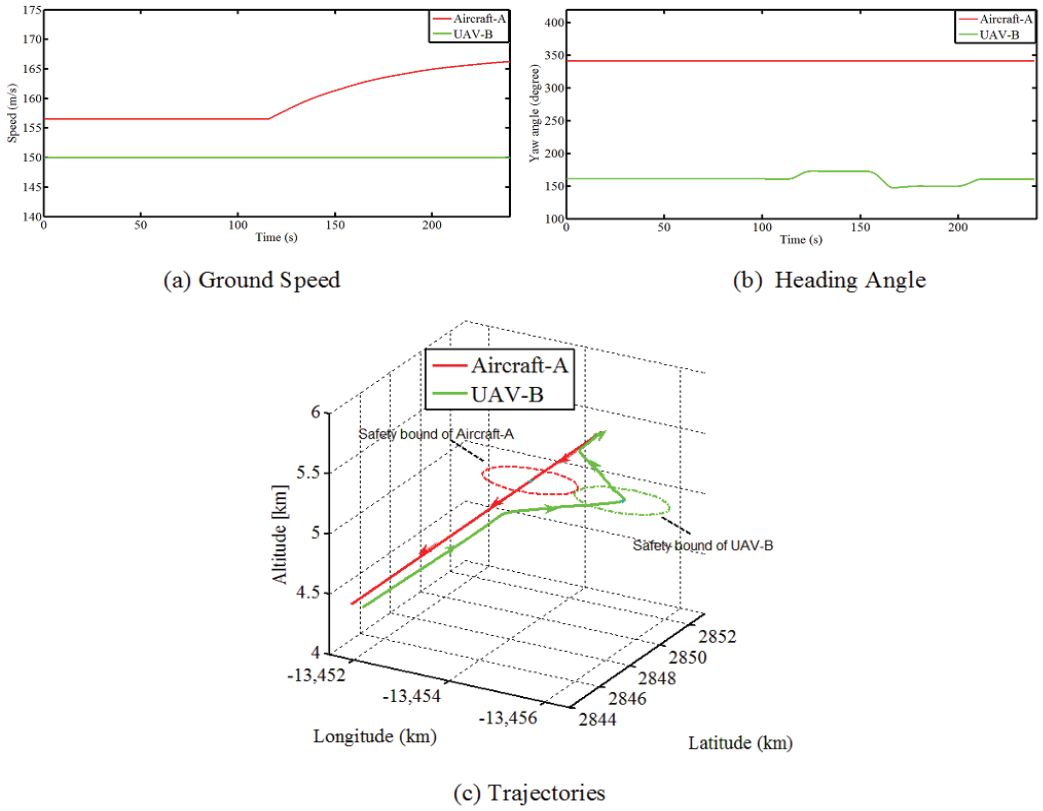
Table 4.1: Minimum relative distance

Scenario	Minimum distance
1	2043 *m*
2	2032 *m*
3	2146 *m*
4	1998 *m*

(2) *Results of Scenario 2*

As shown in Fig. 4.5(a), Aircraft-A speeds up during flight, while UAV-B still keeps the same speed. Based on Fig. 4.5(b), the heading angle of UAV-B is 1° bigger than Scenario 1 at first turning. Eventually, UAV-B flies back to the initial trajectory after last orientation adjustment. From Fig. 4.5(c), one can see that the safety bounds of UAV-B does not

intersect with that of Aircraft-A at the relative minimal distance. Since the relative speed increases, it determines that more aggressive avoidance actions have to be taken.



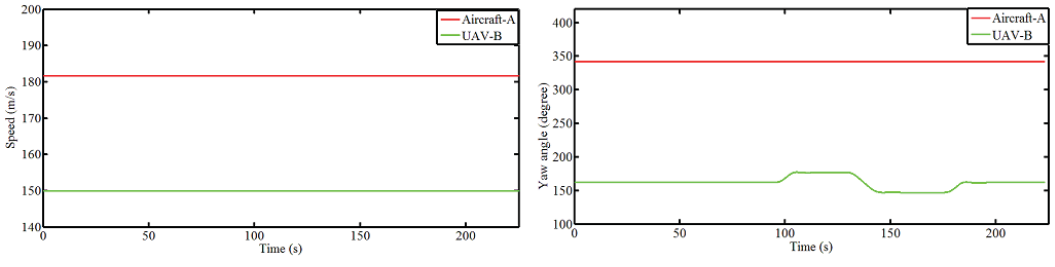
(a) Ground Speed (b) Heading Angle

(c) Trajectories

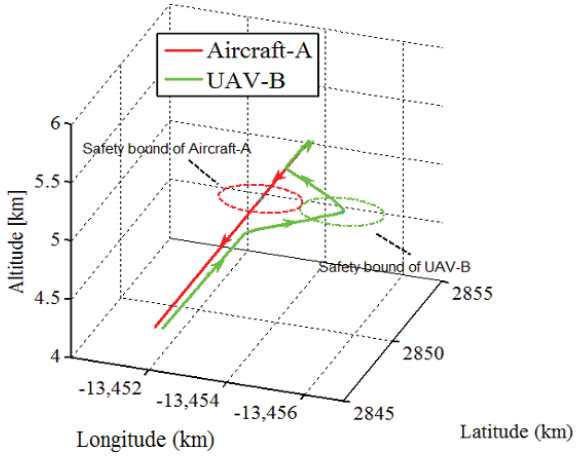
Figure 4.5: Results of Scenario 2.

(3) *Results of Scenario 3*

Fig. 4.6(a) indicates that Aircraft-A flies at a maximum speed of 180 m/s, whilst UAV-B still keeps the same speed. As illustrated in Fig. 4.6(b), because the relative velocity increases significantly, maximal heading angle is 3° bigger than in Scenario 1 to ensure UAV-B avoids the intruder successfully. The radius of the turn of UAV-B is also larger than that of Scenario 1. Based on Fig. 4.6(c), the developed S&A algorithm is capable of ensuring the safe flight of UAV-B in this case.



(a) Ground Speed (b) Heading Angle

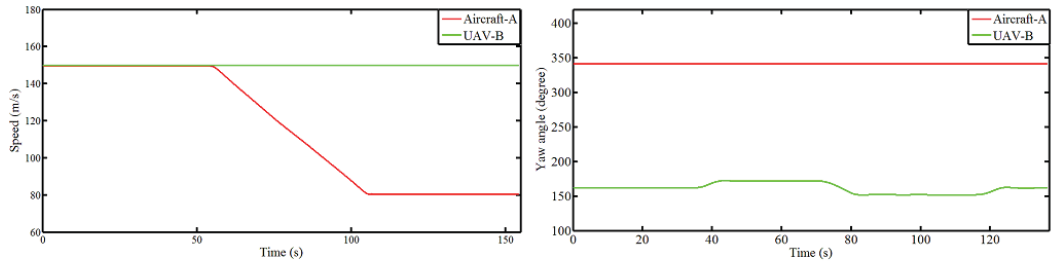


(c) Trajectories

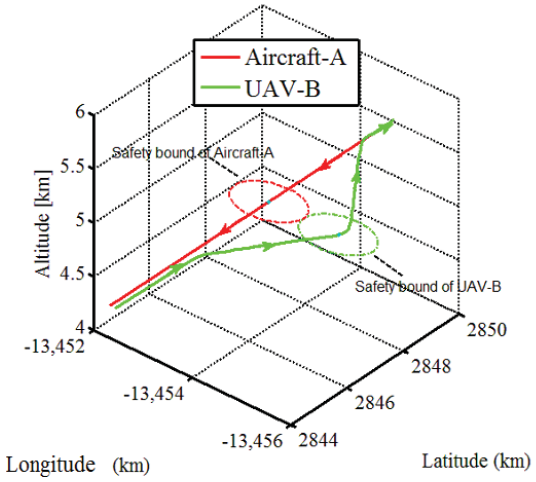
Figure 4.6: Results of Scenario 3.

(4) *Results of Scenario 4*

In Scenario 4, Aircraft-A slows down when UAV-B starts its turning maneuver, as can be observed in Fig. 4.7(a). From Fig. 4.7(b), since relative velocity decreases, maximal heading angle is 4° smaller than that of Scenario 1 to ensure that safety bounds of UAV-B is tangent with that of the intruder. It is indicated in Fig. 4.7(c) that the trajectory of UAV-B is less aggressive than Scenario 1. Thus, the turning radius of UAV-B is smaller than that of Scenario 1.



(a) Ground Speed (b) Heading Angle



(c) Trajectories

Figure 4.7: Results of Scenario 4.

Interested readers are referred to the following NAV Lab YouTube Channel for relevant experimental tests videos: https://www.youtube.com/watch?v=S1vxWBMESUc.

4.4 Summary

This chapter describes the development of a S&A algorithm that can ensure an UAV being able to avoid collisions autonomously. The developed path planner generates the feasible trajectory using differential geometric guidance and flatness techniques, where the constraints of UAV dynamics are considered. The effectiveness of the developed scheme is demonstrated by flight tests carried out at the Marinvent Flight Simulator.

Chapter 5

Sense and Collision Avoidance of Unmanned Aerial Vehicles Using Biogeography Based Optimization Approach

5.1 Collision Avoidance Algorithm

Multiple aircraft flying in a shared airspace, such as landing phase, arise the risk of conflict. Moreover, maintaining safety distance between different airplanes is required for the UAV to complete a safe landing. To address this problem, a minimum cost based S&A strategy is proposed to derive a collision-free trajectory from the initial point to the destination. The procedure of collision avoidance in the landing phase can be separated into two steps, including: 1) BBO-based path planning to produce a flight path with the minimum distance cost; and 2) differential flatness-based approach with consideration of the UAV dynamic constraints to generate feasible reference trajectory. When the environment is represented as a grid-based model, the solution of the collision avoidance problem can be thereby considered as: selecting a set of waypoints for the UAV to minimize the flight distance and maximize the safety distance with intruders. The optimal path directing UAV from the initial position to the pre-arranged destination is produced based on dynamic migration and mutation. A differential flatness approach is then applied such that the physical constraints of the UAV can be respected while the generated trajectory can be feasible.

5.1.1 Biogeography-Based Optimization (BBO)

BBO is a novel population-based optimization algorithm proposed in (Simon, 2008). The basic idea of BBO is influenced by the geographical distribution of species over time and space. Similarly to other artificial heuristic algorithms, the search agent in BBO is called habitat. Each habitat is assigned a vector of habitants, representing as a possible solution for the proposed problem. The fitness and suitability of a habitat are defined by Habitat Suitability Index (HSI). The features of the characterized habitability are called Suitability Index Variables (SIVs), which are equivalent to a gene in GA. In this study, SIVs represent a sequence of discrete waypoints. A solution with high HSI is better than that with low HSI. However, high HSI habitats with large amount of species are more likely to emigrate to the low HSI habitats, while habitats with low HSI tend to immigrate species from high HSI habitats. Motivated by this natural biogeography, sound solutions tend to share their features with poor solutions, and poor solutions are more likely to accept features from good solutions to increase its quality. The movement of species between different habitats is determined by immigration rate (λ) and emigration rate (μ), formulated as the functions of the number of species in the habitat. As illustrated in Fig. 5.1, λ is inversely proportional to the number of species and μ is proportional to the number of species, which are both limited from 0 to 1. S_{max} is the largest number of species that a habitat can keep. The equation for emigration rate (μ_k) and immigration rate (λ_k) can be defined as:

$$\mu_k = \frac{E \times k}{S_{max}}, \tag{38}$$

$$\lambda_k = I \times (1 - \frac{k}{S_{max}}), \tag{39}$$

where k is the number of species in habitat, E, I are the maximum emigration rate and immigration rate, respectively. For the research simplicity, it is assumed that $E = I = 1$, then one can yield

$$\lambda_k + \mu_k = E. \tag{40}$$

The BBO method consists of two main functions: migration (including emigration and immigration) and mutation.

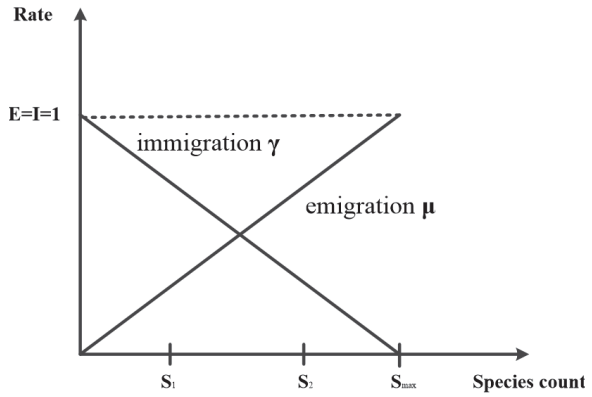


Figure 5.1: Simplified migration model of habitat.

Migration

The evolutionary procedure of migration is to arrange each solution H_i by immigrating and emigrating features between solutions. Each H_i has its own immigration and emigration rates. A good solution has low λ and high μ, while a poor solution has the opposite characters. After initialing a random set of solutions $H = [H_1, H_2, ..., H_n]$, a solution H_i is selected to be modified with the probability that is proportional to immigration rate λ. To share features between solutions, the emigrating solution H_j is selected with the probability that is proportional to emigration rate μ. A randomly selected SIV in solution H_j is choosen to migrate into H_i. Migration can be consequently expressed as:

$$H_i(SIV) \leftarrow H_j(SIV). \tag{41}$$

The emigration operator is used to modify non-elite habitat. The elitism operation is employed to preserve the elite habitat produced in the previous generation by setting immigration rate λ to zero. In such a way, the best solution can be prevented from being replaced after immigration. The detailed procedure of migration is shown in Fig. 5.2.

1. Create a random set of habitat $H_1, H_2, \ldots, H_n$;
2. Compute HSI values;
3. For $i = 1$ to n
4. Select H_i with the probability $\propto \lambda_i$
5. If H_i is selected, then
6. For $j = 1$ to n
7. Select H_j with the probability $\propto \mu_i$
8. If H_j is selected, then
9. Randomly select an SIV from H_j
10. Substitute a random SIV of H_i with the selected SIV in H_j
11. End if
12. End for
13. End if
14. End for

Figure 5.2: Habitat modification of BBO.

Mutation

Mutation is an event that replaces a solution' SIV with a randomly selected SIV based on a mutation rate. The mutation rate is derived by the function of the species count probabilities. The equation of the probabilities of each species is written as:

$$\dot{P}_s = \begin{cases} -(\lambda_s + \mu_s)P_s + \mu_{s+1}P_{s+1}, & S = 0 \\ -(\lambda_s + \mu_s)P_s + \lambda_{s-1}P_{s-1}, & S = S_{max} \\ -(\lambda_s + \mu_s)P_s + \lambda_{s-1}P_{s-1} + \mu_{s+1}P_{s+1}. \\ 1 \leq S \leq S_{max} - 1 \end{cases} \tag{42}$$

The mutation rate of each solution, inversely proportional to the solution probability, is formed as:

$$m_i = m_{max}(1 - \frac{P_i}{P_{max}}), \tag{43}$$

where m_{max}, P_{max}, and P_i are the maximum mutation rate, the maximum probability, and

the solution probability, respectively. Following the mutation process, both high and low HSI solutions are likely to be modified, by which the diversity of the population can be enhanced. The BBO algorithm for mutaion is illustrated in Fig. 5.3.

Remark 1. After a finite number of emigration and mutation in the population, the best solution with the minimum cost is selected as the reference trajectory to evade multiple intruders. Varing values of immigration rate λ and emigration rate μ can improve the exploration ability in global optimization, reducing the computation cost of avoiding multiple collisions. Since features are interacted in the set of solution, the optimal trajectory can be guaranteed in case of trapping in local minima. Randomly selecting a mutation rate decreases the number of grid points searching in the flight space, which is capable of solving multiple collisions resolution in real time. The elitism process prevents elite solutions from disappearing during immigration. As compared to other artificial heuristic algorithms, BBO can provide superior performance, especially in the case of multiple intruders.

