Path Planning and Sensor Management for Multisensor Airborne Surveillance

PATH PLANNING AND SENSOR MANAGEMENT FOR MULTISENSOR AIRBORNE SURVEILLANCE

BY

YINGHUI WANG, B.Sc., M.Sc.

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

AND THE SCHOOL OF GRADUATE STUDIES

OF MCMASTER UNIVERSITY

IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

Doctor of Philosophy (2018) McMaster University

(Electrical & Computer Engineering) Hamilton, Ontario, Canada

TITLE:	Path Planning and Sensor Management for Multisensor Airborne Surveillance
AUTHOR:	Yinghui Wang B.Sc., M.Sc., (Electrical and Computer Engineering) Northwestern Polytechnical University, Xi'an, China
SUPERVISOR:	Prof. T. Kirubarajan
NUMBER OF PAGES:	xix, 185

To my family

Abstract

As a result of recent technological advances in modernized sensor sets and sensor platforms, sensor management combined with sensor platform path planning are studied to conduct intelligence, surveillance and reconnaissance (ISR) operations in novel ways. This thesis addresses the path planning and sensor management for aerial vehicles to cover areas of interest (AOIs), scan objects of interest (OOIs) and/or track multiple detected targets in surveillance missions. The problems in this thesis, which include 1) the spatio-temporal coordination of sensor platforms to observe AOIs or OOIs, 2) the optimal sensor geometry and path planning for localization and tracking of targets in a mobile three-dimensional (3D) space, and 3) the scheduling of sensors working in different (i.e., active and passive) modes combined with path planning to track targets in the presence of jammers, emerge from real-world demands and scenarios.

The platform path planning combined with sensor management is formulated as optimization problems with problem-dependent performance evaluation metrics and constraints. Firstly, to cover disjoint AOIs over an extended time horizon using multiple aerial vehicles for persistent surveillance, a joint multi-period coverage path planning and temporal scheduling, which allows revisiting in a single-period path, is formulated as a combinatorial optimization with novel objective functions. Secondly,

to use a group of unmanned aerial vehicles (UAVs) cooperatively carrying out search-and-track (SAT) in a mobile 3D space with a number of targets, a joint path planning and scanning (JPPS) is formulated based on the predictive information gathered from the search space. The optimal 3D sensor geometry for target localization is also analyzed with the objective to minimize the estimation uncertainty under constraints on sensor altitude, sensor-to-sensor and sensor-to-target distances for active or passive sensors. At last, to accurately track targets in the presence of jammers broadcasting wide-band noise by taking advantage of the platform path planning and the jammer's information captured by passive sensors, a joint path planning and active-passive scheduling (JPPAPS) strategy is developed based on the predicted tracking performance at the future time steps in a 3D contested environment. The constraints on platform kinematic, flyable area and sensing capacity are included in these optimization problems.

For these multisensor path planning and decision making, solution techniques based on the genetic algorithm are developed with specific chromosome representations and custom genetic operators using either the non-dominated sorting multi-objective optimization (MOO) architecture or the weighted-sum MOO architecture. Simulation results illustrate the performance and advantage of the proposed strategies and methods in real-world surveillance scenarios.

Acknowledgements

It is a great pleasure to acknowledge my deepest gratitude to my supervisor, Prof. T. Kirubarajan, for accepting me as a Ph.D. student, for supporting me financially and spiritually, for suggesting the topic of this thesis, and for his advice and guidance throughout the course of this research work. It is a great honor to work under his supervision.

I would like to thank Dr. R. Tharmarasa, for his encouragement, creative and comprehensive advice during these four years. I have accumulated lots of knowledge and built up hand-on skills during the time working with him. I would also like to express my sincere gratitude and appreciation to my committee members, Prof. I. Bruce and Prof. A. Jeremic, for their generous advice and suggestions during the study. I appreciate all your time spent on my supervisory meetings and reading this thesis. I would like to thank our sponsors and collaborators from Defence Research and Development Canada (DRDC), Dr. R. Jassemi, Dr. B. Kim and Mr. N. Kashyap, for their endless help and for sharing their knowledge on the real-world problems and solution techniques covered in this thesis. I would like to express my special gratitude and thanks to the visiting professors to our group, Prof. Y. Zuo, Prof. W. Wang and Prof. Y. Guo, for their valuable comments on my research papers.

I thank all my friends in the lab, in particular, Yuanhao, Qingsong, Krishanth,

Erkan, Dan, Keqi, Ben, Tongyu, and Pengfei, for their help and support throughout my Ph.D. degree. I am thankful to the administrative staff of the ECE department, specially Ms. C. Gies, for their administrative support.

Lastly and most importantly, I would like to express my special gratitude and thanks to my father and mother for their unconditional love and consistent support through all the years.

Abbreviations

2D	Two-dimensional
3D	Three-dimensional
AOA	Angle-of-arrival
AOI	Area of interest
CPP	Coverage path planning
CRLB	Cramér Rao lower bound
DOA	Direction of arrival
EA	Evolutionary algorithm
ECM	Electronic countermeasures
EKF	Extended Kalman filter
EM	Electromagnetic
ESM	Electronic support measures
FA	False alarm
FIM	Fisher information matrix
FOV	Field of view
GA	Genetic algorithm
IMM	Interacting multiple model

IPDA	Integrated probability data association
IRF	Information reduction factor
ISR	Intelligence, surveillance and reconnaissance
ISTAR	Intelligence, surveillance, target acquisition, and reconnaissance
JPPAPS	Jointly path planning and active-passive scheduling
JPPS	Joint path planning and scanning
LOS	Line of sight
MILP	Mixed-integer linear programming
MINLP	Mixed-integer nonlinear programming
ML-PDA	Maximum likelihood probabilistic data association
MMSE	Minimum mean squared error
MOEA	Multiobjective evolutionary algorithm
MOO	Multiobjective optimization
MTT	Multiple target tracking
MTZ-SEC	Miller-Tucker-Zemlin subtour elimination constraint
NLP	Nonlinear programming
NSGA	Nondominated sorting genetic algorithm
OOI	Object of interest
OP	Orienteering problem
PCRLB	Posterior Cramér Rao lower bound
PDAF	Probabilistic data association filter
POI	Point of interest
PSO	Particle swarm optimization
RHC	Receding horizon control

RGPO	Range gate pull off
RMSE	Root-mean-square error
SA	Simulated annealing
SAR	Synthetic aperture radar
SAT	Search-and-track
SINR	Signal-to-interference-plus-noise ratio
SOJ	Standoff jammer
TDOA	Time-difference-of-arrival
TOA	Time-of-arrival
TS	Tabu search
TSP	Travelling salesman problem
UAV	Unmanned aerial vehicle
VRP	Vehicle routing problem

Contents

List of Tables

List of Figures

Declaration of Academic Achievement

This research presents analytical and computational work carried out solely by Yinghui Wang, herein referred to as "the author", with advice and guidance provided by the academic supervisor Prof. T. Kirubarajan. Information that is presented from outside sources which has been used towards analysis or discussion, has been cited when appropriate, all other materials are the sole work of the author.

Chapter 1

Introduction

1.1 Sensor Management and Path Planning: A Brief Review

Technological advances in manned or unmanned vehicles and modernized sensor sets have enabled more efficient ways to conduct intelligence, surveillance and reconnaissance (ISR) operations. The update and upgrade of sensor sets and sensing platforms have become a trend for large-scale surveillance, which has led to the study on efficient mission planning and the integration of multiple surveillance tasks using a group of sensing platforms with different sensors.

In the context of target tracking, major issues in sensor management are optimal path planning [8, 20, 21], optimal sensor placement [9, 19] and optimal sensor selection [25, 30]. In the optimal path planning, the objective is to find the optimal and safe path(s) for one moving sensor or a group of mobile sensors, such as aircraft and UAVs, under the given constraints, such as the maximum turn rate, to monitor the

surveillance region [6, 16, 20, 21, 23]. In the optimal sensor placement, the objective is to decide where, when and how many sensors must be placed such that the target state estimation errors remain below a certain threshold level [13]. In the optimal sensor selection, the objective is to decide which of the already deployed sensors must be used at each measurement time step in order to maximize the tracking performance [24, 26].

In this thesis, we consider the optimal sensor placement for localization and tracking in 3D, joint path planning and scanning for search-and-track in a 3D mobile search space, joint path planning and sensor working mode (i.e., active or passive) scheduling for multitarget tracking in the presence of jammers, and coverage path planning and temporal scheduling for large-scale airborne surveillance missions.

The Cramér-Rao lower bound (CRLB), which is the inverse of Fisher information matrix (FIM) and provides a mean square error bound on the performance of any unbiased estimator of an unknown parameter vector, is frequently used in the optimization formulation for target localization and optimal sensor placement [2, 12, 19, 31]. The posterior Cramér-Rao lower bound (PCRLB), which gives a measure of the achievable optimum performance and can be calculated predictively, is often used in the criteria for optimal sensor resource management problems for target tracking, such as optimal sensor selection, sensor placement, power allocation and path planning [8, 13, 25, 30, 32]. Meanwhile, many path planning approaches for SAT missions use the value of gathered information as the objective function [20, 22].