Remark 2. As an excellent approach of conflict resolution under uncertainty, Partially Observable Markov Decision Process (POMDP) guarantees UAV safety in uncertain and dynamic environments (Ragi & Chong, 2013). Nevertheless, the disadvantage of POMDP is recognized as the high computational complexity. The dimension of POMDP based solution grows exponentially with the number of grid. As the grid number is increasing, the convergence rate slows down while it becomes significantly difficult to generate an optimal solution. As opposed to POMDP, BBO can provide superior performance in convergence. By assigning different emigration and immigration rate to each habitat, the information between habitats can be shared and the exploration is improved consequently. In the process of generations, the low-HSI habitats accept the habitants living in high-HSI habitats, allowing the high convergence rate of BBO.

1. For $i = 1$ to n
2. For $j = 1$ to m
3. Use λ_i and μ_i to calculate the probability P_i
4. Select $H_i(SIV)$ based on mutation probability m_i
5. If $H_i(SIV)$ is selected, then
6. Substitute $H_i(SIV)$ with randomly adopted SIV
7. End if
8. End for
9. End for

Figure 5.3: Habitat mutation of BBO.

5.1.2 The Integration of UAV Dynamics in the Planned Path

An optimal trajectory planned by BBO may require aggressive maneuvers, such as sharp turning, suddenly climbing to a large altitude. However, each UAV has its own physical constraints, which are indicated by the maximal Euler angular rates, turn radius, and etc. Focusing on the differential flatness algorithm, the profile of the trajectory is tuned to ensure that reference path from an initial position to a final position is followed by UAV smoothly. The physical constraints, including flight envelope constraints and control input constraints, are explicitly considered in this study.

Supposing a UAV to avoid collisions in longitudinal plane, the limit of pitch rate among flight envelope is taken into account firstly. Similar to the flatness-based reference trajectory given in Eq. (30), the reference trajectory is designed as:

$$\theta^*(t) = [1 - (1 + \omega_n t)e^{-\omega_n t}]\theta. \tag{44}$$

In order to generate the parameterization of the pitch rate $\dot{\theta}$ in the flat output, $\dot{\theta}$ is derived according to the time derivative of Eq. (44):

$$\dot{\theta}^*(t) = \omega_n^2 \theta t e^{-\omega_n t}. \tag{45}$$

Based on the process of smoothing trajectory proposed in Chapter 4, the settling time of

the reference trajectory t_s has to satisfy:

$$t_s \geq \frac{5.83\theta}{\dot{\theta}_{max} e}. \tag{46}$$

As indicated in Eq. (27), the stability is influenced by the states (flight envelope) and control inputs. Since the flight envelope is limited by Eq. (46), it is necessary to impose the bounds on the maximal control inputs to avoid the UAV from performing an unrealistic maneuver.

The elevator plays an important role in the pitch angle change. Assuming that $\alpha = 0$, $\theta_m = \theta_d$, $\dot{\theta}_m = \dot{\theta}_d$, $\delta_e^*(t)$ can be simplified to:

$$\delta_e^*(t) = \frac{1}{M_{\theta,\delta_e}}[I_y\ddot{\theta} + \frac{\partial F}{\partial \dot{\theta}} - M_{\theta,\delta_{th}}\delta_{th}]. \tag{47}$$

Similar to the calculation of $\dot{\theta}_{max}$, the first step for $\delta_{e,max}$ is to derive the time when the maximal δ_e will take place. Using the time derivative of Eq. (47), $\dot{\delta_e}^*(t)$ is expressed as:

$$\begin{aligned}\dot{\delta_e}^*(t) = &- \frac{I_y}{2925.47V^2}\dddot{\theta}^*(t) - \frac{94317.23}{2925.47V}\ddot{\theta}^*(t) \\ &+ \frac{2mg\dot{\theta}^*(t)}{2925.47V^2},\end{aligned} \tag{48}$$

where

$$\dddot{\theta}^*(t) = \frac{5.83^3}{t_s}(\frac{5.83}{t_s}t - 2)\theta e^{-\frac{5.83}{t_s}t}. \tag{49}$$

The extrema elevator deflection $\delta_{e,ext}$ can be obtained when $t_{ext} = \frac{(0.4t_s - 4.6)t_s}{0.13t_s^2 + 2.3t_s - 13.6}$. Comparing the value of elevator deflection δ_e at $t = 0, t = t_s$, and $t = t_{ext}$, the maximal can be found at $t = 0$. The elevator deflection value at $t = 0$ is:

$$\delta_e^*(0) = \|\frac{14\theta}{t_s^2} + 0.1\|. \tag{50}$$

From Eq. (50) and to ensure $|\delta_e^*(t)| \leq \delta_{e,max}$, inequality (51) holds:

$$t_s \geq \sqrt{\frac{14\theta}{\delta_{e,max} - 0.1}}. \tag{51}$$

After comparing Eq. (46) and Eq. (51), t_s should satisfy $t_s \geq \frac{5.83\theta}{\dot{\theta}_{max} e}$ to keep UAV's stability during collision avoidance process.

5.1.3 Implementaion of the Proposed BBO with Differential Flatness to Avoid Collisions

In this section, the detailed procedure of BBO with differential flatness approach is introduced to generate a feasible trajectory guiding a landing UAV to avoid multiple threats. As can be seen in Fig. 5.4, eight steps are included in the overall procedure.

• **Step 1:**
Initialize the environment and orderly gridding coordinate. A flight range with $10000m \times 1000m$ is decomposed into a grid array by dividing the flight environment as $500m \times 50m$ squared grids. The initial point of grid is the position where UAV starts running collision avoidance function, supposing $(0, 0)$ is located in the bottom left corner of the grid array. The grid is then numbered in a positive direction on the x-axis (horizontally to the right) and in a positive direction on the z-axis (vertically going up).

• **Step 2:**
Initialize BBO parameters, covering the total habitat size n, maximum generation count T_{max}, number of species in each habitat k, maximum mutation probability m_{max}, maximum immigration rate I, maximum emigration rate E, elitism number of best habitats to keep from one generation to the next D. Randomly initialize a habitat of k waypoints $G = [G_1, G_2, ..., G_k]$ in the longitudinal plane under the flight range. G_1 and G_k represent the pre-defined start point and goal point, respectively. A set G can produce a candidate trajectory. Initialize a set of habitat $S = [S_1, S_2, ..., S_n]$, and each individual corresponds to a potential trajectory that connects the start to the goal one.

• **Step 3:**
Calculate the cost of each candidate trajectory with respect to Eqs. (20), (21), and (22). Sort the habitats from the best performance to the worst one. Select the first D number of solution as elite habitats.

• **Step 4:**
Compute the immigration rate λ_k and emigration rate μ_k for each path according to Eqs. (38) and (39). Initialize the solution probability P_i. Modify the non-elite solution probabilistically with the hybrid immigration and emigration as shown in Fig. 5.2. Re-map the fitness to the series of habitats. Compute the new probabilities for each solution.

• **Step 5:**
Calculate the mutation rate m_i for each trajectory. Sort the solution following the probability P_i. Mutate only the worst half of the solution as described in Fig. 5.3. Re-calculate the cost of path.

• **Step 6:**
Replace the habitats with their new versions. Ensure each individual is legal with respect to the constraints as mentioned in Section II. Calculate the cost of each habitats and order from the best to the worst. Replace the worst solution with the previous generation's elite.

• **Step 7:**
End the generation when T_{max} is reached. Output the optimal solution with cost value.

• **Step 8:**
Characterize the optimal solution by the UAV constraints following the concept of differential flatness and calculate the feasible trajectory.

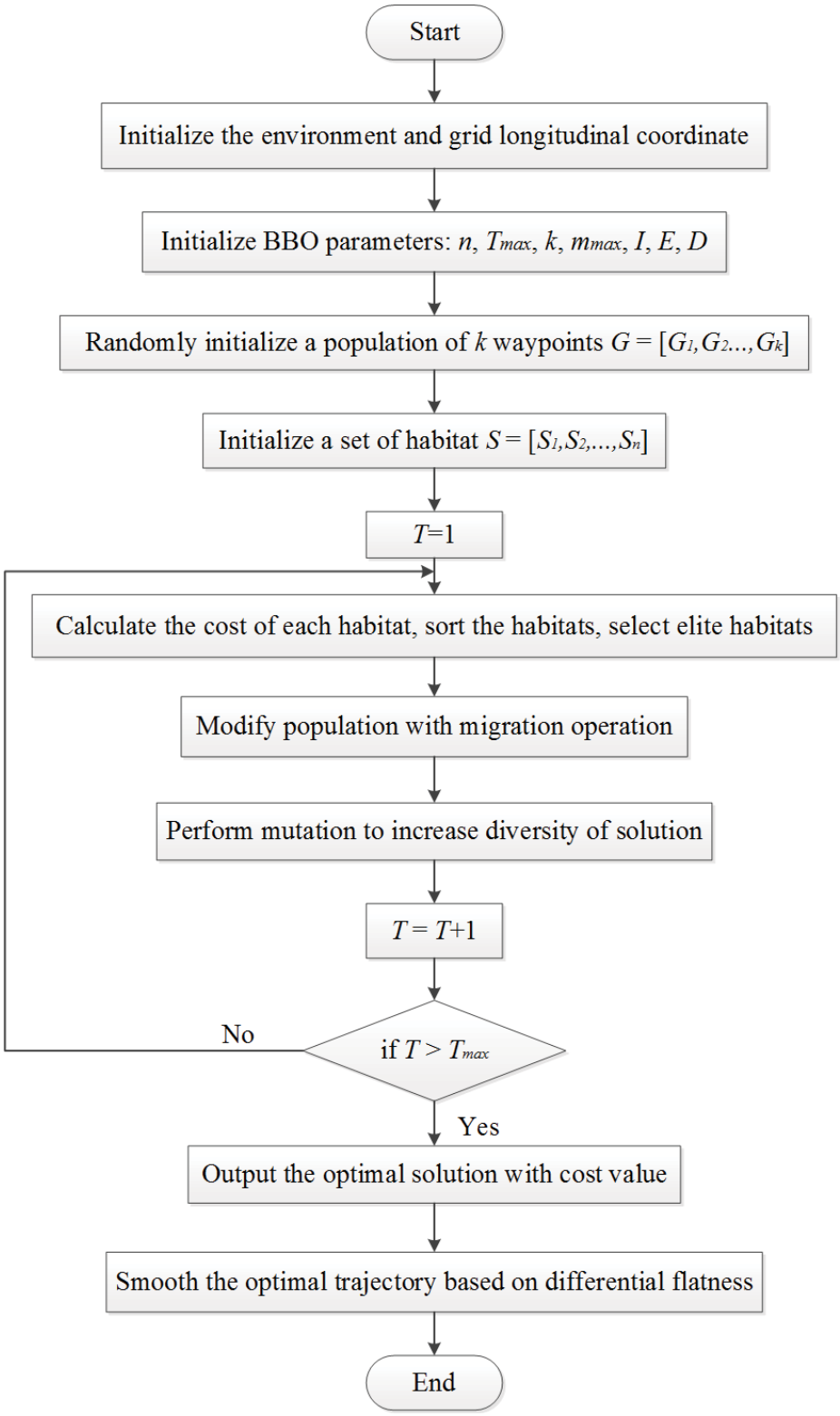


Figure 5.4: The procedure of the proposed approach.

5.2 Path Following Controller Design

Modern UAVs are envisioned to replace human pilots in various missions. In order to accomplish these missions with minimal human intervention, the operation of a UAV should be fully automated from the top level of path planning, down to the inner control loop level. At the top level of the control hierarchy, the collision avoidance algorithm computes a rough approximation of the optimal path toward the goal. A flatness-based algorithm then smooths the path yielding a dynamically feasible, flyable path, by taking into account the dynamics constraints of the UAV. Finally, the path following controller is responsible for guiding the UAV to stay close to the designed path.

5.2.1 Controller Design

Energy is one of the fundamental concepts in science and engineering practice, where it is common to view dynamical systems as energy-transformation devices. This perspective is particularly useful in studying complex nonlinear systems by decomposing them into simpler subsystems that, upon interconnection, add up their energies to determine the full system's behavior. The action of a controller may also be understood in energy terms as another dynamical system typically implemented in a computer interconnected with the process to modify its behavior. The control problem can then be recast as finding a dynamical system and an interconnection pattern such that the overall energy function takes the desired form (Ortega, Van der Schaft, Mareels, & Maschke, 2001).

The energy-based control design method is adopted to modify energy function of the UAV to track the reference trajectory. The resulting controller is relatively simple with guaranteed stability and performance robustness. The energy-based control design method provides a more natural way to control systems as it is derived from easily understood energy principles. Similar to the methodology developed in (Akmeliawati & Mareels, 2010), the procedure of controller design consists of 1) designing an inner-loop control to minimize the tracking error of pitch angle θ and pitch rate $\dot{\theta}$; 2) developing an energy-loop to meet the performance criteria for energy tracking in slow dynamics; and 3) implementing an outer-loop to eliminate the altitude tracking error. The structure of the energy-based controller is shown in Fig. 5.5.

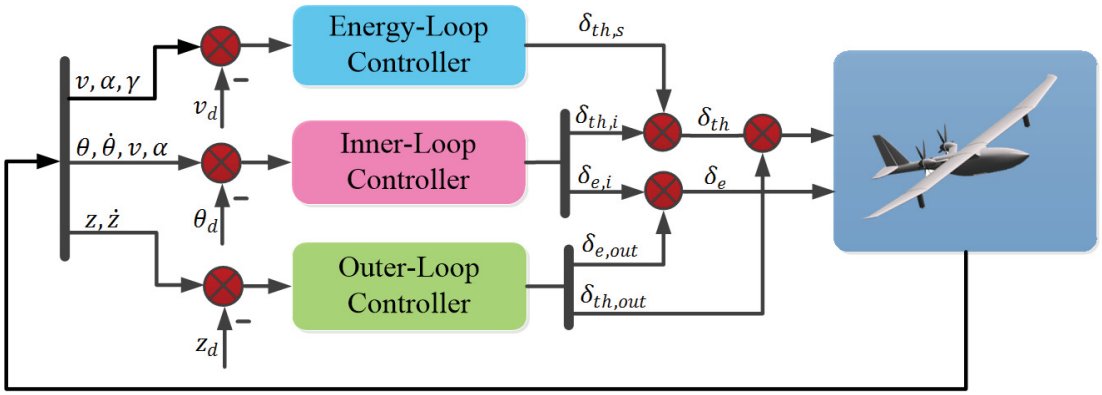


Figure 5.5: The structure of the energy-based controller.

Inner-Loop Control

The objective of inner-loop control design is to eliminate the tracking error of pitch angle θ, which can be modeled as:

$$\Upsilon_i = I_y \dot{e}_i + K_d e_i \longrightarrow 0, \tag{52}$$

where $e_i = \dot{\theta} - \dot{\theta}_d + \zeta(\theta - \theta_d)$. The inner-loop controller is given as:

$$\begin{aligned} \delta_{e_i} = & \frac{1}{M_{\theta,\delta_e}} [I_y \ddot{\theta}_d + \frac{\partial F}{\partial \dot{\theta}} - (I_y \zeta + K_d)(\dot{\theta} - \dot{\theta}_d) \\ & - K_d \zeta(\theta - \theta_d) - M_{\theta,\delta_{th_i}}], \end{aligned} \tag{53}$$

$$\delta_{th_i} = -K_5(\theta - \theta_d) - K_6(\dot{\theta} - \dot{\theta}_d), \tag{54}$$

where $\zeta, K_d \in R^+, K_5, K_6 \in R$. Combining Eq. (52), Eq. (53), Eq. (54) and Eq. (1), $\Upsilon_i = 0$ can guarantee that $\theta \longrightarrow \theta_d$, $\dot{\theta} \longrightarrow \dot{\theta}_d$. On basis of Eq. (1), the desired energy function for the inner-loop is:

$$H_i = \frac{1}{2} I_y e_i^2. \tag{55}$$

For the convergence of tracking error, it is required that the time derivative of energy function is no greater than zero. In order to prove that the desired pitch angle θ_d and desired pitch rate $\dot{\theta}_d$ are achieved following the proposed inner-loop controller, the time derivative of Eq. (55) is:

$$\begin{aligned} \dot{H}_i &= I_y \dot{e} e \\ &= I_y [\ddot{\theta} - \ddot{\theta}_d + \zeta(\dot{\theta} - \dot{\theta}_d)][\dot{\theta} - \dot{\theta}_d + \zeta(\theta - \theta_d)]. \end{aligned} \tag{56}$$

Substituting Eq. (52) into Eq. (56), $\dot{H}_i \leq 0$ is derived to satisfy the requirement of energy objective.