In SAT missions, the goal is not only to find or detect targets, but also to track them for the remainder of the mission. A decentralized cooperative control algorithm, which maximizes the information gathered from an environment and decides the next

area to scan by predicting how much information can be gained by searching that area, is proposed in [22] for multiple cooperative UAVs to track detected targets as well as to search for the undetected ones. An information-based objective function, which integrates the conflicting objectives of target detection, target tracking and vehicle survivability into a single scalar index, is presented for UAV path optimization in [20].

The objective function in optimal sensor placement can be applied to optimal path planning. The analytical solutions for optimal sensor-to-target geometries reveal important insights into path planning for target tracking [19, 31]. The optimal geometries for a two-dimensional (2D) time-of-arrival (TOA) localization configuration is analyzed and the UAV path planning, which is based on minimizing the area of estimation confidence region, is simulated in [19]. Based on the FIM, the UAV path planning for passive emitter localization is presented in [8], where angle-of-arrival, time-difference-of-arrival and scan-based localization are considered. Constraints to the UAV dynamics, flyable region and no-fly zone are considered for the optimal path planning [8].

Considerable effort has been dedicated to UAV path planning for target searching. Pattern templates such as ladder pattern are commonly used in search missions [1], in which the environment is always a bounded region either with or without obstacles or forbidden regions inside. The surveillance region is often divided into cells where each cell is associated with a probability or confidence of target existence, which constitutes a probability map for the region [4].

Although optimal path planning has received considerable attention for detection, localization and tracking, existing algorithms are still limited regarding the

assumptions about the environment, search space and sensing capability. A number of optimal sensor placement and path planning algorithms are developed in 2D space without extending to 3D scenarios [19, 20, 22]. Recently, a comprehensive analysis of optimal sensor-to-target geometries is provided in [31] for 3D AOA target localization with no restriction on the number of sensors. However, it fails to consider that the sensor placement region is always constrained in real applications. Hence, it is worth studying the optimal sensor geometry and path planning under constraints on sensor altitude, sensor-to-sensor distance and sensor-to-target distance in 3D.

Coverage path planning (CPP) has been comprehensively studied for applications such as vacuum cleaning, lawn mowing, robotic demining, and machine milling. The environment is always a bounded region either with or without obstacles inside. Pattern templates such as boustrophedon (also known as the ladder search path or lawnmower pattern) are widely used in these problems [7]. A common approach to coverage planning is to decompose the target region into subareas, select a sequence of those subareas, and then generate a path that covers each subarea in turn [7]. However, very few studies in CPP [7] or path planning using coverage pattern templates [11, 14, 15] have looked at multiple vehicles covering disjoint target regions, which is a common question in surveillance missions. In addition, multiple flights for one aircraft in the mission plan should be considered because each aircraft may take off and land multiple times within the mission time horizon, which is referred to as multi-period path planning [5, 28].

The shortest path or the minimum-cost vehicle route is always the objective in the travelling salesman problem (TSP), vehicle routing problem (VRP) and their variants [3, 18, 27]. In contrast, in many surveillance mission planning scenarios, the mission

duration is given, and the main objective is to maximize the information collected within this fixed time [10, 17]. The orienteering problem (OP, also known as the selective TSP) maximizes the total collected score on a path that visits some of the vertices in limited length [29].

1.2 Theme and Objectives of Dissertation

In compliance with the terms and regulations of McMaster University, this dissertation has been written in sandwich thesis format by assembling three articles. These articles represent the independent research performed by the author of this dissertation, Yinghui Wang.

The articles in the dissertation are focused on path planning and sensor management for target searching, localization and tracking using multiple moving sensors in airborne surveillance missions. The general theme is based on the following:

i) To derive optimal airborne surveillance mission plans, i.e., formulate and solve a path planning and temporal scheduling problem with the evaluation on covered area for target searching using multiple sensor platforms (e.g., patrolling aircraft) in disjoint AOIs over an extended time horizon (Paper I).

ii) To analyse the optimal sensor geometry and optimal path planning for target localization given constraints on sensor altitude, sensor-to-sensor distance and sensor-to-target distance (Paper II).

iii) To propose a joint path planning and scanning strategy for searching and tracking possible targets in a mobile 3D surveillance space by a group of UAVs (Paper II).

iv) To propose a joint path planning and radar active-passive mode scheduling

strategy using a group of UAVs for target tracking in the presence of hostile interference (Paper III).

v) To develop cooperative control schemes and online path planning solution techniques for a group of UAVs (Papers II and III).

1.3 Summary of Enclosed Articles

The papers enclosed in this thesis are listed as follows:

1.3.1 Paper I (Chapter 2)

Yinghui Wang, T. Kirubarajan, R. Tharmarasa, Rahim Jassemi-Zargani, Nathan Kashyap

Multi-period Coverage Path Planning and Scheduling for Airborne Surveillance, Accepted in final form for *IEEE Transactions on Aerospace and Electronic Systems*, Mar. 2018. (doi: 10.1109/TAES.2018.2812538)

Preface: This paper studies a new variant of multi-constrained vehicle routing problems for airborne surveillance missions, which integrates coverage path planning, temporal scheduling and area revisiting. New evaluation metrics for coverage path planning are provided as objective functions, which promptly update information collected from different areas and make full use of aircraft capacity to cover AOIs. The optimization formulation of multi-period path planning and scheduling, which allows revisiting in a single-period path, is first provided. Revisiting in single-period paths and scheduling of take-off time are incorporated in the objectives and constraints, which makes the formulation different from the existing ones in vehicle routing and

path planning problems. A new chromosome structure is given to represent the mission plan, which integrates the path selection and the take-off time decision so that multi-period path planning and temporal scheduling can be effectively handled in the multi-objective evolutionary algorithm (MOEA). The proposed algorithm is intelligent not only because the chromosome structure and the custom genetic operators are used, but also because the final solution can be selected from a set of mission plans provided by the MOO according to the user preferences. The proposed solution framework can be used in a variety of surveillance problems, even though most of the surveillance mission planning problems are application-dependent or environment-dependent.

1.3.2 Paper II (Chapter 3)

Yinghui Wang, R. Tharmarasa, T. Kirubarajan, Bumsoo Kim, Rahim Jassemi-Zargani Multisensor Joint Path Planning and Scanning for Mobile 3D Search-and-Track, Submitted to *IEEE Transactions on Aerospace and Electronic Systems*, June 2018.

Preface: This paper expands on the optimal 3D sensor geometry and path planning for target localization and tracking by cooperative sensors from different angles with restrictions on the sensor altitude, sensor-to-sensor and sensor-to-target distances. Different constraints on sensor placement are given based on real-world applications. The optimization problem is formulated to minimize the trace of the inverse of FIM. Extensive examples of the optimal sensor geometry and path planning for 3D localization are shown in the simulation studies. This paper also addresses the path planning and scan decision making for a group of UAVs cooperatively carrying out SAT in a mobile 3D space with a number of targets. A weighted-sum multiobjective

optimization formulation for mobile 3D surveillance space SAT, which integrates the tracking performance of the search space, detected targets within this space as well as the yet-to-be-detected targets by different sensors, is presented. Note that the search space is mobile and in 3D, which distinguishes our problem and the proposed JPPS strategy from those approaches for 2D SAT within a fixed surveillance region.

1.3.3 Paper III (Chapter 4)

Yinghui Wang, R. Tharmarasa, T. Kirubarajan, Anthony Damini, Martie Goulding Joint Path Planning and Radar Mode Scheduling for Multitarget Tracking in ECM, To be submitted to *IEEE Transactions on Aerospace and Electronic Systems*, August 2018.

Preface: This paper jointly considers path planning and radar active-passive mode scheduling using radar systems on UAVs in a 3D contested environment with hostile interference. The study expands on target tracking in the presence of jammers and taking advantage of the jammers' information to suppress the interference and further enhance the tracking performance. A mixed-integer nonlinear programming in a receding horizon optimization based on the predictive PCRLB is formulated to achieve accurate target state estimation, where the PCRLB values in the future steps for different radar modes are predicted. At each time step, UAV paths and radar working modes are determined to leverage the tracking of targets or jammers considering possible target/jammer fire avoidance, UAV collision avoidance and hostile interference mitigation.

Bibliography

[1] J. F. Araújo, P. B. Sujit, and J. B. Sousa, "Multiple UAV area decomposition and coverage," in *2013 IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA)*, Apr. 2013, pp. 30–37.

[2] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software.* John Wiley & Sons, 2004.

[3] T. Bektas, "The multiple traveling salesman problem: An overview of formulations and solution procedures," *Omega*, vol. 34, no. 3, pp. 209–219, 2006.

[4] L. F. Bertuccelli and J. P. How, "Robust UAV search for environments with imprecise probability maps," in *Proceedings of the 44th IEEE Conference on Decision and Control*, Dec. 2005, pp. 5680–5685.

[5] N. Bostel, P. Dejax, P. Guez, and F. Tricoire, "Multiperiod planning and routing on a rolling horizon for field force optimization logistics," in *The Vehicle Routing Problem: Latest Advances and New Challenges*, B. Golden, S. Raghavan, and E. Wasil, Eds. Boston, MA: Springer US, 2008, pp. 503–525.

[6] H. Chen, K. Chang, and C. S. Agate, "UAV path planning with Tangent-plus-Lyapunov vector field guidance and obstacle avoidance," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 2, pp. 840–856, Apr. 2013.

[7] H. Choset, "Coverage for robotics—A survey of recent results," *Annals of Mathematics and Artificial Intelligence*, vol. 31, no. 1–4, pp. 113–126, 2001.

[8] K. Doğançay, "UAV path planning for passive emitter localization," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 2, pp. 1150–1166, Apr. 2012.

[9] K. Doğançay and H. Hmam, "Optimal angular sensor separation for AOA localization," *Signal Processing*, vol. 88, no. 5, pp. 1248 – 1260, 2008.

[10] H. Ergezer and K. Leblebicioglu, "Path planning for UAVs for maximum information collection," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 1, pp. 502–520, Jan. 2013.

[11] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1258–1276, 2013.

[12] M. L. Hernandez, A. Farina, and B. Ristic, "PCRLB for tracking in cluttered environments: measurement sequence conditioning approach," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 42, no. 2, pp. 680–704, Apr. 2006.

[13] M. L. Hernandez, T. Kirubarajan, and Y. Bar-Shalom, "Multisensor resource deployment using posterior Cramer-Rao bounds," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, no. 2, pp. 399–416, Apr. 2004.

[14] P. J. Jones, "Cooperative area surveillance strategies using multiple unmanned systems," Ph.D. dissertation, School of Elect. & Comp. Eng., Georgia Tech, Atlanta, GA, 2009.

[15] O. Karasakal, "Minisum and maximin aerial surveillance over disjoint rectangles," *TOP*, vol. 24, no. 3, pp. 705–724, 2016.

[16] M. Mallick, V. Krishnamurthy, and B.-N. Vo, *Integrated Tracking, Classification, and Sensor Management: Theory and Applications*. John Wiley & Sons, 2012.

[17] A. Moitra, R. Mattheyses, V. DiDomizio, L. Hoebel, R. Szczerba, and B. Yamrom, "Multivehicle reconnaissance route and sensor planning," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 3, pp. 799–812, July 2003.

[18] J. R. Montoya-Torres, J. L. Franco, S. N. Isaza, H. F. Jiménez, and N. Herazo-Padilla, "A literature review on the vehicle routing problem with multiple depots," *Computers & Industrial Engineering*, vol. 79, pp. 115–129, 2015.

[19] N. H. Nguyen and K. Doğançay, "Optimal geometry analysis for multistatic TOA localization," *IEEE Transactions on Signal Processing*, vol. 64, no. 16, pp. 4180–4193, Aug. 2016.

[20] R. Pitre, X. Li, and R. Delbalzo, "UAV route planning for joint search and track missions—An information-value approach," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 3, pp. 2551–2565, July 2012.

[21] S. Ragi and E. K. P. Chong, "UAV path planning in a dynamic environment via

partially observable Markov decision process," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 4, pp. 2397–2412, Oct. 2013.

[22] A. Sinha, T. Kirubarajan, and Y. Bar-Shalom, "Autonomous surveillance by multiple cooperative UAVs," *Proc. SPIE*, vol. 5913, pp. 59 131V–59 131V–12, 2005.

[23] ——, "Autonomous search, tracking and classification by multiple cooperative UAVs," *Proc. SPIE*, vol. 6235, pp. 623 508–623 508–12, 2006.

[24] R. Tharmarasa, T. Kirubarajan, and M. Hernandez, "Large-scale optimal sensor array management for multitarget tracking," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 5, pp. 803–814, Sept. 2007.

[25] R. Tharmarasa, T. Kirubarajan, M. L. Hernandez, and A. Sinha, "PCRLB-based multisensor array management for multitarget tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 2, pp. 539–555, Apr. 2007.

[26] R. Tharmarasa, T. Kirubarajan, A. Sinha, and T. Lang, "Decentralized Sensor Selection for Large-Scale Multisensor-Multitarget Tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 2, pp. 1307–1324, Apr. 2011.

[27] P. Toth and D. Vigo, Eds., *The Vehicle Routing Problem*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2001.

[28] F. Tricoire, M. Romauch, K. F. Doerner, and R. F. Hartl, "Heuristics for the multi-period orienteering problem with multiple time windows," *Computers & Operations Research*, vol. 37, no. 2, pp. 351–367, 2010.

[29] P. Vansteenwegen, W. Souffriau, and D. V. Oudheusden, "The orienteering problem: A survey," *European Journal of Operational Research*, vol. 209, no. 1, pp. 1–10, 2011.

[30] M. Xie, W. Yi, T. Kirubarajan, and L. Kong, "Joint node selection and power allocation strategy for multitarget tracking in decentralized radar networks," *IEEE Transactions on Signal Processing*, vol. 66, no. 3, pp. 729–743, Feb. 2018.

[31] S. Xu and K. Doğançay, "Optimal sensor placement for 3-D angle-of-arrival target localization," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 3, pp. 1196–1211, June 2017.

[32] J. Yan, H. Liu, W. Pu, S. Zhou, Z. Liu, and Z. Bao, "Joint beam selection and power allocation for multiple target tracking in netted colocated MIMO radar system," *IEEE Transactions on Signal Processing*, vol. 64, no. 24, pp. 6417–6427, Dec. 2016.

The following chapter is a reproduction of an Institute of Electrical and Electronics Engineers (IEEE) copyrighted, published paper:

Yinghui Wang, T. Kirubarajan, R. Tharmarasa, Rahim Jassemi-Zargani, Nathan Kashyap, Multi-period Coverage Path Planning and Scheduling for Airborne Surveillance, Accepted in final form for *IEEE Transactions on Aerospace and Electronic Systems*, Mar. 2018. (doi: 10.1109/TAES.2018.2812538)

Chapter 2

Multi-period Coverage Path Planning and Scheduling for Airborne Surveillance

2.1 Abstract

In this paper, optimal surveillance mission plans are developed to cover disjoint areas of interest (AOIs) over an extended time horizon using multiple aerial vehicles. AOIs to be covered are divided into a number of cells. To promptly update information collected from AOIs and to ensure persistent surveillance, each cell is to be revisited within a time slot. Joint path planning and temporal scheduling is formulated as a combinatorial optimization with the proposal of novel objective functions: 1) maximizing the minimum number of non-repeatedly covered cells in a sliding-window fashion and 2) maximizing the total number of covered cells in the mission plan. A

multi-objective evolutionary algorithm (MOEA) with a specific chromosome representation and custom genetic operators, in which the constraint that each cell be revisited within a time slot is transformed into the third objective to handle infeasibility, is developed. The initial single-period paths are generated by solving a series of orienteering problems. The initial population is obtained by connecting these single-period paths and selecting the take-off time for each flight. Three mutation moves are proposed to enable revisiting in a single-period path and rescheduling of take-off time. The solutions converge in the MOEA and are selected by a weighted-sum model according to user preferences in decision making. Simulation results on different mission scenarios and different criteria show the superiority of the proposed algorithm. The algorithm is done offline ahead of the missions and requires modest computational resources.

2.2 Introduction

In order to have the capabilities to observe, analyze, record and report activities in disjoint areas of interest (AOIs), it is often necessary to upgrade expensive airborne surveillance systems to maximize effectiveness [19, 23]. Technological advances in manned or unmanned vehicles and modernized sensor sets have enabled patrolling frequency to be increased and surveillance areas expanded. Efficient mission plans have become critical for intelligence, surveillance and reconnaissance (ISR) operations [23].

The purpose of this study is to derive a mission plan for multiple aircraft over an extended time horizon (e.g., 48 hours) in order to maximize the information collected over a number of disjoint AOIs (as shown in Fig. 2.1). It is required that regions inside

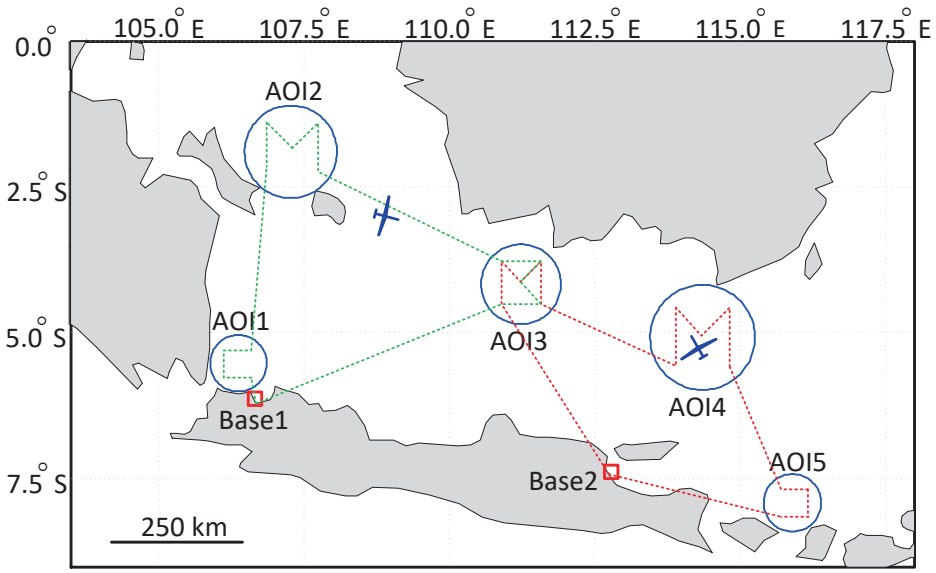


Figure 2.1: Mission planning scenario: two aircraft conduct intelligence-gathering over five AOIs.