Energy-Loop Control

The energy-based controller to guarantee energy tracking is written as:

$$\begin{aligned}\delta_{th_s} =& \frac{1}{19.62\cos(\alpha)}[\frac{D_r}{m} + g\sin(\gamma) \\ &+ \dot{V}_d - K_{ds}(V - V_d)],\end{aligned} \tag{57}$$

where $K_{ds} > 0$, $D_r = 25.46V^2 + 337.21V^2\alpha^2 + 80.20V^2\alpha$. The objective function of energy tracking is introduced as:

$$\Upsilon_s = \dot{e_s} + K_{ds}e_s \longrightarrow 0, \tag{58}$$

where $e_s = V - V_d$. Using Eq. (6), Eq. (7) and Eq. (57), $\Upsilon_s = 0$ is derived to ensure that $V \longrightarrow V_d$. In accordance with Eq. (6) and Eq. (7), the desired energy function of energy-loop is:

$$H_s = \frac{1}{2}m(V - V_d)^2. \tag{59}$$

The time derivative of Eq. (59) is:

$$\begin{aligned}\dot{H}_s &= m(V - V_d)(\dot{V} - \dot{V}_d) \\ &= m(V - V_d)(\Upsilon_s - K_{ds}e_s) \\ &= -mK_{ds}(V - V_d)^2.\end{aligned} \tag{60}$$

Thus, it is known that $\dot{H}_s \leq 0$.

Outer-Loop Control

With respect to tracking error in z-position, outer-loop controller to achieve altitude stabilization is introduced as:

$$\begin{aligned}\delta_{e,out} =& K_1(z - z_d) + K_2\tanh(0.1\int(z - z_d)) \\ &+ K_3(\dot{z} - \dot{z_d})\end{aligned} \tag{61}$$

$$\delta_{th,out} = K_4(z - z_d), \tag{62}$$

where $K_1, K_2, K_3, K_4 \in R$.

Remark 3. Energy-based control design formulates the UAV control in the form of the energy equation following the principle of conservation of energy. The control design is applied to track the reference trajectory and keep energy balance without the need of gain-scheduling.

5.3 Simulation and Results Analysis

5.3.1 Simulation Scenarios

The proposed approach is validated by numerical simulation conducted in the MATLAB/Simulink environment with a PC that is configured with a 2 GHz Intel Core i7-4510U processor with 8 GB RAM. The landing phase is explicitly considered in the simulation study, where the reference signals of the pitch angle and altitude are indicated in Table 5.1 and Table 5.2, respectively. The UAV follows the initial conditions as shown in Table 5.3. For aircraft presented as intruders in this study, it is required to follow the original trajectory with a constant airspeed of $60\ m/s$. The initial parameters of BBO algorithm are given as:

- Habitat size: $n = 30$;
- Maximal genereation count: $T_{max} = 500$;
- Number of species in each habitat: $k = 20$;
- Maximum mutation probability: $m_{max} = 0.005$;
- Maximum immigration rate: $I = 1$;
- Maximum emigration rate: $E = 1$; and
- Elitism parameter: $D = 2$.

Table 5.1: Reference pitch angle describtion

Level	Time (s)	Pitch angle (deg)	Velocity (m/s)
1	$0 < t \leq 110$	0	80
2	$110 < t \leq 150$	$-5\times(t-110)/40$	80
3	$150 < t \leq 190$	-5	80
4	$190 < t \leq 230$	$-5+10\times(t-190)/40$	80
5	$230 < t \leq 260$	-5	80
6	$260 < t \leq 300$	$5-10\times(t-260)/40$	80
7	$300 < t \leq 340$	0	80
8	$340 < t \leq 380$	$-6\times(t-340)/40$	80
9	$380 < t \leq 420$	-6	80
10	$420 < t \leq 450$	$-6+3\times(t-420)/30$	80
11	$450 < t \leq 500$	-3	80

Table 5.2: Reference altitude describtion

Level	Time (s)	Altitude (m)	Velocity (m/s)
1	$0 < t \leq 110$	1000	80
2	$110 < t \leq 190$	$1000-500\times(t-110)/80$	80
3	$190 < t \leq 300$	$500+400\times(t-190)/110$	80
4	$300 < t \leq 380$	900	80
5	$380 < t \leq 500$	$900-900\times(t-380)/120$	80

Table 5.3: The initial parameters for simulation

Parameter	Value
Initial x position	1000 m
Initial z position	1000 m
Initial velocity along x-axis	80 m/s
Initial velocity along z-axis	0 m/s
Initial pitch angle	1.627 deg
Initial pitch rate	0
Airspeed	80 m/s

5.3.2 Performance Analysis

Trajectory Generation for S&A

Figure 5.6 demonstrates that the scenario in which potential threats are detected, the UAV in the landing phase turns to adjust the attitude to avoid collisions, following a feasible trajectory produced by the BBO-based algorithm. During the landing period, the collision detection algorithm is employed to determine when and where the collisions will occur. According to the currently predicted conflict position, BBO is carried out to produce an optimal solution by migration and mutation of each generation. The purple line in Fig. 5.6 illustrates that the UAV successfully avoids three coming intruders and climbs to the pattern altitude waiting for instructions from control tower for landing again. Figure 5.7 presents the actual trajectory of UAV in longitudinal plane. Since no threat is detected before x position of 14000 m, the trajectory performing regular landing in this period is smooth. As the UAV continues to fly east, it switches to collision avoidance mode and finally arrives at the required altitude.

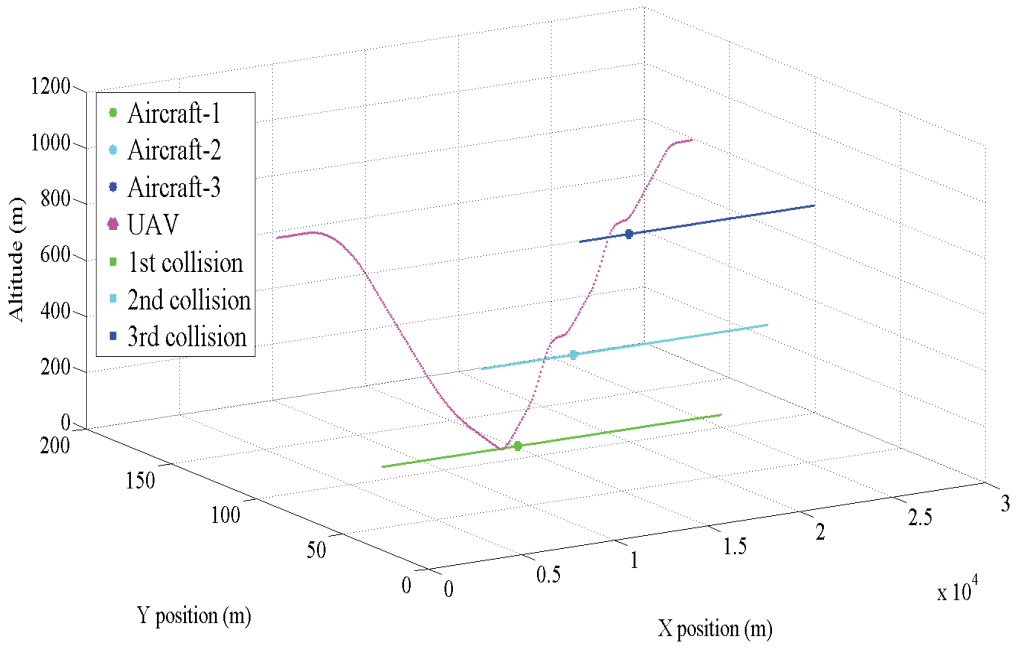


Figure 5.6: The UAV-intruders collision avoidance scenario.

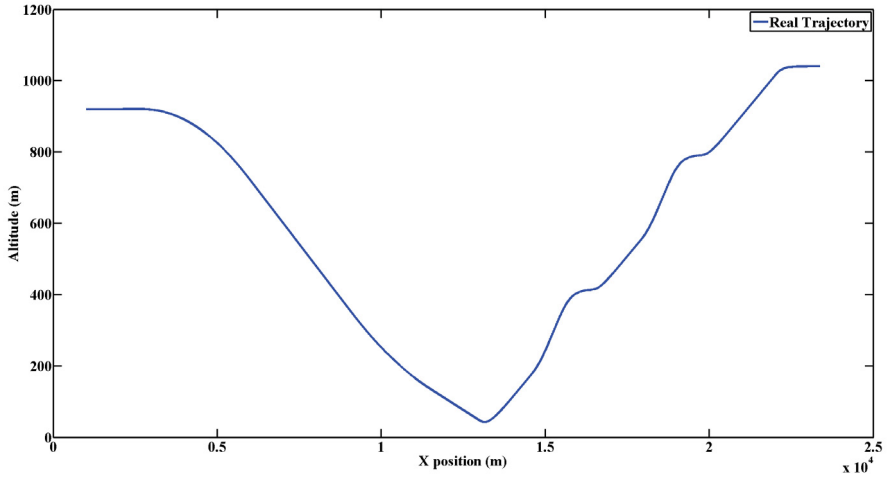


Figure 5.7: Actual trajectory of UAV in collision avoidance scenario.

Path Following Performance

- **Result for following the reference pitch angle in landing phase:** The response for this scenario are shown in Fig. 5.8 and Fig. 5.9. Figure 5.8(a) is the response of the pitch angle. The measured pitch angle can track the reference one, which indicates that the inner-loop controller can handle variation of the pitch angle without overshoot. Figure 5.8(b,c,d) indicates that the velocity, altitude, flight path angle do not break the limits. Under supervision of the inner-loop controller, the elevator deflection and throttle angle exhibited in Fig. 5.9 are not satuated in the suddenly climbing and descending maneuvers.
- **Result for following the reference altitude in landing phase:** As can be seen from Fig. 5.10(a), the outer-loop controller can maintain the altitude tracking performance even in the sharp climbing and descending maneuvers. Figure 5.10(b,c,d) reveals that the velocity, the pitch angle and the flight path angle are controlled to return to the intended values after the transition maneuver in altitude. As shown in Fig. 5.11, the deviation of elevator deflection and throttle angle achieved by the proposed outer-loop controller are within the allowable operating ranges.
- **Result for collision avoidance scenario:** Considering the measurement noise, the angular position with the magnitude 0.01 *rad* of white noise and the velocity with the magnitude 1.5 *m/s* of white noise are conducted in the simulation studies. Following the differential flatness-based re-planning algorithm in the longitudinal plane, the

tracking performance of pitch angle is displayed in Fig. 5.12. The blue line (actual pitch angle) in Fig. 5.12 is fairly close to the red line (desired pitch angle). This figure confirms that the UAV follows the commanded pitch angle quite well.

The responses of altitude and velocity are illustrated in Fig. 5.13. When the pitch angle is increased, the UAV is forced to climb up, conversely the UAV has to descend as shown in Fig. 5.13(a). Fig. 5.13(b) exhibits that velocity of UAV is stable in the entire flight under the energy-loop controller.

From Fig. 5.14, it is clear that even the UAV involved in aggressive maneuvers with large pitch angles, elevator angle and throttle angle are not saturated. Such a desired performance is contributed by the differential flatness-based algorithm with consideration of control input constraints.

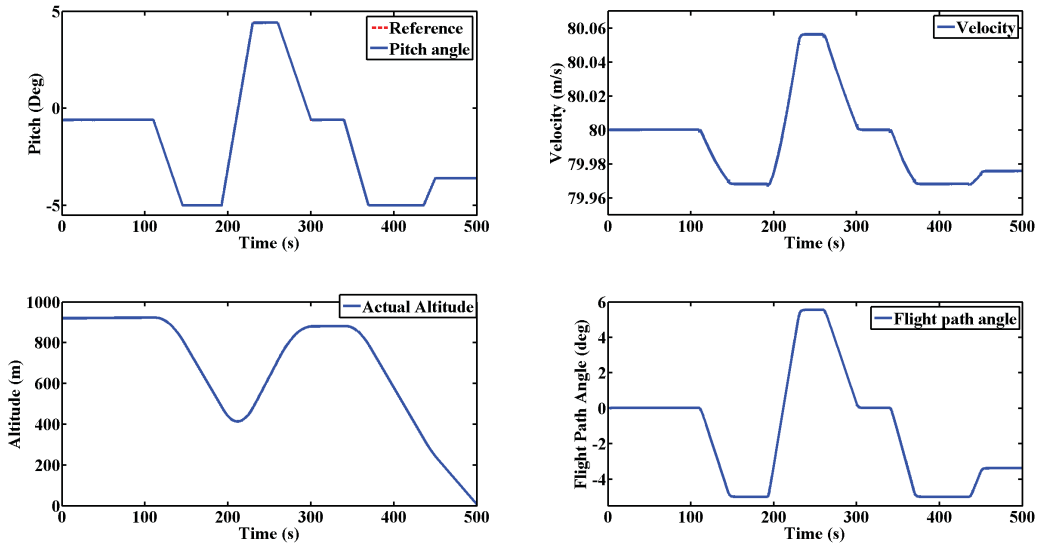


Figure 5.8: UAV output responses in following reference pitch angle scenario.

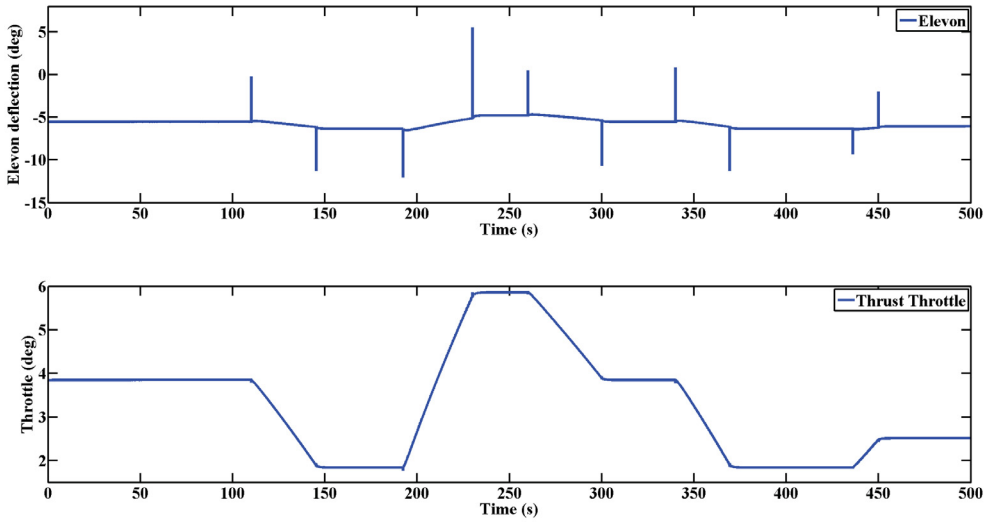


Figure 5.9: Elevator angle and throttle angle responses in following reference pitch angle scenario.