AOIs be revisited within a given time slot (e.g., 8-hour slot). The spatio-temporal control of aircraft for a given planning horizon is determined to provide prompt and useful information as well as to achieve persistent and effective surveillance.

Very few studies in coverage path planning (CPP) [4] or path planning using coverage pattern templates [12, 18, 19] have looked at multiple vehicles covering disjoint target regions, which is a common question in surveillance missions. Similar to [19], our problem includes disjoint AOIs, path planning inside AOIs, and aircraft take off from and land at the same base. However, in our study, an AOI can be of any shape and the path inside an AOI does not necessarily follow the strips as in [19]. Instead of using coverage pattern templates, paths for one flight are generated by a heuristic approach. In addition, multiple flights for one aircraft in the mission plan are considered because each aircraft may take off and land multiple times within the mission time horizon, which is referred to as multi-period path planning [2, 32].

The shortest path or the minimum-cost vehicle route is always the objective in the travelling salesman problem (TSP), vehicle routing problem (VRP) and their variants [1, 24, 31]. In contrast, in many surveillance mission planning scenarios, the mission duration is given, and the main objective is to maximize the information collected within this fixed time [11, 23]. In this study, we concentrate on information maximization given the mission time. The orienteering problem (OP, also known as the selective TSP) maximizes the total collected score on a path that visits some of the vertices in limited length [34]. However, our problem cannot be solved by the existing approaches in OPs or turning the cost minimization to a reward maximization in VRPs. The main reasons are as follows:

1) Due to the nature of dynamic surveillance environment, new threats may emerge at any time and the objects of suspicion may evade, which requires periodic or aperiodic revisits of AOIs to update information [18]. Aircraft should make full use of their capacity to survey and not be satisfied with visiting AOIs only once. Therefore, revisit of an area by the same or different aircraft should be allowed. However, every vertex is visited at most once and the score of each vertex can be collected by one vehicle at most in OP [32]. In multiple travelling salesman problems (mTSPs) or VRPs, every vertex is visited exactly once [1, 24].

2) In the proposed problem, collecting information requires searching inside AOIs by covering different areas (as shown in Fig. 2.1), which not only takes time but also requires path planning inside AOIs. Even though service time is needed when the vehicle visits a customer in some variants of VRP [10], path planning is not conducted within this service time.

3) The proposed problem cannot be solved by simultaneously maximizing travel

time (the cost of paths) inside AOIs and minimizing travel time outside AOIs, because each region inside AOIs is required to be revisited within a given time slot. The topology of mission scenario and the mission criteria lead to new objectives and constraints (Sections 2.3 and 2.4), which further generate new solution techniques (Section 2.5).

Note that the information collected in this paper is evaluated by the coverage of AOIs [11, 23] rather than by detecting and tracking targets [25]. Paths are planned offline even if there is no target in the AOIs. Online search and tracking is the topic of a separate publication.

A path planning method for a single unmanned aerial vehicle (UAV) is proposed to maximize the information collected (based on images captured by a camera at the bottom of the UAV) from desired regions while avoiding flying over forbidden regions in [11]. A multivehicle team planning approach is described to maximize the target coverage (number of cells scanned in the region of interest) for surveillance missions [23]. In our paper, the sensing model is similar to what is described in [11, 16, 23]. AOIs to be covered are divided into a number of cells. However, we use two objectives to evaluate the information collected: 1) the number of non-repeatedly covered cells in a sliding-window fashion [5] and 2) the total number of covered cells in the mission plan. Maximizing these objectives not only makes full use of aircraft capacity to cover AOIs but also conquers the problem described in [11] that, once an aircraft enters an AOI, it remains inside instead of flying to other AOIs. Moreover, the surveillance scenarios in our study are more realistic and complicated than those in [11, 23].

In this paper, a sequence of cells is selected to generate a path for one flight (namely, a single-period path) that covers these cells in turn. Since aircraft take

off and land multiple times in the mission, single-period paths are connected head to tail with a given minimum downtime (corresponding to aircraft maintenance and refueling time) and an adjustable wait time between two flights, by which paths for multiple flights (namely, multi-period paths) are generated. The mission plan is a collection of multi-period paths for all aircraft, in which paths are planned and the take-off time for each flight is determined.

We formulate the proposed problem as a combinatorial optimization. A multi-objective evolutionary algorithm (MOEA) with a specific chromosome representation and custom genetic operators, which not only selects the paths but also determines the take-off time schedule and enables revisiting in each flight, is developed. The constraint that each cell be revisited within a time slot is transformed into an objective in the evolutionary algorithm (EA).

We believe that our path planning and scheduling yields many original contributions.

1) The proposed objective functions provide new evaluation metrics for coverage path planning in surveillance missions, which promptly update information collected from different areas and make full use of aircraft capacity to cover AOIs.

2) The optimization formulation of multi-period path planning and scheduling, which allows revisiting in a single-period path, is first provided. Path selection and take-off time determination are jointly considered. Revisit of cells in single-period paths and scheduling of take-off time are incorporated in the objectives and constraints, which makes the formulation different from the existing ones in VRPs, OPs and CPPs [12, 24, 34].

3) A new chromosome structure is given to represent the mission plan, which

integrates the path selection and the take-off time decision so that multi-period path planning and temporal scheduling can be effectively handled in the MOEA. In the EA, the crossover is performed based on the unit (gene segment) of single-period paths rather than the unit of cells. Three custom moves in the mutation operator, namely *Remove*, *Insert* and *Reschedule*, to enable revisit of cells and scheduling of take-off times, are developed. Because of the chromosome structure and the genetic operators, the proposed algorithm differs from the existing ones in VRPs, OPs and CPPs.

4) The proposed algorithm is intelligent not only because the chromosome structure and the custom genetic operators are used, but also because the final solution can be selected from a set of mission plans provided by the multi-objective optimization (MOO) according to the user preferences [8]. The proposed solution framework can be used in a variety of surveillance problems, even though most of the surveillance mission planning problems are application-dependent or environment-dependent [11].

The remainder of the paper is structured as follows. Section 2.3 outlines the preliminary information needed to understand the mission scenario and the objectives. Section 2.4 presents the formulation of this optimization problem. The solution technique is developed in Section 2.6. Simulation results are given in Section 2.6. The conclusions are discussed in Section 2.7.

2.3 Problem Description

2.3.1 Mission Scenario

The geographic AOI can be of any shape, but the geographical and dimensional information about AOIs are known a priori. Each AOI to be searched is divided into a number of cells as shown in Fig. 2.2. The shape of a cell is determined by the sensor coverage in one scan: for example, the sensing area of an airborne camera facing downward is designed to be a circular region [16] or a rectangular region [11, 18]. Note that a variety of coverage sweep patterns can be used for a cell and that the shape of a cell is not constrained to be circular as shown in Fig. 2.2 [18, 23]. However, AOI segmentation is not the focus of this paper. The center of a cell is denoted as a point of interest (POI). A cell is considered entirely covered if its center (POI) is visited by an aircraft. In this surveillance problem, we assume that any POI can be visited by at least one aircraft.

The assets consist of a given number of patrolling aircraft. Each aircraft is characterized by its speed of travel, a fixed altitude above ground level, a maximum flight time (constrained by fuel), a minimum and a maximum downtime between two successive flights. The above attributes are given for each aircraft and need not be same for all aircraft. Aircraft with the same speed and the same constraints on flight time and downtime are of the same type. Aircraft are to take off and land from/at the same base station, as shown in Fig. 2.1 and Fig. 2.2. It is assumed that flight paths are collision-free. For the sake of simplicity, no atmospheric condition, such as wind [26], is taken into consideration, but the inclusion of environmental conditions does not change the overall problem formulation.

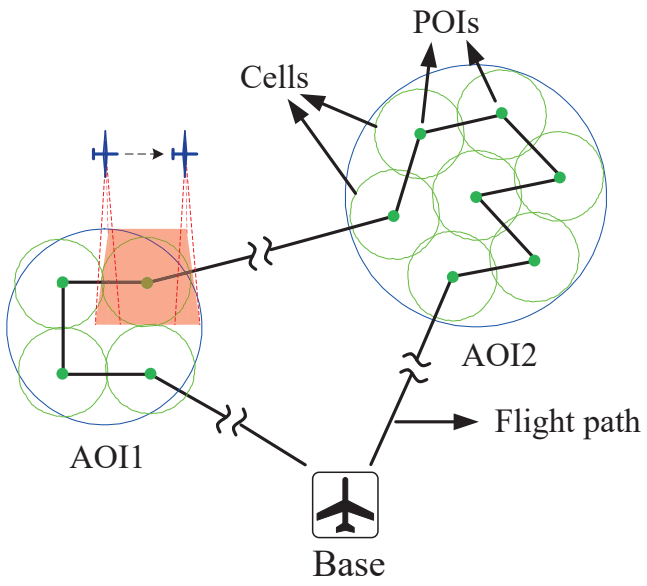


Figure 2.2: Illustration of AOIs, cells, POIs, and a flight path.