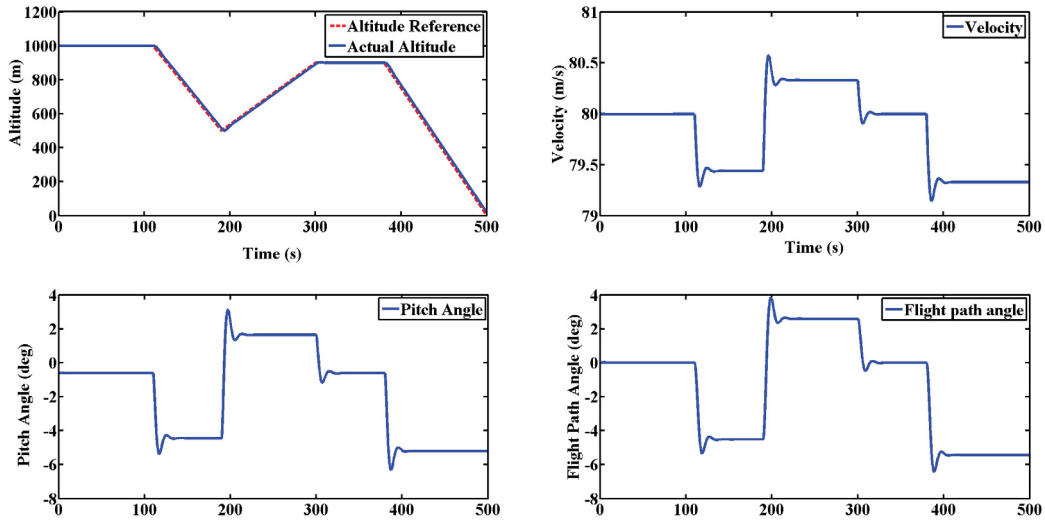


Figure 5.10: UAV output responses in following reference altitude scenario.

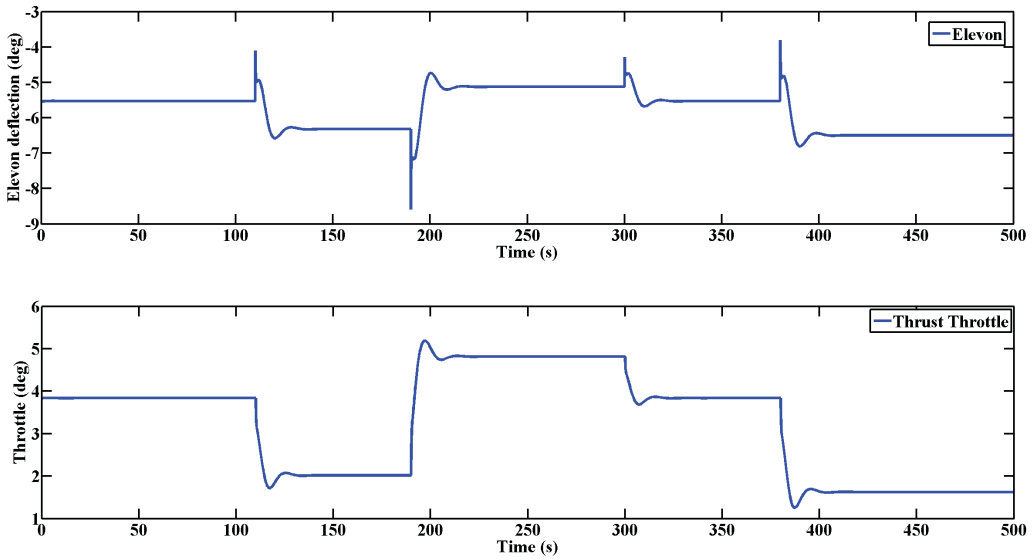


Figure 5.11: Elevator angle and throttle angle responses in following reference altitude scenario.

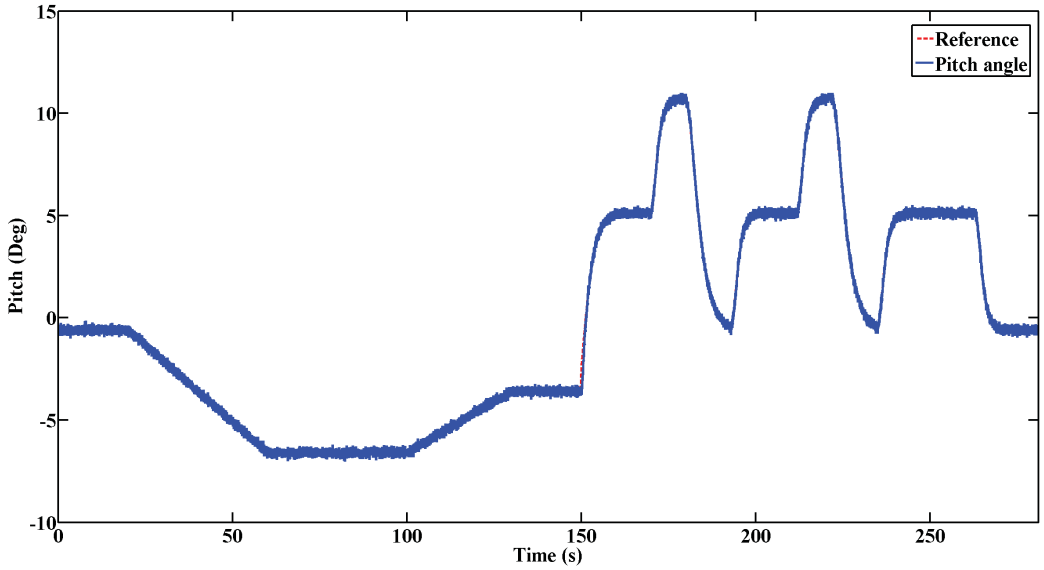


Figure 5.12: Pitch angle of UAV in collision avoidance scenario.

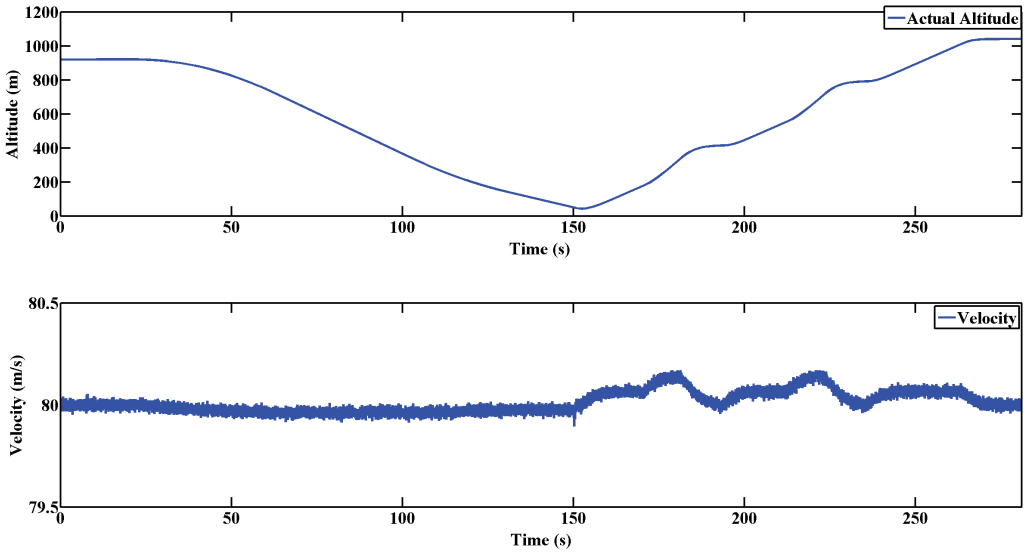


Figure 5.13: Actual altitude and velocity in collision avoidance scenario.

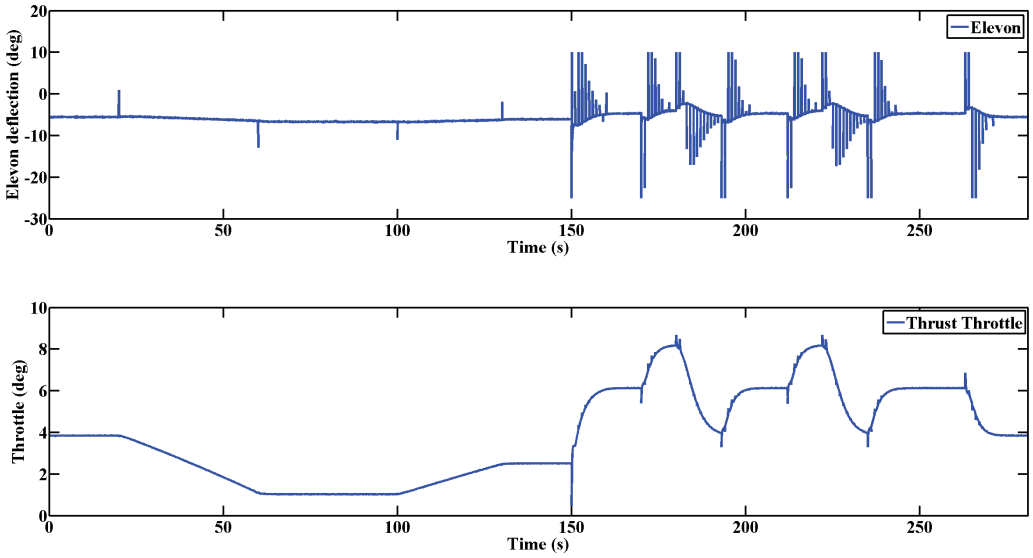


Figure 5.14: Actual elevator angle and throttle angle in collision avoidance scenario.

For interested readers, the demonstration of sense and collision avoidance strategy for the landing phase performed in this work can be found in the Google Shared Link: https://drive.google.com/file/d/0B3iMpBBeeEYPUkNJdWROWHFGSzQ/view?usp=sharing

5.4 Summary

In this chapter, a novel algorithm for UAV avoiding multiple intruders' threats in the landing phase is proposed. The generated trajectory ensures the maximum safety with the minimum cost of the UAV. To get the feasible trajectory, the flight space is modeled as gridded environment as the first step, in which the BBO approach is exploited to generate the resolution logic in the presence of multiple intruders. Secondly, considering that the UAV safety is directly related to the dynamic constraints, the differential flatness technique is developed to smoothen the optimal trajectory. Lastly, energy-based controller is designed such that the UAV is capable of following the generated trajectory. Numerical simulation studies are carried out to illustrate the effectiveness of the developed scheme.

Chapter 6

Sense and Collision Avoidance of Unmanned Aerial Vehicles Using Markov Decision Process and Flatness Approach

In this chapter, two collision avoidance algorithms are presented to ensure that UAV can avoid multiple intruders autonomously.

In first section, the scheme of collision avoidance in 2D plane can be separated into two steps, including: 1) MDP-based path planning to produce an optimal trajectory, such that multiple threats can be avoided; 2) differential flatness-based algorithm with consideration of UAV dynamics to ensure the planned trajectory is feasible. In this work, the airspace is represented as a gridding system while each grid point is regarded as a waypoint. With the implementation of MDP, the optimal path of UAV is produced based on the current states of UAV and multiple threats. The collision avoidance is to select a set of waypoints for the UAV maximizing the safety distance. Following the MDP-based reference trajectory, UAV is required to avoid collision between multiple intruders, while the physical constraints are respected.

In second section, the MDP framework is extended to derive an optimal trajectory in 3D plane. The replanned path is smoothened by the differential flatness-based approach under the physical constraints of UAV. Energy-based controller is then applied such that the UAV can follow the generated trajectory in the 3D plane.

6.1 MDP Based Collision Avoidance in 2D Plane

6.1.1 Markov Decision Process

Markov Decision Processes (MDP) was introduced to the scientific community from Bellman in 1957 (Puterman, 1994). An innovative technique is represented to solve decision problems splitting a complex problem in successive decision stages and providing the optimality of the whole solution in the last stage. MDP has been applied to a wide set of optimization problems from computer vision to bioengineering and from air traffic control to computer networks. Research on autonomous agents exploited MDP trying to implement decision making into artificial intelligences. Work has been done in this direction trying to use MDP to manage path planning and collision avoidance problems.

To implement an MDP, a stochastic model addresses the decision problem by discretizing the system into a finite number of states and actions. MDP is defined as $M =< S, A, T, R >$, where S, A, T, and R denote the system's states, actions, transition function, and reward function, respectively. At each time step, the system takes an action $a \in A$ to move from a state $s \in S$ to $s' \in S$ and receives a real-valued reward $R(s, a)$. The reward function in this work is influenced by the position of the intruders and target. The goal of solving the collision avoidance problem is transformed to choose a sequence of actions that maximizes the expected total reward.

To generate an available trajectory, these units with respect to the flight requirement and environment are defined. The parameters of MDP in this work are designed as:

- **States:** A flight range with 800 m×800 m is decomposed into a grid array by gridding the flight environment as 20 m×20 m square grid. Based on Cartesian coordinate method to identify grids, the initial point (0,0) is in the upper left corner of the grid array. The grid is then numbered in a positive direction on the x-axis (horizontally to the right) and in a positive direction on the y-axis (vertically going down).
- **Actions:** As shown in Fig. 6.1, UAV can move in eight directions, including: North, North-East, East, South-East, South, South-West, West, North-West.
- **Transition Function:** $S \times A \times S \rightarrow$ [0,1] is the state transition model for the discretized state space. The probability of transitioning from state s to s' after action a is written as:

$$P^a_{ss'} = Pr\{s_{t+1} = s' | s_t = s, a_t = a\}. \tag{63}$$

  The state-transition function specifies the next-state distribution given an action at a current state. Fig. 6.2 describes one scenario of probability transition. Assuming

state s of a UAV at the center of this grid array, if the UAV is moving to the North with 0.9 probability, then the actions to NE and NW with 0.04 probability while to E and W with 0.01 probability, respectively. The UAV has eight probability transitions.

- **Reward Function:** The immediate reward when taking action a in state s is defined as:

$$R^a_{ss'} = E\{r_{t+1}|s_t = s, a_t = a, s_{t+1} = s'\}. \tag{64}$$

The reward function denotes the reward of taking an action in a given state. It is convenient to define the reward function for each object separately. Comparing the safe distance with Euclidean distance between the UAV and the intruder, the reward of intruders is defined as:

$$r_1(s) = \sum_{i=1}^{n}(d_i - d_{safe}) \times w_1, \tag{65}$$

where d_i is Euclidean distance between the UAV and the ith intruder, w_1 is a weighting factor for threats reward function. The reward function of the UAV to the goal is

$$r_2(s) = (D_{max} - D_{goal}) \times w_2, \tag{66}$$

where D_{max} is the maximum sensing range of the sensor onboard the UAV, D_{goal} is Euclidean distance between the UAV and the goal, w_2 is a weighting factor for the goal reward function. Eq. (66) implies a negative reward for the target not in the UAV's sensing range.

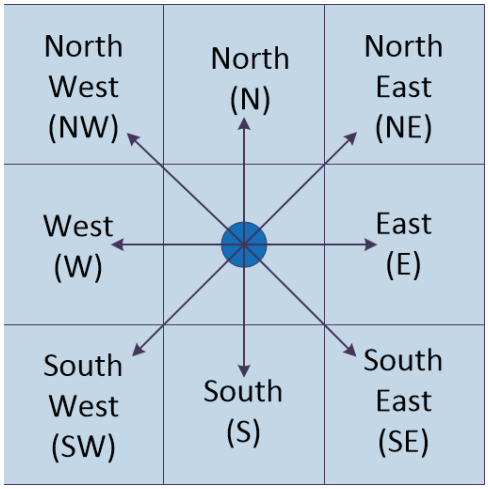


Figure 6.1: The basic action of UAV.

A policy π is a mapping from each state s and action a to the probability $\pi(s,a)$ of taking action a under the current state s. The time-accumulative value function under a policy π when starting from the state s is derived as:

$$\begin{aligned} V^{\pi}(s) &= E_{\pi}\{R_t|s_t = s\} \\ &= E_{\pi}\{\sum_{k=0}^{\infty}\lambda^k r_{t+k+1}|s_t = s\} \\ &= \sum_{a}\pi(s,a)\sum_{s'}P^{a}_{ss'}[R^{a}_{ss'} + \gamma V^{\pi}(s')], \end{aligned} \tag{67}$$

where $\gamma \in$ [0,1] is a discount factor to denote the effect of the future rewards on present decisions. $\pi(s,a)$ is the probability of taking action a in state s under policy π. $E_{\pi}\{\}$ denotes the expected value following policy π , t is a step time and s' is the next state followed by the current state s when action a is applied.

The expected value of taking action a in a state s under policy π is:

$$\begin{aligned} Q^{\pi}(s,a) &= E_{\pi}\{R_t|s_t = s, a_t = a\} \\ &= E_{\pi}\{\sum_{k=0}^{\infty}\lambda^k r_{t+k+1}|s_t = s, a_t = a\}. \end{aligned} \tag{68}$$

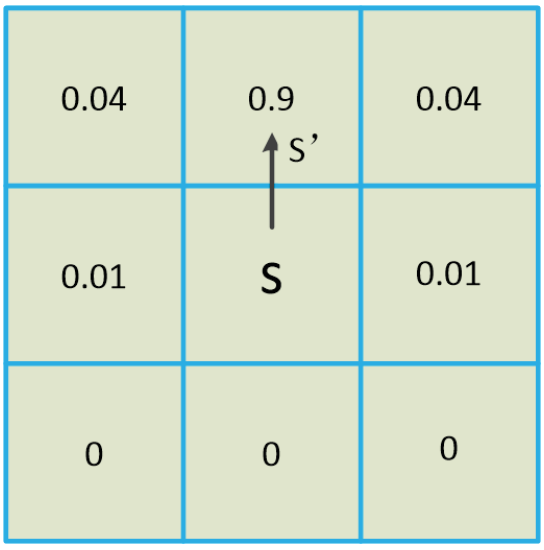


Figure 6.2: The assignment of the transition probability.

6.1.2 Policy Iteration

With respect to the initial policy and initial value functions, policy iteration is adopted to converge to an optimal policy. The policy iteration algorithm is carried out by the following steps:

Policy Evaluation

Policy evaluation is denoted as computing the state value function V^{π} for an arbitrary policy π. A sequence of approximate value functions is obtained as:

$$\begin{aligned} V_{k+1}(s) &= E_{\pi}\{r_{r+1} + \gamma V_k(s_t + 1) | s_t = s\} \\ &= \sum_{a} \pi(s,a) \sum_{s'} p^{a}_{ss'}[R^{a}_{ss} + \gamma V_k(s')]. \end{aligned} \tag{69}$$

The sequence $\{V_k\}$ can converge into V_{π} as $k \to \infty$ under the same initial conditions, by which the existence of V^{π} is ensured. To generate the successive approximation from V_k to V_{k+1}, the policy evaluation based operation is adopted to each state s: the new value obtained from the old value of the successor states of s' is substituted for the old value of s.

Policy Improvement

The process of making a new policy to obtain a higher value function is called policy improvement. Let π and π' be any pair of deterministic policies, if

$$Q^{\pi}(s, \pi'(s)) \geq V^{\pi}(s). \tag{70}$$

Moreover, the value under policy π', $V^{\pi'}(s)$, is not smaller than the value under policy π, $V^{\pi}(s)$. Then the greedy policy π' must be no worse than π. The greedy policy π' is given as:

$$\begin{aligned} \pi'(s) &= argmax Q^{\pi}(s,a) \\ &= argmax \sum_{s'} P^{a}_{ss'}[R^{a}_{ss'} + \gamma V^{\pi}(s')]. \end{aligned} \tag{71}$$

Assuming that a better policy π_1 can be derived from the initial policy π_0 using V^{π}, an even better policy π_2 can be achieved by computing $V^{\pi'}$. A sequence of improving policies and value functions is produced as:

$$\pi_0 \xrightarrow{PE} V^{\pi_0} \xrightarrow{PI} \pi 1 \xrightarrow{PE} V^{\pi_1} ... \xrightarrow{PI} \pi^* \xrightarrow{PE} V^*, \tag{72}$$

where PE defines a policy evaluation and PI is a policy improvement. Owing to the finite number of MDP, this process must finally converge to an optimal policy and optimal value function in a finite number of iterations.

Based on the Bellman function (Sutton & Barto, 1998), the optimal control policy π^* can be produced by the objective function, which maximizes the expected reward:

$$\pi^*(s) = argmax(R(s,a) + \lambda \sum_{s' \in S} (P(s'|s,a)V^*(s')), \tag{73}$$

where $V^*(s')$ is the optimal objective function as:

$$\begin{aligned} V^*(s) &= maxQ^{\pi^*}(s,a) \\ &= max \sum_{s'} P^a_{ss'}[R^a_{ss'} + \gamma V^{\pi}(s')]. \end{aligned} \tag{74}$$

6.1.3 The Integration of UAV Dynamics in the Planned Path

The optimal trajectory produced by MDP may require aggressive maneuvers for UAV to avoid intruders. However, UAVs have their physical constraints in practice, which are indicated by the maximal Euler angular rates, turning radius, and etc.. Based on the differential flatness approach, the profile of the trajectory is tuned to make sure that UAV follows the planned path smoothly. Due to the 2D flight scenario developed in this work, only the limit of the roll angle is taken into account. Following the flatness-based reference trajectory given in Eq. (30), the parameterization of ϕ^* in the flat output can be presented as:

$$\phi^* = atan(\frac{5.83^2 V_a \phi t e^{-\frac{5.83}{t_s}t}}{g t_s^2}). \tag{75}$$

And the time derivative of the roll angle is derived as:

$$\dot{\phi^*} = \frac{\frac{5.83^2}{t_s^2}\phi V_a g e^{-\frac{5.83}{t_s}t}(1 - \frac{5.83}{t_s})}{\frac{5.83^4}{t_s^4}\phi^2 t^2 V_a^2 e^{-2\frac{5.83}{t_s}t} + g}. \tag{76}$$

To determine the time where the maximal roll angle is required, it is necessary to calculate the extrema of the roll angle by setting Eq. (76) to zero at $t = t_s/5.83$:

$$\phi^*_{ext} = atan(\frac{5.83\phi V_a e^{-1}}{gt_s}), \tag{77}$$

where the extrema denotes the maximal or the minimal of the function. It is of importance to check the value of the roll angle at the beginning and at the end of the mission. After comparing these three solutions, when $t = t_s/5.83$, the corresponding solution is the maximal one. Finally, to ensure that $|\phi^*| \leq \phi_{max}$, t_s has to satisfy:

$$t_s = \frac{5.83\phi V_a e^{-1}}{gtan(\phi_{max})}. \tag{78}$$

Based on the obtained solution t_s, the re-planned MDP-based optimal trajectory must be less than or equal to the maximal allowable constraints.

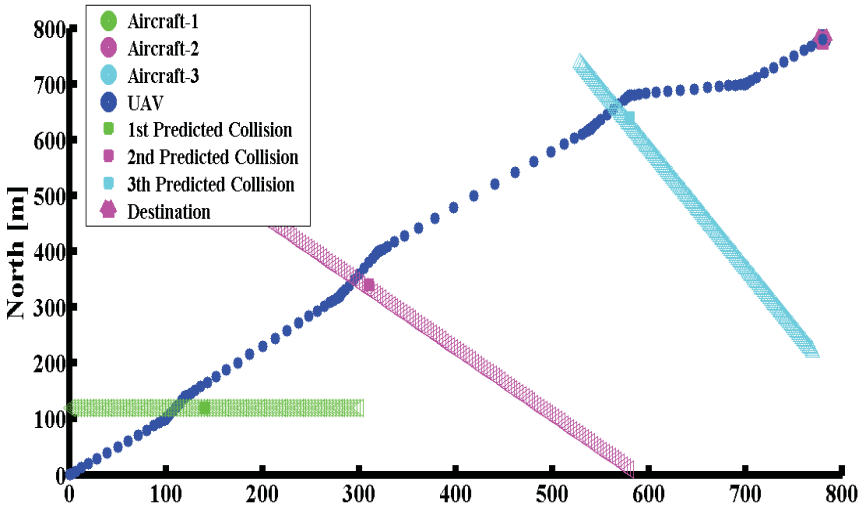


Figure 6.3: The UAV - intruders collision avoidance scenario.

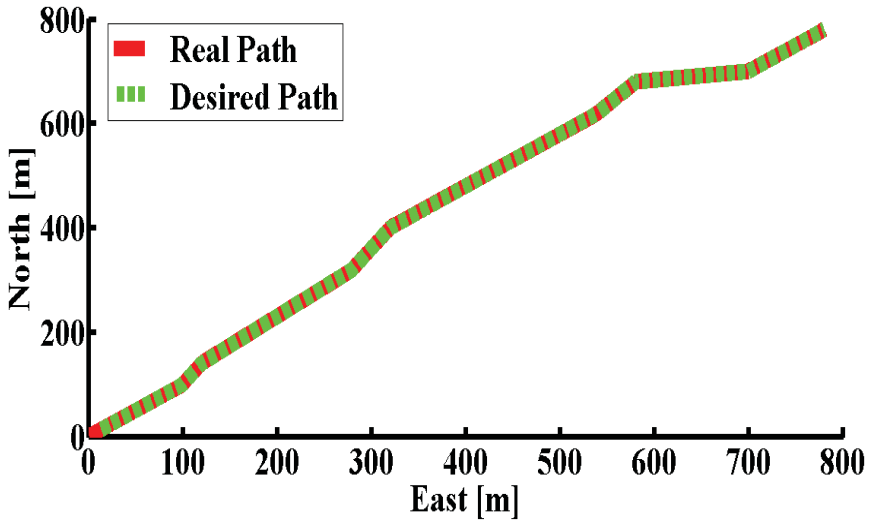


Figure 6.4: Trajectory of the UAV.

6.1.4 Validation

The proposed algorithm is validated by numerical simulation in a PC that is configured with a 2.0 GHz Intel Core i7-4510U processor with 8 GB RAM. It is assumed that a UAV is cruising at 20 m/s, while three intruders are following the original trajectory at a speed of 15 m/s without avoidance maneuver.

Fig. 6.3 shows the scenario in which three conflicts are detected, MDP-based algorithm produces a feasible trajectory for the UAV to avoid collisions. During the flight, the collision detection algorithm is adopted to determine when and where the collisions will occur. Based on the currently predicted conflict position and goal waypoint, the state, action, transition function and reward function of flight system are derived. MDP is therefore deployed to produce an optimal policy by calculating the policy iteration. The blue path in Fig. 6.3 illustrates that the UAV with the developed S&A system successfully avoids three coming threats and reaches the destination finally.

Fig. 6.4 indicates the trajectory for the UAV using the differential flatness-based replanning algorithm in the lateral direction. The plot shows the two-dimensional trajectory of the UAV (red line) with a desired trajectory (green line). As shown in Fig. 6.4, the red line closely follows the green line. This result demonstrates that UAV can follow the desired trajectory quite well.

The tracking position errors in x direction and y direction, which are generated by comparing the real position with the desired position, are shown in Fig. 6.5 and Fig. 6.6, respectively. Fig. 6.5 exhibits that during the period of the entire flight, UAV follows x

direction command within an error of $\pm 0.35\ m$, for 89% of the flight time. The peak error for x direction is approximately 0.55 m in this case. As can be seen from Fig. 6.6, y direction command is within an error of $\pm 0.8\ m$, for 85% of the flight time. The peak error for y direction is approximately 1.3 m. The sum of errors in x and y directions are illustrated in Fig. 6.7. The deviations from x and y directions are 26.8433 m and 67.0565 m within the entire duration. In summary, the simulation results exemplify the effectiveness of the developed S&A strategy.

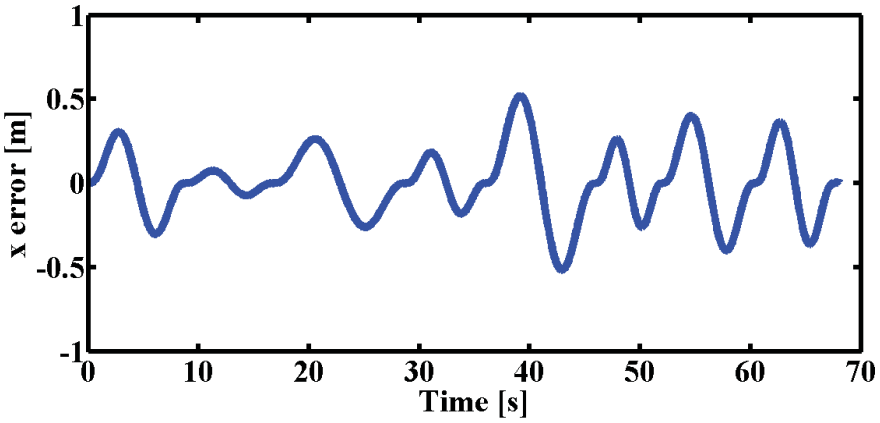


Figure 6.5: x position error.

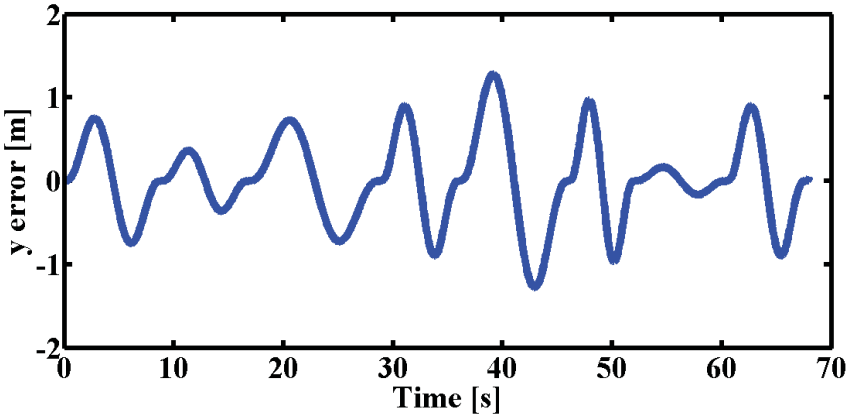


Figure 6.6: y position error.

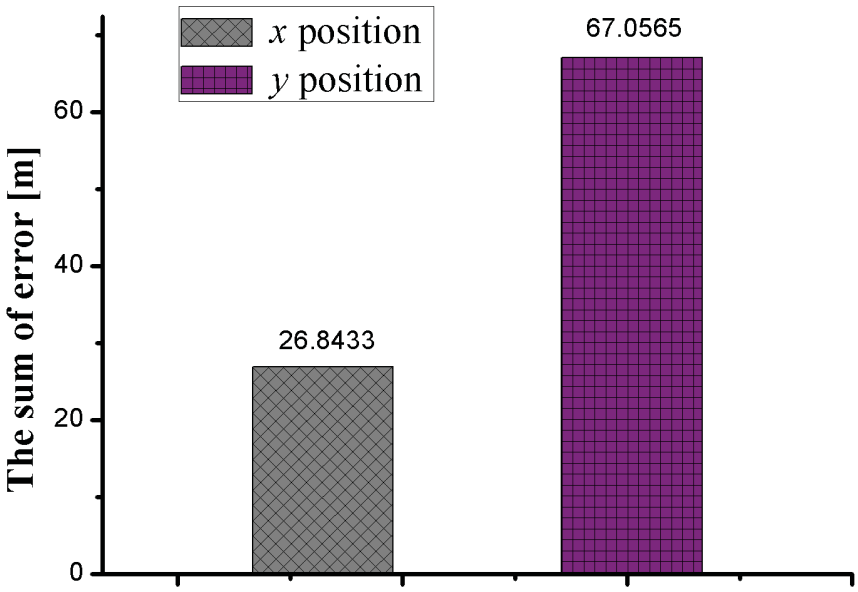


Figure 6.7: The accumulative errors in x and y positions.