There is a camera at the bottom of the aircraft to capture images from the AOIs. The coverage of the camera is studied rather than the detections in images (the measurements). Sensor capabilities (e.g., scan angle, scan direction and coverage radius) are assumed to be constant during the mission time for each aircraft and same for all aircraft. Therefore, the cells in AOIs are of the same size.

Base stations are located out of AOIs (see Fig. 2.1) and their locations are known a priori. An aircraft has already been assigned to a base station. Each base station has one or more aircraft. In the following, when we refer to an aircraft, the corresponding base station is designated and specified.

A mission time is given and each aircraft should land at the base by the end of the mission plan. An aircraft may take off and land multiple times in the mission horizon.

The mission objective is to maximize information collected via AOI coverage: aircraft should strive to cover as many cells as possible and make full use of their flight

time to cover visited or unvisited cells. The persistent monitoring of a particular AOI and the survey of different AOIs are both desired. However, it is always difficult to achieve the above two goals simultaneously when assets are limited. To address this, a revisit requirement to ensure prompt update of surveillance information is used: each cell inside AOIs is to be covered within a given time interval, e.g., within each 8-hour slot.

2.3.2 Mission Plan

A flight is one sortie of an aircraft, which consists of a sequence of waypoints (bases and POIs) that starts from a base, visits a group of POIs and turns back to the same base from which the aircraft departs, as shown in Fig. 2.2. A flight is also referred to as a single-period path (sp) in this paper. For one aircraft, there are multiple flights in the mission plan. The connection of multiple single-period paths with the take-off time for each flight forms a multi-period path (mp). A mission plan consists of the multi-period path for each aircraft. The paths are to be planned and the take-off time schedule is to be determined for each mission plan, in which aircraft survey AOIs cooperatively in space and time.

The visit time point of each POI is recorded along with the planned path, which forms a visiting time sequence. In other words, the mission planner registers aircraft's arrival time at each POI. Once the mission plan is derived, the multiple visits of one POI by all aircraft correspond to a set of visit time, which can be sorted in a chronological order. Then the revisit time interval of this POI can be calculated. Based on the planned paths and the visit time of each POI, the performance of the mission plan can be evaluated, which will be described as follows.

2.3.3 Mission Objective Evaluation

Since the cells are of the same size, we use the number of covered cells (visited POIs) to evaluate the coverage. The mission objective is evaluated by two metrics: 1) the number of non-repeatedly visited POIs in a sliding-window fashion (referred to as the *revisit-unrepeated* metric), and 2) the total number of visited POIs in the mission plan (referred to as the *revisit-total* metric). These two metrics are proposed based on the following assumptions:

1) Even though the mission lasts for a very long time, the user wants to know the overall situation in each AOI as soon as possible. Therefore, covering different POIs/AOIs is more important than continuously covering a particular POI/AOI.

2) The probability of new information gathered from a particular POI increases with time. Revisiting a POI contributes to the information collected. However, revisiting a POI immediately contributes little to the collected information [11]. If an aircraft still has the capacity (e.g., fuel) to cover cells after all POIs are visited, it should continue covering to achieve a greater number of revisited POIs.

Given a mission plan, the *revisit-unrepeated* metric is evaluated in a sliding window fashion [5]. The initial time of the sliding window is the start time of the mission. The size and the step length of the sliding window are fixed, where the length is much smaller than the mission time. The window is moved over the time horizon to evaluate the *revisit-unrepeated* metric. In each time window, revisit to a POI does not increase the *revisit-unrepeated* metric value. Therefore, an aircraft should visit as many different POIs as possible and avoid revisiting within the window if maximizing the *revisit-unrepeated* metric is used as the only objective. However, if an aircraft has enough fuel after all AOIs are covered, it will return to the base instead of fully

using its capacity to cover when maximizing the *revisit-unrepeated* metric is its only objective.

Since the *revisit-unrepeated* metric cannot direct aircraft to revisit POIs and make full use of the surveillance capabilities of aircraft, the *revisit-total* metric is developed as a complement to the *revisit-unrepeated* metric. All visited POIs in the mission plan are counted in the *revisit-total* metric. However, if maximizing the *revisit-total* metric is the only objective, aircraft will keep surveying the nearest AOI without the need to leave it [11]. The result is that no information is gathered from other AOIs in the mission plan. Thus, both the *revisit-unrepeated* metric and the *revisit-total* metric are needed.

2.4 Problem Formulation

2.4.1 Notations and Definitions

The proposed problem is defined on a complete directed graph comprising a set V of vertices or waypoints (that represents the physical locations of POIs and bases) together with a set of edges connecting each pair of vertices. The set V is further partitioned into two subsets: $V = S \cup B$, where S is the set of POIs to be visited and B is the set of base stations. The set of aircraft is denoted as A and the set of AOIs as E. A list of symbols and notations used in this paper is given in Tabel 2.1.

The decision variables are the take-off time and the path selection for each flight. Let t_{b1aw} denote the take-off time of aircraft a in its w-th flight from base b_a, where $w \in W_a$ and $0 \leq t_{b1aw} < T_{\mathrm{m}}$ (the mission start time is used as the zero time reference). Since once the aircraft is specified, the corresponding base of this aircraft is also

Table 2.1: Symbols and Notations

Symbol	Description
b_a	base for aircraft a (known a priori), $a \in A$, $b_a \in B$
S_e	set of POIs inside AOI e, $e \in E$, $S_e \subseteq S$
v_a	speed of aircraft a
d_{ij}	travel distance (by the shortest path) from vertex i to vertex j, $\{i, j\} \in V$
C_{ija}	travel time from vertex i to vertex j by aircraft a. $C_{ija} = d_{ij}/v_a$
T_a^{f}	the maximum flight time of aircraft a
T_a^{dmin}	the minimum downtime of aircraft a between two successive flights
T_a^{dmax}	the maximum downtime of aircraft a between two successive flights
T_{h}	a time slot aircraft landing at base
T_{m}	mission time
T_{r}	required revisit time interval: each POI should be revisited within T_{r}
T_{c}	sliding window size
δ	step length of the sliding window
W_a	set of flights of aircraft a
N_a^W	total number of flights of aircraft a (the cardinality of W_a, $\lvert W_a \rvert = N_a^W$)
F_{iaw}	set of visit time indices of vertex i in the w-th flight of aircraft a, $i \in V$, $a \in A$, $w \in W_a$
K_i	set of visit time indices of POI i in the mission plan by all aircraft, $i \in S$

identified, the subscript of b_a is omitted in t_{b1aw}. In the following, we will omit the subscript of b_a when a and b_a are both in the notation. Let $x_{i\kappa j\mu aw}$ denote the path selection binary decision variable:

$$x_{i\kappa j\mu aw} = \begin{cases} 1 & \text{if the } \kappa\text{-th visit to vertex } i \text{ is followed by the } \mu\text{-th visit} \\ & \text{to vertex } j \text{ in the } w\text{-th flight of aircraft } a,\ i \neq j \\ 0 & \text{otherwise} \end{cases} \tag{2.1}$$

When the path selection decision variables $x_{i\kappa j\mu aw}$ are determined, a path is formed [19, 28]. In the proposed problem, a single-period path for the w-th flight of aircraft

a at base b_a is defined as:

$$\begin{aligned} \mathrm{sp}_{aw} =& \{b_a, \overbrace{k, \cdots, l}^{\text{POIs}}, b_a \mid x_{i\kappa j\mu aw}, \\ & \forall \kappa \in F_{iaw},\ \mu \in F_{jaw},\ \{i,j\} \in V\}, \end{aligned} \tag{2.2}$$

where the starting vertex and the ending vertex are the same base station b_a, κ and μ are the visit time indices of POIs in a single flight.

In the proposed problem, one POI is allowed to be visited more than once in a single-period path. Moreover, an aircraft can fly from POI i to POI j more than once in a single flight to revisit POIs. If $x_{ij} = 1$ is used to denote a visit to vertex i is followed by a visit to vertex j [24, 34], the notation x_{ij} cannot refer to the multiple path selections from vertex i to vertex j caused by revisiting in a single-period path. Hence, the classical notations in VRPs [24] will lead to ambiguity in the formulation of our problem. To address this, $\{\gamma, \eta, \kappa, \mu\}$ are used to denote the visit time indices of POIs. The visit of a POI becomes unique in a single-period path, when the visit time index is associated with the POI.

When the planned single-period path, take-off time, speed of aircraft and the locations of vertices are given, the visit time of POIs in this path and the landing time at the base can be computed. The κ-th visit time of POI i by aircraft a's w-th flight is denoted as $t_{i\kappa aw}$, where $\kappa \in F_{iaw}$. When $i = b_a$ and $\kappa = 1$, the notation $t_{i\kappa aw}$ becomes t_{b1aw}, which is the take-off time. The landing time of aircraft a at base b_a is defined as t_{b2aw}, which also conforms to the format of notation $t_{i\kappa aw}$ (when $i = b_a$, $\kappa = 2$). The formulation of $t_{i\kappa aw}$ will be provided in Section 2.4.2.