For interested readers, the demonstration of sense and collision avoidance strategy using 2D-MDP performed in this work can be found in the Concordias NAV Lab YouTube Channel: https://www.youtube.com/watch?v=m7ar6Ij2dpw.

6.1.5 Summary

This section presents a new development of collision avoidance algorithm that ensures an UAV can avoid multiple intruders autonomously. Firstly, the MDP based approach generates the multiple threats resolution logic for the collision avoidance system. Secondly, the optimal trajectory is smoothed by the differential flatness technique where the constraints of the UAV dynamics are considered. In such a way, the planned trajectory is feasible for the UAV. The effectiveness of the developed scheme is illustrated by the numerical simulation studies.

6.2 MDP Based Collision Avoidance in 3D Plane

6.2.1 Path Planning in 3D

As mentioned in Section 6.1, the 2D collision avoidance algorithm had been developed for free flight scenarios. For the three-dimensional case, the two-dimensional collision-free action is insufficient to capture the additional degree of freedom in the movement of the

UAV. For this purpose, one would need to introduce another actions such as *Up* or *Down* in order to avoid the boundary of 3D intruders in a desired manner.

To generate an available 3D trajectory, the parameters of MDP should be re-defined in this unit.

- **States:** A flight range with 15000 m $\times$15000 $m\times$1000 m is decomposed into a grid array by gridding the flight environment as 100 $m\times$100 $m\times$10 m 3D-grid. The position of initial point (0,0,0) is in the upper left corner of the grid array. The grid is then numbered in a positive direction on the x-axis (horizontally to the right) and in a positive direction on the z-axis (vertically going up).
- **Actions:** As shown in Fig. 6.8, the directions of UAV movement can be categorized into: North, North-East, North-West, Up, Up-East, Up-West, Down, Down-East, Down-West.
- **Transition Functions:** As mentioned at Section 6.1, the state-transition defines the next state distribution based on taking an action at a current state. One scenario of probability transition under the 3D flight environment is presented in Fig. 6.9. Assuming the target waypoint at the north of grid array and state s of a UAV at the center of grid array, if the UAV is moving to the North with 0.9 probability, then the actions to NE and NW with 0.04 probability while to UP and DN with 0.01 probability, respectively. The UAV has nine probability transitions.
- **Reward Functions:** The reward function stands for the value of arriving in state s' after executing action a in state s. The reward function is built and updated at each action, according to two objects: the distance of UAV to intruders and the distance to the goal. The equation of reward function is same with 2D scenario.

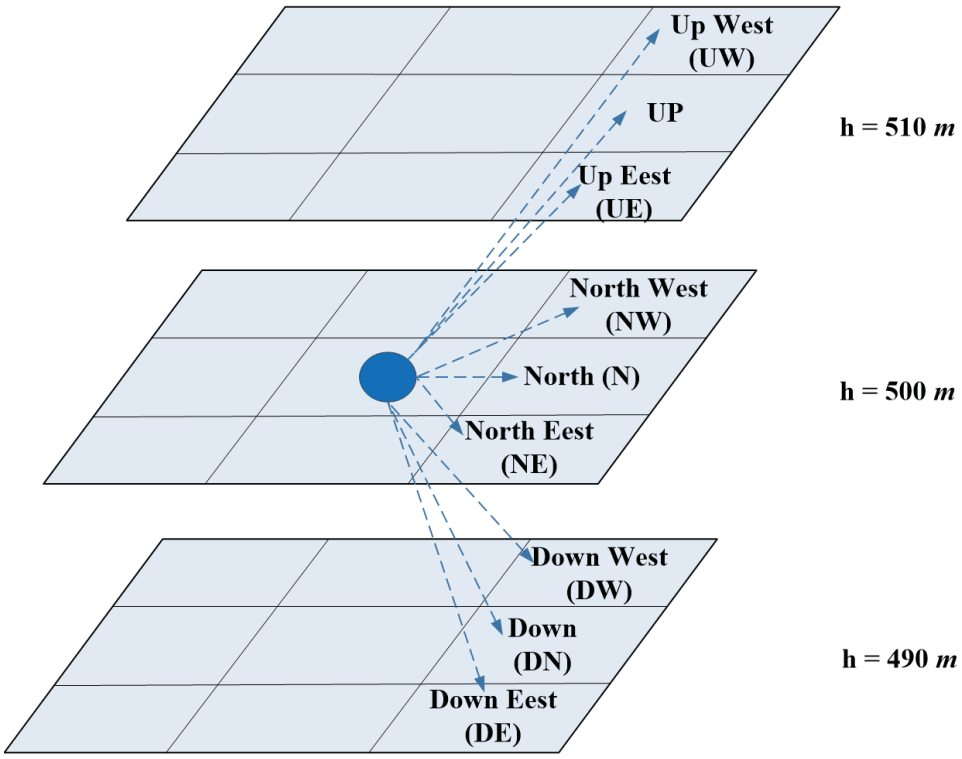


Figure 6.8: The basic action of UAV in 3D plane.

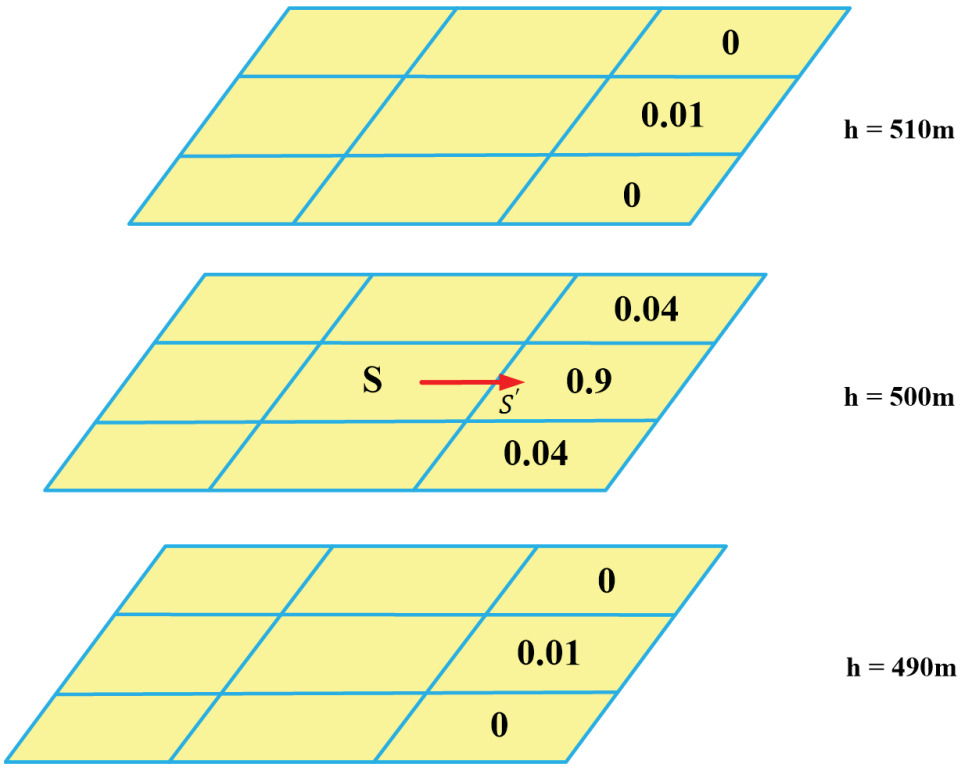


Figure 6.9: The assignment of the transition probability in 3D plane.

Remark 4. Considering the physical constraints of UAV, the optimal trajectory planned by MDP in 3D plane is smoothened by the differential flatness-based approach. Due to the three-dimensional flight scenario developed in this work, the limit of the roll angle and the pitch angle is taken into account. The detailed technology can be followed in Section 5.1.2 and Section 6.1.3.

6.2.2 Path Following Controller Design

Distinctive from the path following controller designed in Section 5.2, the lateral dynamics of the aircraft is explicitly taken into account. From Fig. 6.10, the proposed path following controller using nonlinear energy method aims at ensuring UAV track roll, pitch, and yaw commands of the reference trajectory, where the physical limits of normal actuators are also respected. The proposed controller is constituted by a lateral controller and a longitudinal controller. The lateral controller is applied to provide automatic tracking and stabilization of roll and yaw angle. The longitudinal controller serves to minimize the tracking error of pitch angle and pitch rate, and meet the performance criteria for energy tracking.

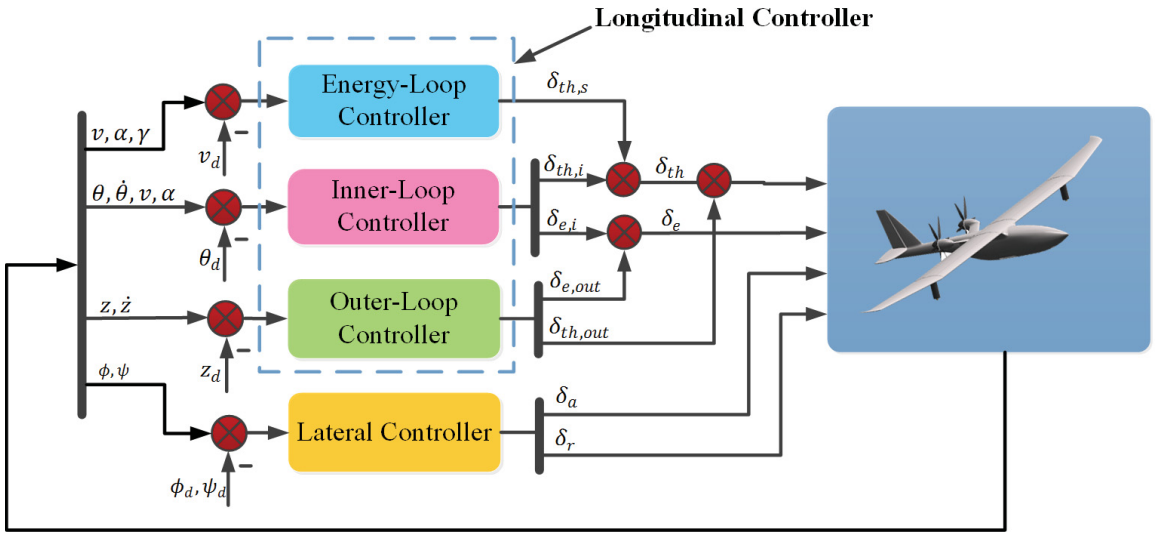


Figure 6.10: The path following controller.

Lateral Controller Design

Based on the lateral dynamics modeled with EL equation, the aileron deflection, δ_a, and the rudder deflection, δ_r, are designed to control the heading and roll angle by modifying the energy function of UAV.

From the expression of the roll angle ϕ, given by Eq. (1) and the dynamics of ψ given by Eq. (2), it is observed that the dynamics between $q_l = [\phi \quad \psi]^T$ and the control input $u_l = [\delta_a \quad \delta_r]^T$ is a system of relative degree two. Denoted q_{ld} as the desired value of q_l, the corresponding error function of q_l is presented as:

$$\lambda_l = R_l \ddot{e}_{q_l} + C_p e_{q_l} + C_d \dot{e}_{q_l}, \tag{79}$$

where $e_{q_l} = q_l - q_{ld}$, $C_p, C_d \in R^+$. The control input u_l selected to enforce that $e_{q_l} \longrightarrow 0$ as $q_l \longrightarrow q_{ld}$ is given by:

$$u_l = M_{L,u}^{-1}[R_l \ddot{q}_l + \frac{\partial F}{\partial \dot{q}_l} - C_p e_{ql} - C_d \dot{e}_{ql}]. \tag{80}$$

For the convergence of tracking error, it is required that the time derivative of energy function is no greater than zero. The desired energy function in the case of lateral dynamics is selected as:

$$H_l = \frac{1}{2}\dot{e}_{ql}^T R_l \dot{e}_{ql} + \frac{1}{2} e_{ql}^T C_p e_{ql}. \tag{81}$$

Substituting Eq. (80) into Eq. (79) yields that:

$$\lambda_l = 0. \tag{82}$$

Taking the time derivative of Eq. (81) and substituting for λ_l from Eq. (82), one obtains:

$$\begin{aligned} \dot{H}_l &= \dot{e}_{ql}(\lambda_l^T - C_d \dot{e}_{ql}^T). \\ &= -\dot{e}_{ql} C_d \dot{e}_{ql}^T \\ &\leq 0. \end{aligned} \tag{83}$$

This indicates that the lateral dynamics is stable with the control law in Eq. (80).

Remark 5. As the lateral controller regulates the aircraft's roll and yaw, a longitudinal controller as introduced in Section 5.2 is designed to maintain the pitch stabilization and follow the desired trajectory. The decoupled design of lateral and longitudinal controller is presented to track the roll, pitch and yaw commands of reference collision-free trajectory.

6.2.3 Simulation Studies

A. Simulation Environment Description

The simulation studies are conducted in the environment where a PC is configured with a 2.0 GHz Intel Core-i7-4510U processor and 8 GB RAM. The UAV dynamics can be found in Chapter 3, while the physical limits of the UAV states and control surfaces are indicated in Table 6.1. The design parameters in the developed scheme are selected as Table 6.2:

Table 6.1: The operating of the UAV

Parameter	Symbol	Amplitude limit
Pitch angle	θ	[-25°, 30°]
Roll angle	ϕ	[-25°, 25°]
Aileron	δ_α	[-25°, 25°]
Elevator	δ_e	[-25°, 10°]
Rudder	δ_r	[-30°, 30°]
Throttle	δ_T	[-0.5°, 10°]

Table 6.2: The design parameters of the UAV

Parameter	Value
Initial x position	0 m
Initial y position	0 m
Initial z position	920 m
Airspeed	80 *m/s*
Initial pitch angle	-0.6°
Initial yaw angle	0
Initial roll angle	0

B. Simulation Scenario

In the simulation scenario, it is assumed that the UAV starts landing from the altitude of 920 m. The reference signal during the landing stage is given in Fig. 6.11. It consists of following twelve distinct phases:

- **Segment 1:** Level flight at 920 m at the heading angle of 0. Velocity is maintained at 80 m/s.
- **Segment 2:** A coordinated left turn with heading angle of 85° at 920 m to align with the runway.
- **Segment 3:** Decent on the pitch rate -0.125°$/s$ to the pitch angle -5°.
- **Segment 4:** Maintain at the pitch angle -5°.
- **Segment 5:** Ascent on the pitch rate 0.125°$/s$ to the pitch angle 5°.
- **Segment 6:** Climb at the pitch angle 5°.
- **Segment 7:** Decent on the pitch rate -0.125°$/s$ to the pitch angle 0.
- **Segment 8:** Level flight at 870 m.
- **Segment 9:** Decent on the pitch rate -0.15°$/s$ to the pitch angle -6°.
- **Segment 10:** Maintain at the pitch angle -6°.
- **Segment 11:** Decent on the pitch rate 0.1°$/s$ to the pitch angle -3°.
- **Segment 12:** Approach at the pitch angle of -3°.

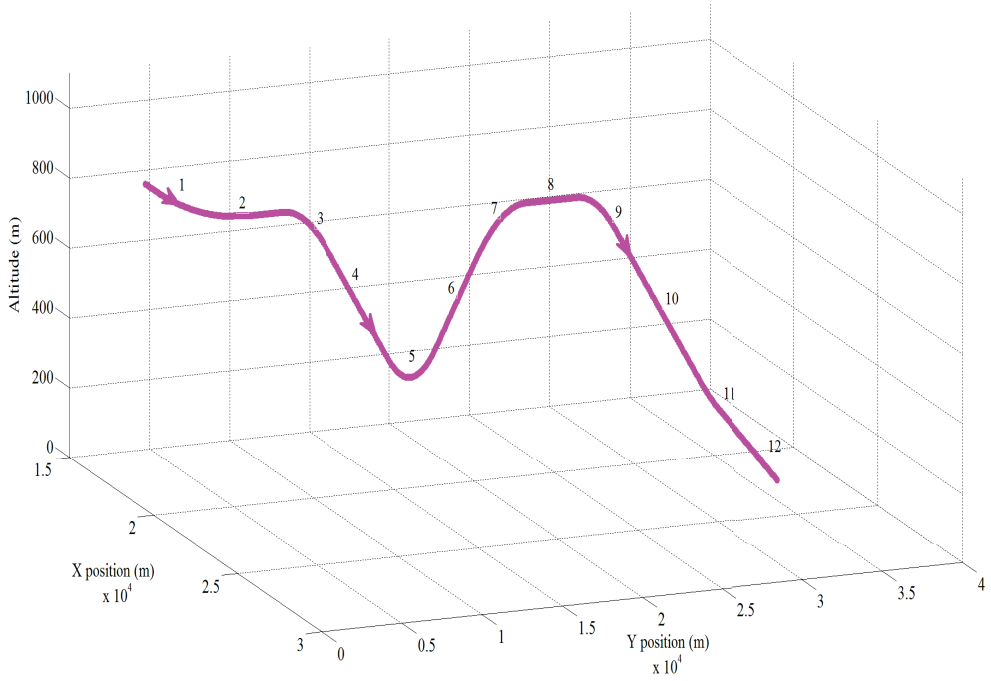


Figure 6.11: The reference signal during the landing stage.

To evaluate the effectiveness of the proposed algorithm, three intruders move toward the UAV from different directions. To be more specific, 1) the first intruder flies toward the UAV from the right hand side; 2) the second threat appears in front of the UAV while the intruder velocity is in the reverse direction of the UAV; and 3) the third one flies to the UAV from the left hand side. In order to quantitatively exam the performance over the entire landing phase, the tracking error metrics are defined as:

$$e_{tra} = \sqrt{\sum_{t_1}^{t_2}[(\theta-\theta_d)^2+(\phi-\phi_d)^2+(\psi-\psi_d)^2]/(t_2-t_1)}, \tag{84}$$

where [t_1,t_2] covers the simulation duration.

C. Simulation Results and Analysis

The 3D trajectory of the UAV is illustrated in Fig. 6.12. It is clear that the UAV is faced with two intruders from lateral and longitudinal directions at the cruising flight stage. The third intruder appears when the UAV is descending. The UAV is capable of completing the

landing maneuvers by resorting the developed S&A methodology. The position information with respect to x-axis, y-axis, and z-axis is exhibited individually in Fig. 6.13. Key observations of Fig. 6.13 include the following: 1) when the first intruder comes from the lateral direction, the proposed S&A algorithm can promptly adjust the lateral trajectory (y-axis position) such that the safety distance between the UAV and the first threat is maintained; 2) when the second intruder is flying along the direction of the UAV, the altitude reference is changed by the proposed S&A, by which the UAV can avoid the potential collision; 3) after the collision avoidance for the second intruder is completed, the UAV starts descending from the current altitude to the previous one; and 4) the proposed S&A drives the UAV turns right such that the safety bound between the UAV and the third intruder can be ensured even during this descending process. Figure 6.14 reveals the tracking performance during the overall landing phase. The designed energy based controller guarantees that the studied UAV can track the reference signals from both transient and steady-state aspects. As a result, the UAV can follow the regenerated trajectory in the presence intruders. The responses of the UAV actuators are highlighted in Fig. 6.15. It is worth noting that the actuators of the UAV work within the allowable limits. This in turn confirms that the planned trajectory by the developed S&A algorithm is feasible, as the physical constraints of the UAV have been already considered in the S&A. Moreover, as can be seen in Fig. 6.16, the defined metrics corresponding to the pitch angle, roll angle, and yaw angle are 17.71°, 32.41°, and 6.29°, respectively. As a results, Figs. (6.12)-(6.16) have demonstrated that the designed S&A scheme is capable of 1) guaranteeing the safety bound between the UAV and the intruder; and 2) driving the UAV to complete the avoidance maneuvers with satisfied performance.

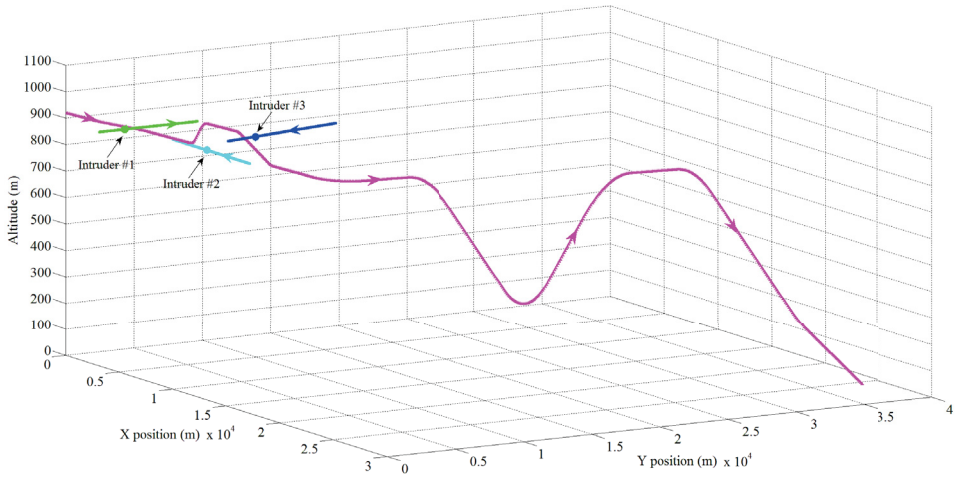


Figure 6.12: The 3D trajectory of the UAV landing.

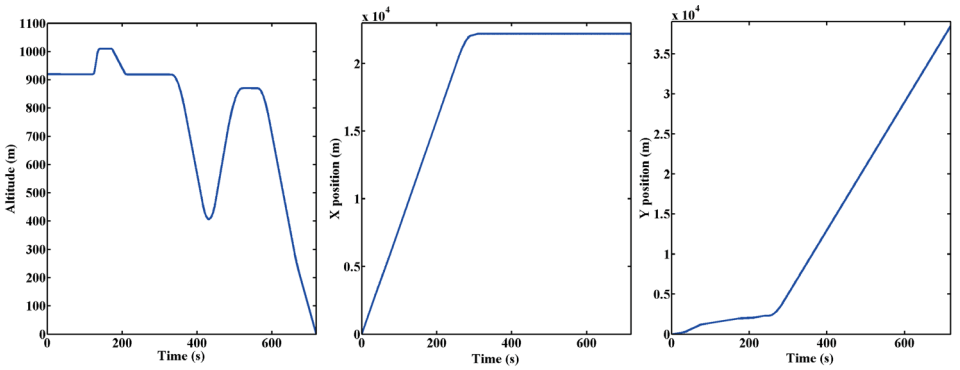


Figure 6.13: The UAV position information with respect to three directions.

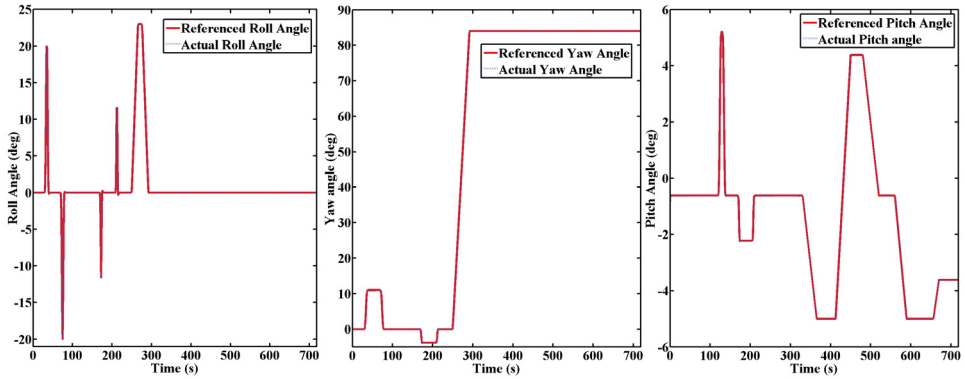


Figure 6.14: The tracking performance of Euler angles.

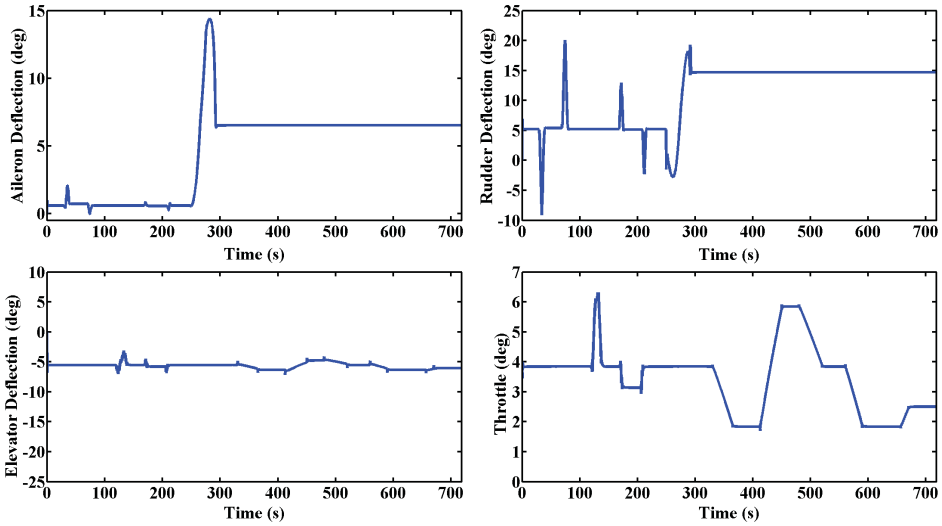


Figure 6.15: The responses of the UAV actuators.

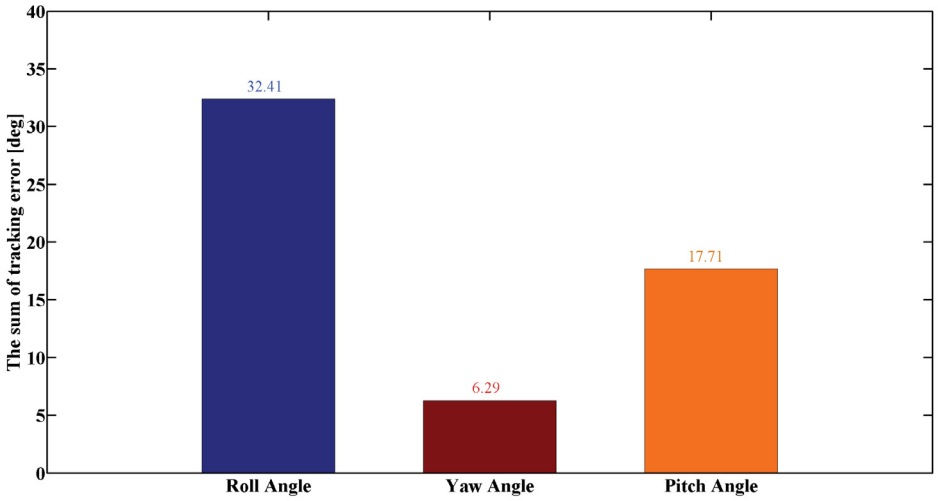


Figure 6.16: The tracking errors.

Interested readers are referred to the following Google Shared Link for relevant simulation videos: https://drive.google.com/file/d/0B3iMpBBeeEYPT1VXM3hNWUNoUGc/view?usp=sharing

6.2.4 Summary

This section presents a 3D S&A approach for UAVs in the landing phase. Firstly, the 3D flight space is modeled as gridded environment in the presence of multiple intruders.

Secondly, the multiple threats resolution logic is generated by the MDP based approach such that the safe separation can be maintained. Thirdly, considering the UAV dynamic constraints, the optimal trajectory is smoothened by the differential flatness technique to make the planned trajectory feasible. Finally, the decoupled energy-based controller is proposed, by which the UAV can be driven to follow the re-planned trajectory and the intruders can be avoided. The numerical simulations have illustrated the effectiveness of the proposed SAA strategy.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

This thesis focus on collision avoidance methods for UAV to ensure safe flight under a subset of possible collision scenarios. The contributions in this thesis can be summarized as:

- The conflict can be predicted by the CPA while DGG is deployed to avoid one intruder.
- The optimal trajectory generated by BBO is available to avoid multiple intruders in the landing phase.
- Modeling the collision avoidance problem as MDP framework, two trajectories are generated to avoid multiple intruders in 2D and 3D plane.
- The feasible trajectory smoothed by flatness method is under the physical limit of UAV.
- Energy based controller is capable of following the generated trajectory.

7.2 Future Work

The following work are recommended for further research into the S&A system.

- The probability of conflict derived in Chapter 3 only considered the sensor measurement noises. The other factors including wind factor, flight technical errors, and time delay should be considered.
- In this thesis, feasible trajectories are generated considering the dynamic constraints of UAV. For the further research, different types of constraints, such as sector bound

aries, weather, terrain, should be investigated.

- Although the numerical simulation results demonstrated the effectiveness of the proposed algorithms in this thesis, large-scaled experiments for different navigation environment and complicated intruders simulations should be conducted in a real UAV testbed to evaluate the approach effectively.

7.3 My Publications Related to the Thesis

[1] Fu, Y., Soupin, P., Yu, X., Zhang, Y., & Cole, P. (2015). Sense and collision avoidance of unmanned aerial vehicles using geometric guidance and flatness approaches. In *Proceedings of 2015 International Conference on Unmanned Aircraft Systems (ICUAS), Denver, USA*, (pp. 601-606).

[2] Fu, Y., Yu, X., & Zhang, Y. (2015). Sense and collision avoidance of unmanned aerial vehicles using markov decision process and flatness approach. In *Proceedings 2015 IEEE International Conference on Information and Automation (ICIA), Lijiang China*, (pp. 714-719) (Received the **Best Paper in Automation Award**).

[3] Fu, Y., Yu, X., & Zhang, Y. An advanced sense and collision avoidance strategy for unmanned aerial vehicles in landing phase. Revision submitted to *IEEE Aerospace and Electronic Systems Magazine*.

References

Akmeliawati, R., & Mareels, I. M. (2010). Nonlinear energy-based control method for aircraft automatic landing systems. *IEEE Transactions on Control Systems Technology*, *18*(4), 871–884.

Billingsley, T. B., Kochenderfer, M. J., & Chryssanthacopoulos, J. P. (2012). Collision avoidance for general aviation. *IEEE Aerospace and Electronic Systems Magazine*, *27*(7), 4–12.

Borrelli, F., Subramanian, D., Raghunathan, A. U., & Biegler, L. T. (2006). MILP and NLP techniques for centralized trajectory planning of multiple unmanned air vehicles. In *Proceedings of American Control Conference, Minneapolis, USA* (pp. 5763–5768).

Breen, B. C. (1999). Controlled flight into terrain and the enhanced ground proximity warning system. *IEEE Aerospace and Electronic Systems Magazine*, *14*(1), 19–24.

Burgess, D. W., Altman, S. I., & Wood, M. L. (1994). TCAS: Maneuvering Aircraft in the Horizontal Plane. *Lincoln Laboratory Journal*, *7*(2).

Campbell, S. E., Bragg, M. B., & Neogi, N. A. (2013). Fuel-optimal trajectory generation for persistent contrail mitigation. *Journal of Guidance, Control, and Dynamics*, *36*(6), 1741–1750.

Carbone, C., Ciniglio, U., Corraro, F., & Luongo, S. (2006). A novel 3D geometric algorithm for aircraft autonomous collision avoidance. In *Proceedings of 2006 45th IEEE Conference on Decision and Control, San Diego, USA* (pp. 1580–1585).