The multi-period path for aircraft a is defined as $\mathrm{mp}_a = \{\mathrm{sp}_{aw},\ t_{b1aw},\ \forall w \in W_a\}$. The total number of flights N_a^W of aircraft a is determined when each flight is planned

and connected, and is not known a priori. The take-off time schedule for every aircraft in the mission plan is denoted as $T_0 = \{t_{b1aw},\ \forall a \in A,\ b_a \in B,\ w \in W_a\}$. The mission plan is $P = \{\text{mp}_a,\ \forall a \in A\}$. Therefore, a mission plan is formed by determining the path selection and take-off time decision variables for each flight. These decision variables enable the spatio-temporal cooperation of aircraft to cover AOIs and jointly affect the performance of the mission.

2.4.2 Problem Formulation

We first provide the calculation for the visit time and the revisit time interval of a POI, and then derive the formulations for the objective functions. Finally, the constraints are discussed.

For the w-th flight of aircraft a, suppose the generated path is sp_{aw} and the corresponding take-off time is t_{b1aw}. Consider this path only and ignore the cooperation across multiple aircraft. The time point that POI k is visited by aircraft a for the γ-th time in this path is:

$$\begin{aligned} t_{k\gamma aw} = t_{b1aw} &+ \sum_{i\in V}\sum_{j\in S}\sum_{\kappa\in F_{iaw}}\sum_{\mu\in F_{jaw}} C_{ija}x_{i\kappa j\mu aw} \\ &+ \sum_{l\in V}\sum_{\eta\in F_{law}} C_{lka}x_{l\eta k\gamma aw}, \end{aligned} \tag{2.3a}$$

$$\begin{aligned} &\{u_{i\kappa aw},\ u_{j\mu aw},\ u_{l\eta aw}\} < u_{k\gamma aw}, \\ &\forall a \in A,\ b_a \in B,\ w \in W_a,\ \{i,l\} \in V,\ \{j,k\} \in S, \\ &\kappa \in F_{iaw},\ \mu \in F_{jaw},\ \eta \in F_{law},\ \gamma \in F_{kaw}, \end{aligned} \tag{2.3b}$$

where the additional real variables $\{u_{i\kappa aw},\ u_{j\mu aw},\ u_{l\eta aw},\ u_{k\gamma aw}\}$ in (2.3b) are used to

order all vertices [27], which follow the Miller-Tucker-Zemlin subtour elimination constraints (MTZ-SECs) [22]. The formulations of MTZ-SECs using $\{u_{i\kappa aw},\ u_{j\mu aw},\ u_{l\eta aw},\ u_{k\gamma aw}\}$ will be provided later in this section. The total number of visited POIs (with repeat) in sp_{aw} can be expressed as

$$n_{aw} = \sum_{i\in V}\sum_{j\in S}\sum_{\kappa\in F_{iaw}}\sum_{\mu\in F_{jaw}} x_{i\kappa j\mu aw}. \tag{2.4}$$

The variable $u_{i\kappa aw}$ satisfies

$$u_{i1aw} = 1,\ i = b_a, \tag{2.5a}$$

$$2 \le u_{i\kappa aw} \le n_{aw} + 1,\ i \in S,\ \kappa \in F_{iaw}, \tag{2.5b}$$

and the variable $u_{i\kappa aw}$ denotes the position of vertex i in path sp_{aw}, while the visit time index of vertex i is κ [27].

The visit time point of each POI in a mission plan can be calculated by (2.3) when flights for all aircraft are planned. The visited POIs in the mission plan can be sorted by their visit time in chronological order. For POI i, the chronological set of visit times by all aircraft in the mission plan can be formed, which is denoted as K_i.

Let $U_{i\varepsilon P}$ denote the position of POI i in mission plan P, where the visit time index of POI i is ε ($\varepsilon \in K_i$). Let $t(U_{i\varepsilon P})$ denote the time point at position $U_{i\varepsilon P}$, which can be obtained after sorting $t_{i\kappa aw}$ for all $a \in A$, $b_a \in B$, $w \in W_a$, $\kappa \in F_{iaw}$. The revisit time interval between the $(\varepsilon-1)$-th visit and the ε-th visit of POI i in the mission plan is expressed as:

$$\mathrm{rvt}(i,\varepsilon) = t(U_{i\varepsilon P}) - t(U_{i(\varepsilon-1)P}), \quad \varepsilon > 1,\ \varepsilon \in K_i, \tag{2.6}$$

where $\mathrm{rvt}(\cdot)$ stands for the revisit time interval. The first visit time of POI i is denoted as $\mathrm{rvt}(i,1) = t(U_{i1P})$, $\forall i \in S$. It is required that the first visit of each POI happen within T_r after the mission starts.

In the following, we provide the evaluation of the *revisit-unrepeated* metric in the sliding windows.

A sliding window with size T_c and step length δ moves over the mission time horizon. The total number of windows is $N_\tau = (T_\mathrm{m} - T_\mathrm{c})/\delta + 1$. The value of δ should guarantee that the end time of the last evaluation window is the mission time. Usually, δ is less than T_c so that all visited POIs in the mission plan are included in the sliding windows. The τ-th evaluation window starts from $(\tau-1)\delta$ and ends at $T_\mathrm{c} + (\tau-1)\delta$, where $\tau = 1, 2, \cdots, N_\tau$.

The visited POIs in mission P, whose visit time drops into the τ-th evaluation window, is the set of POIs under test:

$$\begin{aligned} Q(\tau) = &\{\forall i \in S \mid (\tau-1)\delta \le t(U_{i\varepsilon P}) \le T_\mathrm{c} + (\tau-1)\delta, \\ &\forall \varepsilon \in K_i\}. \end{aligned} \tag{2.7}$$

For POIs in $Q(\tau)$, define a binary variable $y(i,\tau)$

$$y(i,\tau) = \begin{cases} 1 & \text{if POI } i \in Q(\tau) \\ 0 & \text{otherwise} \end{cases} \tag{2.8}$$

The condition for $y(i,\tau) = 1$ means that POI i is visited at least once in the τ-th window. Therefore, repeated visit to a POI in the evaluation window does not increase the value of $y(\cdot)$.

The *revisit-unrepeated* metric in the τ-th sliding window is defined as the sum of $y(i,\tau)$ with respect to (w.r.t.) all POIs: $\sum_{i\in S} y(i,\tau)$. To evaluate the *revisit-unrepeated* metric in the mission plan, we present a maximin criteria [19] that maximizes the minimum *revisit-unrepeated* metric in all evaluation windows. The minimum *revisit-unrepeated* metric is defined as:

$$f_1 = \min(\sum_{i\in S} y(i,\tau),\ \forall \tau = 1, 2, \cdots, N_\tau). \tag{2.9}$$

The first objective is expressed as:

$$\max_{P}\ f_1. \tag{2.10}$$

There are other ways to evaluate the *revisit-unrepeated* metric, e.g., $\sum_{\tau=1}^{N_\tau}\sum_{i\in S} y(i,\tau)$. Since there are overlapping time slots between two evaluation windows, adding up $\sum_{i\in S} y(i,\tau)$ in all windows is not used in our method but the maximin criteria as shown in (2.9) and (2.10) are used.

The evaluation of the *revisit-total* metric is the sum of all visited POIs in the mission plan, which is expressed as:

$$\begin{aligned} f_2 =& \sum_{a\in A}\sum_{w\in W_a}\sum_{i\in V}\sum_{j\in S}\sum_{\kappa\in F_{iaw}}\sum_{\mu\in F_{jaw}} x_{i\kappa j\mu aw} \\ =& \sum_{a\in A}\sum_{w\in W_a} n_{aw}. \end{aligned} \tag{2.11}$$

The second objective, maximizing the total number of visited POIs, is expressed as:

$$\max_{P}\ f_2. \tag{2.12}$$

Note that the decision variables in (2.10) and (2.12) are the path selection and the take-off time for each flight. Here, mission plan P is placed in (2.10) and (2.12) as the argument for simplicity.

Formulations of constraints are listed as follows:

$$\sum_{i\in S}\sum_{\kappa\in F_{iaw}} x_{k1i\kappa aw} = \sum_{j\in S}\sum_{\mu\in F_{jaw}} x_{j\mu k2aw} = 1, \quad \forall a \in A,\ w \in W_a,\ k = b_a \tag{2.13}$$

$$\sum_{i\in S}\sum_{\kappa\in F_{iaw}} x_{k1i\kappa aw} = 0, \quad \forall a \in A,\ w \in W_a,\ k \in B \setminus \{b_a\} \tag{2.14}$$

$$\sum_{i\in V}\sum_{\kappa\in F_{iaw}} x_{i\kappa k\gamma aw} = \sum_{j\in V}\sum_{\mu\in F_{jaw}} x_{k\gamma j\mu aw} \leq 1, \quad \forall a \in A,\ w \in W_a,\ k \in S,\ \gamma \in F_{kaw} \tag{2.15}$$

$$\sum_{i\in V}\sum_{j\in V}\sum_{\kappa\in F_{iaw}}\sum_{\mu\in F_{jaw}} C_{ija} x_{i\kappa j\mu aw} \leq T_a^{\mathrm{f}}, \quad \forall a \in A,\ w \in W_a \tag{2.16}$$

$$
\begin{aligned}
&t_{b1a(w+1)} \geq t_{b1aw} + \\
&\sum_{i\in V}\sum_{j\in V}\sum_{\kappa\in F_{iaw}}\sum_{\mu\in F_{jaw}} C_{ija}x_{i\kappa j\mu aw} + T_a^{\mathrm{dmin}}, \\
&\forall a \in A,\ b_a \in B,\ w = 1, \cdots, N_a^W - 1
\end{aligned}
\tag{2.17}
$$

$$
\begin{aligned}
&t_{b1a(w+1)} \leq t_{b1aw} + \\
&\sum_{i\in V}\sum_{j\in V}\sum_{\kappa\in F_{iaw}}\sum_{\mu\in F_{jaw}} C_{ija}x_{i\kappa j\mu aw} + T_a^{\mathrm{dmax}}, \\
&\forall a \in A,\ b_a \in B,\ w = 1, \cdots, N_a^W - 1,
\end{aligned}
\tag{2.18}
$$

$$
\begin{aligned}
&t_{b1aw} + \sum_{i\in V}\sum_{j\in V}\sum_{\kappa\in F_{iaw}}\sum_{\mu\in F_{jaw}} C_{ija}x_{i\kappa j\mu aw} \leq T_{\mathrm{m}}, \\
&\forall a \in A,\ b_a \in B,\ w = N_a^W
\end{aligned}
\tag{2.19}
$$

$$
\mathrm{rvt}(i,\varepsilon) \leq T_{\mathrm{r}}, \quad \forall i \in S,\ \varepsilon \in K_i \tag{2.20}
$$

$$
\begin{aligned}
&2 \leq u_{k\gamma aw} \leq n_{aw} + 1, \\
&\forall a \in A,\ w \in W_a,\ k \in S,\ \gamma \in F_{kaw}
\end{aligned}
\tag{2.21}
$$

$$
\begin{aligned}
&u_{l\eta aw} - u_{k\gamma aw} + 1 \leq n_{aw}\left(1 - x_{l\eta k\gamma aw}\right), \\
&\forall a \in A,\ w \in W_a,\ \{l,k\} \in S,\ \eta \in F_{law},\ \gamma \in F_{kaw}
\end{aligned}
\tag{2.22}
$$

$$
\begin{aligned}
&x_{i\kappa j\mu aw} \in \{0,1\},\ t_{b1aw} \geq 0, \qquad \forall a \in A,\ b_a \in B, \\
&w \in W_a,\ \{i,j\} \in V,\ \kappa \in F_{iaw},\ \mu \in F_{jaw}
\end{aligned}
\tag{2.23}
$$

Constraint (2.13) guarantees that an aircraft starts from its base station and ends at the same base station in each flight. Constraints (2.13) and (2.14) guarantee that an aircraft only take off from the base specified to this aircraft, where the backslash symbol \ in (2.14) denotes a set difference.

Constraint (2.15) guarantees the connectivity of POIs in each flight. When the γ-th visit to POI k in the w-th flight of aircraft a is given, $\sum_{i\in V}\sum_{\kappa\in F_{iaw}} x_{i\kappa k\gamma aw}$ equals 0 or 1.

When a path is planned, the continuity of time sequence is guaranteed:

$$
\begin{aligned}
&t_{i\kappa aw} + C_{ija} - t_{j\mu aw} \leq M\left(1 - x_{i\kappa j\mu aw}\right), \\
&\forall a \in A,\ w \in W_a,\ \{i,j\} \in V,\ \kappa \in F_{iaw},\ \mu \in F_{jaw},
\end{aligned}
\tag{2.24}
$$

where M is a very large positive constant ($M \gg T_{\mathrm{m}}$). If there exists a path selection from the κ-th visit of vertex i to the μ-th visit of vertex j in aircraft a's w-th flight, $x_{i\kappa j\mu aw} = 1$, and $t_{i\kappa aw} + C_{ija} - t_{j\mu aw} = 0$.

Constraint (2.16) ensures that the travel time of each flight is no more than the maximum flight time.

Constraints (2.17) and (2.18) ensure that the downtime between two successive flights is no less than the minimum downtime and no greater than the maximum downtime, respectively. The wait time in addition to the minimum downtime before

the $(w+1)$-th flight of aircraft a is denoted as

$$h_{aw} = t_{b1a(w+1)} - t_{b1aw} - \sum_{i\in V}\sum_{j\in V}\sum_{\kappa\in F_{iaw}}\sum_{\mu\in F_{jaw}} C_{ija}x_{i\kappa j\mu aw} - T_a^{\mathrm{dmin}}, \tag{2.25}$$

where $h_{aw} \leq T_a^{\mathrm{dmax}} - T_a^{\mathrm{dmin}}$. It will be shown in Section 2.5 that the take-off time schedule is generated by determining the take-off time of the first flight t_{b1a1} and the following wait time h_{aw} $\left(w = 1, \cdots, N_a^W - 1\right)$.

Constraint (2.19) guarantees that the time point of aircraft returning to base for the last flight is not greater than the mission time.

Constraint (2.20) guarantees that each POI is visited for the first time within T_{r} after the mission starts (for the case $\varepsilon = 1$), and that each POI is revisited within T_{r} for the following visits (for the cases $\varepsilon > 1$). The revisit time interval $\mathrm{rvt}(\cdot)$ is calculated as (2.6).

Constraints (2.21) and (2.22) are used to prevent subtours in each single-period path, which are derived from MTZ-SECs [1, 22]. The total number of visited POIs in the w-th flight of aircraft a is n_{aw} as in (2.4), where the repetition of POIs is considered.

2.5 Solution Technique

The proposed mission planning problem is more complex than the existing VRPs and OPs because of the revisiting in single-period paths and the spatio-temporal cooperation of aircraft in multi-period paths. Furthermore, only small instances of the VRP or OP can be solved exactly [30, 6]. It is clear that one cannot solve this path

planning and scheduling with the above formulation. We have therefore opted for the development of an MOEA. The choice is motivated by 1) the success of EA in VRP and its variants [10, 35], and 2) the ability of MOEA to find multiple Pareto-optimal solutions or converge near the true Pareto-optimal set in one single simulation run [8].

The flowchart of the path planning and scheduling algorithm is illustrated in Fig. 2.3. The algorithm consists of four parts: pre-processing, multi-objective evolution, decision making, and waypoint transformation. This paper mainly describes methods in the pre-processing and the multi-objective evolution.

At the very beginning, parameters are initialized, AOIs are divided into cells, and POIs are generated. Single-period paths are generated by solving a series of OPs, in which revisit to POIs and take-off time schedule are not considered. For one aircraft, these single-period paths are connected to form the multi-period path, where the take-off time of each flight is randomly selected under constraints.

A set of mission plans is generated as the initial population in the EA, where the paths for all aircraft are evaluated as shown in (2.9) to (2.12). The revisit constraint (2.20) is transformed into an objective function in the EA, which will be described later in this section.

During the evolution of individuals, rescheduling of take-off time and revisit to POIs are enabled by the mutation operator; exchanging single-period paths between mission plans is conducted by the crossover operator. These genetic operators differ from standard mutation and crossover [15] and are especially designed for the proposed problem. Objective functions are evaluated and sorted following the non-dominated sorting genetic algorithm II (NSGA-II) procedure [8].

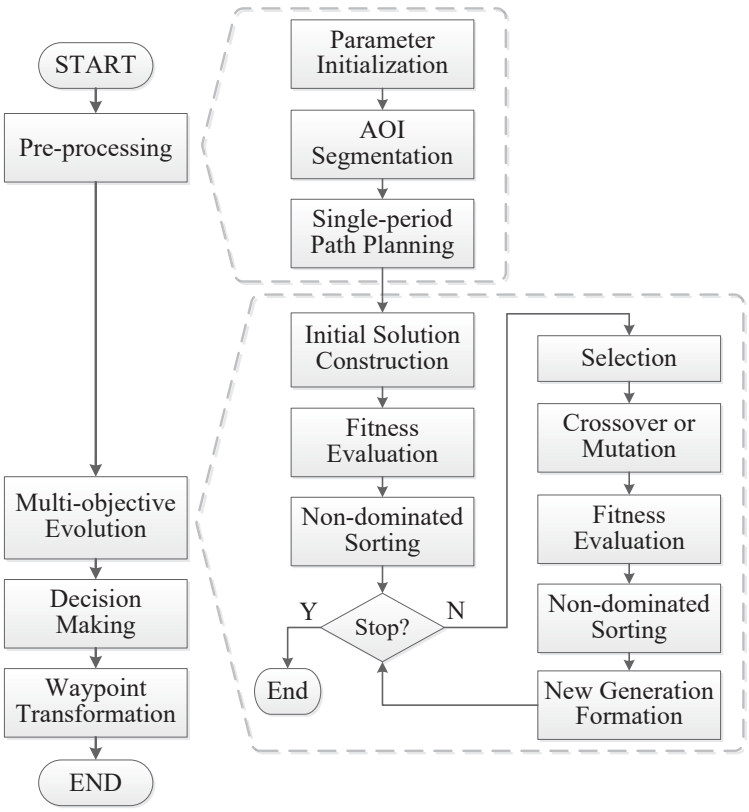


Figure 2.3: Flowchart of the coverage path planning and scheduling algorithm.