Chamseddine, A., Zhang, Y., Rabbath, C. A., Join, C., & Theilliol, D. (2012). Flatness-based trajectory planning/replanning for a quadrotor unmanned aerial vehicle. *IEEE Transactions on Aerospace and Electronic Systems*, *48*(4), 2832–2848.

Chamseddine, A., Zhang, Y., Rabbath, C. A., & Theilliol, D. (2012). Trajectory planning and replanning strategies applied to a quadrotor unmanned aerial vehicle. *Journal of Guidance, Control, and Dynamics*, *35*(5), 1667–1671.

Chuang, J.-H., & Ahuja, N. (1998). An analytically tractable potential field model of free space and its application in obstacle avoidance. *IEEE Transactions on Systems, Man,*

and Cybernetics, Part B: Cybernetics, *28*(5), 729–736.

Dai, R., & Cochran, J. (2010). Path planning and state estimation for unmanned aerial vehicles in hostile environments. *Journal of Guidance, Control, and Dynamics*, *33*(2), 595–601.

De Filippis, L. (2012). *Advanced path planning and collision avoidance algorithms for uavs* (Unpublished doctoral dissertation). Politecnico di Torino.

Eele, A. J., & Richards, A. (2009). Path-planning with avoidance using nonlinear branch-and-bound optimization. *Journal of Guidance, Control, and Dynamics*, *32*(2), 384–394.

Ergezer, H., & Leblebicioglu, K. (2013). Path planning for UAVs for maximum information collection. *IEEE Transactions on Aerospace and Electronic Systems*, *49*(1), 502–520.

Finn, A., & Franklin, S. (2011). Acoustic sense & avoid for UAV's. In *Proceedings of 2011 Seventh International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), Adelaide, Australia* (pp. 586–589).

Frazzoli, E., Dahleh, M., Feron, E., et al. (2001). Real-time motion planning for agile autonomous vehicles. In *Proceedings of the 2001 American Control Conference, Arlington, USA* (Vol. 1, pp. 43–49).

Fu, Y., Ding, M., & Zhou, C. (2012). Phase angle-encoded and quantum-behaved particle swarm optimization applied to three-dimensional route planning for UAV. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, *42*(2), 511–526.

Fu, Y., Ding, M., Zhou, C., & Hu, H. (2013). Route planning for unmanned aerial vehicle (uav) on the sea using hybrid differential evolution and quantum-behaved particle swarm optimization. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *43*(6), 1451–1465.

Geiger, B. R., & Horn, J. F. (2012). Neural network-based trajectory optimization for unmanned aerial vehicles. *Journal of Guidance, Control, and Dynamics*, *35*(2), 548–562.

Geiger, B. R., Horn, J. F., Sinsley, G. L., Ross, J. A., Long, L. N., & Niessner, A. F. (2008). Flight testing a real-time direct collocation path planner. *Journal of Guidance, Control, and Dynamics*, *31*(6), 1575–1586.

Geyer, C., Dey, D., & Singh, S. (2009). Prototype sense-and-avoid system for UAVs. *Robotics Institute, Carnegie Mellon University, Tech. Rep. CMU-RI-TR-09-09*.

Goerzen, C., Kong, Z., & Mettler, B. (2010). A survey of motion planning algorithms from

the perspective of autonomous uav guidance. *Journal of Intelligent and Robotic Systems*, *57*(1-4), 65–100.

Gu, T., Snider, J., Dolan, J. M., & Lee, J.-w. (2013). Focused trajectory planning for autonomous on-road driving. *2013 IEEE Intelligent Vehicles Symposium Proceedings*, 547–552.

Hottman, S. B., Hansen, K., & Berry, M. (2009). Literature review on detect, sense, and avoid technology for unmanned aircraft systems. *Tech. Report DOT/FAA/AR-08/41, U.S. Department of Transportation, Federal Aviation Administration (FAA), USA*.

Israelsen, J., Beall, M., Bareiss, D., Stuart, D., Keeney, E., & Berg, J. (2014). Automatic collision avoidance for manually tele-operated unmanned aerial vehicles. In *Proceedings of 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong* (pp. 6638–6643).

Jain, S., & Tsiotras, P. (2008). Trajectory optimization using multiresolution techniques. *Journal of Guidance, Control, and Dynamics*, *31*(5), 1424–1436.

Jorris, T. R., & Cobb, R. G. (2008). Multiple method 2-D trajectory optimization satisfying waypoints and no-fly zone constraints. *Journal of Guidance, Control, and Dynamics*, *31*(3), 543–553.

Jung, D., & Tsiotras, P. (2013). On-line path generation for unmanned aerial vehicles using B-spline path templates. *Journal of Guidance, Control, and Dynamics*, *36*(6), 1642–1653.

Karaboga, D., & Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (abc) algorithm. *Journal of Global Optimization*, *39*(3), 459–471.

Karelahti, J., Virtanen, K., & Öström, J. (2008). Automated generation of realistic near-optimal aircraft trajectories. *Journal of Guidance, Control, and Dynamics*, *31*(3), 674–688.

Karimi, J., & Pourtakdoust, S. H. (2013). Optimal maneuver-based motion planning over terrain and threats using a dynamic hybrid pso algorithm. *Aerospace Science and Technology*, *26*(1), 60–71.

Ketema, Y., & Zhao, Y. J. (2010). Controllability and reachability for micro-aerial-vehicle trajectory planning in winds. *Journal of Guidance, Control, and Dynamics*, *33*(3), 1020–1024.

Kim, J., Lee, D., Cho, K., Jo, S., Kim, J., Min, C., & Cho, S. (2010). Development of an electro-optical system for small UAV. *Aerospace Science and Technology*, *14*(7), 505–511.

Kim, Y., Gu, D.-W., & Postlethwaite, I. (2008). Real-time path planning with limited information for autonomous unmanned air vehicles. *Automatica*, *44*(3), 696–712.

Kothari, M., & Postlethwaite, I. (2013). A probabilistically robust path planning algorithm for UAVs using rapidly-exploring random trees. *Journal of Intelligent & Robotic Systems*, *71*(2), 231–253.

Krozel, J., Peters, M. E., & Hunter, G. (1997). Conflict detection and resolution for future air transportation management. *NASA CR-97-205 944*.

Kuchar, J. K., & Yang, L. C. (2000). A review of conflict detection and resolution modeling methods. *IEEE Transactions on Intelligent Transportation Systems*, *1*(4), 179–189.

Lentilhac, S. (2010). UAV flight plan optimzed for sensor requirements. *IEEE Aerospace and Electronic Systems Magazine*, *25*(2), 11–14.

Li, B., Gong, L.-g., & Yang, W.-l. (2014). An improved artificial bee colony algorithm based on balance-evolution strategy for unmanned combat aerial vehicle path planning. *The Scientific World Journal*, *2014*, 1–10.

Lin, & Goodrich, M. (2014). Hierarchical heuristic search using a Gaussian Mixture Model for UAV coverage planning. *IEEE Transactions on Cybernetics*, *44*(12), 2532–2544.

Lin, Hyyppä, J., & Jaakkola, A. (2011). Mini-UAV-borne LIDAR for fine-scale mapping. *IEEE Geoscience and Remote Sensing Letters*, *8*(3), 426–430.

Magni, J.-F., Bennani, S., & Terlow, J. (1997). Robust flight control (a design challenge). *Lecture notes in control and information sciences*.

Marsh, A. K. (2015). *Avoiding midair collisions.* http://flighttraining.aopa.org/students/presolo/skills/midair.html.

Marshall, D., Barnhart, R. K., Shappee, E., & Most, M. T. (2015). *Introduction to unmanned aircraft systems*. Taylor & Francis Group, Boca Raton, FL, USA.

Masoud, A., et al. (2012). Motion planning with gamma-harmonic potential fields. *IEEE Transactions on Aerospace and Electronic Systems*, *48*(4), 2786–2801.

McCalmont, J., Utt, J., Deschenes, M., & Taylor, M. (2005). Sense and avoid, phase i (man-inthe-loop) advanced technology demonstration. *In Proceedings of AIAA Infotech@ Aerospace, Arlington, VA, USA*.

McGee, T. G., & Hedrick, J. K. (2007). Optimal path planning with a kinematic airplane model. *Journal of Guidance, Control, and Dynamics*, *30*(2), 629–633.

Milkie, T. (2007). Passive Acoustic Non-Cooperative Collision Alert System (PANCAS) for UAV Sense and Avoid. *Unpublished white paper by SARA, Inc*.

Minwalla, C., Thomas, P., Ellis, K., Hornsey, R., & Jennings, S. (2012). Flight test evaluation of a prototype optical instrument for airborne sense-and-avoid applications. In

Proceedings of SPIE, Unmanned Systems Technology XIV, Baltimore, ML, USA (pp. 83870R1–14).

Moses, A., Rutherford, M. J., & Valavanis, K. P. (2011). Radar-based detection and identification for miniature air vehicles. In *Proceedings of 2011 IEEE International Conference on Control Applications (CCA), Denver, USA* (pp. 933–940).

Nikolos, I. K., & Tsourvelouds, N. C. (2009). Path planning for cooperating unmanned vehicles over 3-D terrain. In *Informatics in Control, Automation and Robotics* (pp. 153–168).

Obermeyer, K. J., Oberlin, P., & Darbha, S. (2012). Sampling-based path planning for a visual reconnaissance unmanned air vehicle. *Journal of Guidance, Control, and Dynamics*, *35*(2), 619–631.

Ogren, P., & Winstrand, M. (2008). Minimizing mission risk in fuel-constrained unmanned aerial vehicle path planning. *Journal of Guidance, Control, and Dynamics*, *31*(5), 1497–1500.

Ortega, R., Van der Schaft, A., Mareels, I., & Maschke, B. (2001). Putting energy back in control. *IEEE Control Systems*, *21*(2), 18-33.

Osborne III, R., Bar-Shalom, Y., Willett, P., & Baker, G. (2011). Design of an adaptive passive collision warning system for UAVs. *IEEE Transactions on Aerospace and Electronic Systems*, *47*(3), 2169–2189.

Owen, M. P., Duffy, S. M., & Edwards, M. W. (2014). Unmanned aircraft sense and avoid radar: Surrogate flight testing performance evaluation. In *Proceedings of the IEEE Radar Conference, Cincinnati, OH, USA* (pp. 0548–0551).

Park, J.-W., Oh, H.-D., & Tahk, M.-J. (2008). UAV collision avoidance based on geometric approach. In *Proceedings of SICE Annual Conference, Tokyo, Japan* (pp. 2122–2126).

Patel, R. B., & Goulart, P. J. (2011). Trajectory generation for aircraft avoidance maneuvers using online optimization. *Journal of Guidance, Control, and Dynamics*, *34*(1), 218–230.

Paull, L., Thibault, C., Nagaty, A., Seto, M., & Li, H. (2014). Sensor-driven area coverage for an autonomous fixed-wing unmanned aerial vehicle. *IEEE Transactions on Cybernetics*, *44*(9), 1605–1618.

Pehlivanoglu, Y. V. (2012). A new vibrational genetic algorithm enhanced with a voronoi diagram for path planning of autonomous UAV. *Aerospace Science and Technology*, *16*(1), 47–55.

Puterman, M. L. (1994). *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, New York, USA.

Radmanesh, M., Kumar, M., Nemati, A., & Sarim, M. (2015). Dynamic optimal UAV trajectory planning in the National Airspace System via mixed integer linear programming. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 1–15.

Raghunathan, A. U., Gopal, V., Subramanian, D., Biegler, L. T., & Samad, T. (2004). Dynamic optimization strategies for three-dimensional conflict resolution of multiple aircraft. *Journal of Guidance, Control, and Dynamics*, *27*(4), 586–594.

Ragi, S., & Chong, E. K. (2013). UAV path planning in a dynamic environment via partially observable markov decision process. *IEEE Transactions on Aerospace and Electronic Systems*, *49*(4), 2397–2412.

Rosen, P., Hensley, S., Wheeler, K., Sadowy, G., Miller, T., Shaffer, et al. (2007). UAVSAR: New NASA airborne SAR system for research. *IEEE Aerospace and Electronic Systems Magazine*, *22*(11), 21–28.

Satish, K. V., Sudesh, K. K., & Shantha Kumar, N. (2014). Detection of runway and obstacles using electro-optical and infrared sensors before landing. *Defence Science Journal*, *64*(1), 67–76.

Schrijver, A. (1998). *Theory of linear and integer programming*. John Wiley & Sons, New York, USA.

Simon, D. (2008). Biogeography-based optimization. *IEEE Transactions on Evolutionary Computation*, *12*(6), 702–713.

Subbarao, K., & Shippey, B. M. (2009). Hybrid genetic algorithm collocation method for trajectory optimization. *Journal of Guidance, Control, and Dynamics*, *32*(4), 1396–1403.

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction* (Vol. 1) (No. 1). MIT press Cambridge, Cambridge, MA.

Techy, L., & Woolsey, C. A. (2009). Minimum-time path planning for unmanned aerial vehicles in steady uniform winds. *Journal of Guidance, Control, and Dynamics*, *32*(6), 1736–1746.

Tisdale, J., Kim, Z., & Hedrick, J. K. (2009). Autonomous UAV path planning and estimation. *IEEE Robotics & Automation Magazine*, *16*(2), 35–42.

Vachtsevanos, G., Tang, L., Drozeski, G., & Gutierrez, L. (2005). From mission planning to flight control of unmanned aerial vehicles: strategies and implementation tools. *Annual Reviews in Control*, *29*(1), 101–115.

Valovage, E. (2007). Enhanced ADS-B research. *IEEE Aerospace and Electronic Systems Magazine*, *22*(5), 35-38.

Wang, Y., Hussein, I., & Erwin, R. S. (2011). Risk-based sensor management for integrated detection and estimation. *Journal of Guidance, Control, and Dynamics*, *34*(6), 1767–1778.

Williams, P. (2009). Hermite-Legendre-Gauss-Lobatto direct transcription in trajectory optimization. *Journal of Guidance, Control, and Dynamics*, *32*(4), 1392–1395.

Winter, G., Periaux, J., Galan, M., & Cuesta, P. (1996). *Genetic algorithms in engineering and computer science*. John Wiley & Sons, New York, USA.

Wollkind, S., Valasek, J., & Ioerger, T. R. (2004). Automated conflict resolution for air traffic management using cooperative multiagent negotiation. In *Proceedings of AIAA Guidance, Navigation, and Control Conference, Providence, Rhode Island* (pp. 1–11).

Xu, C., Duan, H., & Liu, F. (2010). Chaotic artificial bee colony approach to Uninhabited Combat Air Vehicle (UCAV) path planning. *Aerospace Science and Technology*, *14*(8), 535–541.

Yang, H., & Zhao, Y. (2004). Trajectory planning for autonomous aerospace vehicles amid known obstacles and conflicts. *Journal of Guidance, Control, and Dynamics*, *27*(6), 997–1008.

Yokoyama, N. (2012). Path generation algorithm for turbulence avoidance using real-time optimization. *Journal of Guidance, Control, and Dynamics*, *36*(1), 250–262.

Yokoyama, N., & Ochi, Y. (2009). Path planning algorithms for skid-to-turn unmanned aerial vehicles. *Journal of Guidance, Control, and Dynamics*, *32*(5), 1531–1543.

Yu, Beard, R., & Byrne, J. (2010). Vision-based navigation frame mapping and planning for collision avoidance for miniature air vehicles. *Control Engineering Practice*, *18*(7), 824–836.

Yu, Cao, X.-B., & Zhang, J. (2011). A real-time schedule method for aircraft landing scheduling problem based on cellular automation. *Applied Soft Computing*, *11*(4), 3485–3493.

Yu, & Zhang, Y. (2015). Sense and avoid technologies with applications to unmanned aircraft systems: Review and prospects. *Progress in Aerospace Sciences*, *74*, 152–166.

Zhu, W., & Duan, H. (2014). Chaotic predator–prey biogeography-based optimization approach for UCAV path planning. *Aerospace Science and Technology*, *32*(1), 153–161.