The best mission plan is selected by a weighted-sum model in decision making from the MOO solution set. The mission plan generated by the steps described above is a sequence of vertices, which correspond to ground locations/waypoints. Finally, paths are transformed into aerial waypoints for each aircraft based on the aircraft motion model (e.g., Dubins airplane) [13, 20] and sensor parameters [18]. However, waypoint transformation is not the focus of this paper.

2.5.1 Mission Plan Structure

The mission plan structure is also the genetic representation of an individual (chromosome), which is shown in Fig. 2.4. The mission plan for one aircraft is made up of

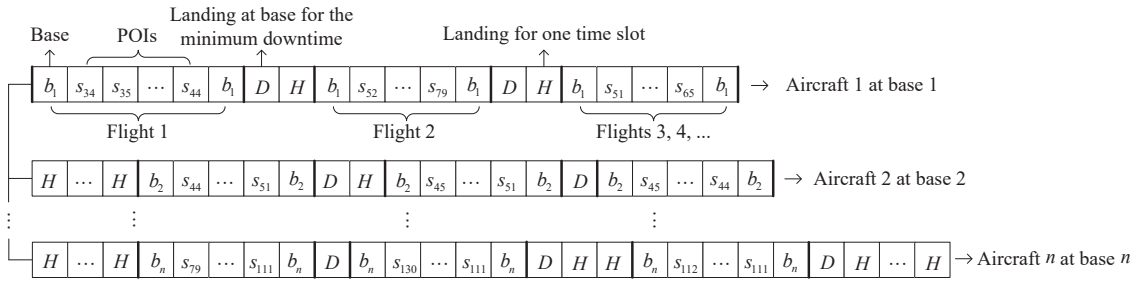


Figure 2.4: Mission plan structure.

multiple single-period paths connected by elements representing the state of aircraft landing at the base. The wait time before the first flight and the wait time in addition to the minimum downtime are discretized into time slots with a fixed length of T_{h}. The genetic representation of the minimum downtime is D and that of the discrete time slot is H. The genetic representation of POI i is s_i and that of base for aircraft a is b_a.

As shown in Fig. 2.4, the number of H affects the take-off time: aircraft can take off immediately after the minimum downtime or stay at the base for several time slots. When the time left is not enough for the aircraft to visit the nearest POI at the end of the mission, this remaining time is denoted as several time slots as well. This is because taking off, in this case, will not increase the objective values. The determination of the number of time slots before each flight will be described later in this section.

Given the mission plan structure, a time table for each aircraft, which consists of the take-off time of each flight, the visit time of POIs, the landing time of each flight, and minimum downtime and the wait time slots, can be computed. Elements in the mission plan and time points in the time tables have one-to-one correspondence. The mission plan structure, along with the time tables, effectively handles the path

selection and the take-off time decision at the same time. It also manipulates the variable-length chromosome, which makes the EA easy to operate [14, 29].

2.5.2 Single-period Path Planning

Single-period paths are the construction blocks in the mission plan. By connecting the single-period paths and determining the take-off time for each flight, the initial population to EA can be generated, which has the potential to achieve the mission goal, as described in (2.9) to (2.12). Hence, single-period path planning is preliminary to the construction of initial population of EA.

The path planning for each aircraft is formulated as a series of OPs. In OP, each POI is associated with a score, the goal is to determine a path with a maximum sum of the collected scores in limited length [34]. Each POI is visited at most once in these paths.

Let T_a^{lmt} denote a set of time limits for aircraft a: $T_a^{\text{lmt}} = \{t_a^{\text{lmt}}(\phi),\ \phi = 1, 2, \cdots\}$, which may range from a small value (e.g., the time needed to fully cover the nearest AOI by aircraft a) to the maximum flight time T_a^{f} with a fixed step size. Within the given time interval $t_a^{\text{lmt}}(\phi)$, paths are designed to maximize the collected scores associated with POIs. Initially, POIs are assigned with the same scores to generate the first path by solving the OP. When the path is generated, unvisited POIs within this aircraft's reach (limited by $t_a^{\text{lmt}}(\phi)$) are assigned with higher scores than already visited POIs; an OP is solved again to visit POIs with the updated scores. The score of a POI depends on whether this POI is visited or not in the already planned paths. This process continues until all POIs within the aircraft's reach are visited. Then, the time limit increases and paths are planned by the above approach once again.

By assigning higher scores to unvisited POIs, single-period paths are generated to cover unvisited POIs within the aircraft's reach. Paths with different POIs enrich the diversity of single-period paths. Unvisited POIs in a flight can be explored either by the same aircraft's subsequent flights or by other aircraft. Because of the different time limits, paths of different time length are included in the mission plan. By connecting these paths of different lengths, multi-period paths with different flight number N_a^W can be generated, which enrich the diversity of multi-period paths. When the time limit $t_a^{\text{lmt}}(\phi)$ is less than the maximum flight time, it is possible to use the remaining time $(T_a^{\text{f}} - t_a^{\text{lmt}}(\phi))$ to revisit POIs by the mutation operator. Therefore, solving the proposed series of OPs ensures the diversity of the mission plan.

The path planning decision variable in OP is defined as:

$$x_{ij} = \begin{cases} 1 & \text{if a visit to vertex } i \text{ is followed by a visit to vertex } j \\ 0 & \text{otherwise} \end{cases} \tag{2.26}$$

Given the scores for each POI, one OP for aircraft a at base b_a is formulated as:

$$\max_{x_{ij}} \quad \sum_{i \in S} \sum_{j \in V} \text{sc}(i) x_{ij} \tag{2.27}$$

$$s.t. \quad \sum_{j \in S} x_{kj} = \sum_{i \in S} x_{ik} = 1, \quad k = b_a \tag{2.28}$$

$$\sum_{i \in V} x_{ik} = \sum_{j \in V} x_{kj} \leq 1, \quad \forall k \in S \tag{2.29}$$

$$\sum_{i \in V} \sum_{j \in V} C_{ija} x_{ij} \leq t_a^{\text{lmt}}(\phi), \ \forall t_a^{\text{lmt}}(\phi) \in T_a^{\text{lmt}} \tag{2.30}$$

$$2 \leq u_k \leq \sum_{i \in V} \sum_{j \in V} x_{ij}, \quad \forall k \in S \tag{2.31}$$

$$u_l - u_k + 1 \leq \sum_{i \in V} \sum_{j \in S} x_{ij} \left(1 - x_{lk}\right),$$

$$\forall \{l, k\} \in S \tag{2.32}$$

$$x_{ij} \in \{0, 1\}, \quad \forall \{i, j\} \in V \tag{2.33}$$

where $\mathrm{sc}(i)$ is the score assigned to POI i, u_i is the position of vertex i in the path [27, 34]. The purpose of objective function (2.27) is to maximize the total collected score. Constraint (2.28) guarantees that the paths start and end at the same base. Constraint (2.29) ensures the connectivity of the POIs in the path. Constraint (2.30) ensures the limited time budget. Constraints (2.31) and (2.32) are necessary to prevent subtours [34]. The constraints (2.28), (2.29), (2.30), (2.31) and (2.32) correspond to (2.13), (2.15), (2.16), (2.21) and (2.22), respectively. A five-step heuristic is used to solve the OP [3].

For the series of OPs, the solution is described in Algorithm 1, where the updating of scores and the traversing of time limits are shown. In Algorithm 1, C_{bia} represents the travel time from base b_a to POI i by aircraft a; S_{ab} is a set of POIs within the reach of aircraft a taking off from base b_a; $S(\text{unvisit})$ is a set of unvisited POIs in S_{ab}; $\mathrm{sp}_{\mathrm{cnt}}^{\phi}$ denotes the single-period path generated in the cnt-th iteration of the ϕ-th time limit; $\emptyset$ denotes the empty set. Note that different values of ρ ($\rho > 1$) lead to different paths: the greater the ρ, the more unvisited POIs will be included in the path. All planned paths are stored to ensure the population diversity in the EA.

Note that no-fly zones [9, 11] outside AOIs are not considered in the mission scenario, but no-fly zones can be handled in our solution technique. A discussion on no-fly zones in single-period path planning is provided in the Appendix.

Algorithm 1 Single-period path planning

Input:

Aircraft a. Base station b_a. Set of POIs S.

Fixed score c. Factor ρ. Set of time limits T_a^{lmt}.

Output:

Single-period paths $\text{sp}_{\text{cnt}}^{\phi}$,

$\phi = 1, 2, \cdots, |T_a^{\text{lmt}}|$, $\text{cnt} = 1, 2, \cdots$.

for $t_a^{\text{lmt}}(\phi) \in T_a^{\text{lmt}}$ **do**

  $S_{ab} = \left\{ i | 2C_{bia} \leq t_a^{\text{lmt}}(\phi),\ i \in S \right\}$

  $S(\text{unvisit}) = S_{ab}$

  Counter $\text{cnt} = 1$

  Score $\text{sc}(i) = c$, $\forall i \in S_{ab}$

  while $S(\text{unvisit}) \neq \emptyset$ **do**

    Solve OP: (2.27)–(2.33), generate and save $\text{sp}_{\text{cnt}}^{\phi}$

    Update $S(\text{unvisit}) = \left\{ i | i \in S_{ab} \ \&\ i \notin \bigcup_{\text{cnt}} \text{sp}_{\text{cnt}}^{\phi} \right\}$

    Update $\text{sc}(i) = \rho^{\text{cnt}} c$, $\forall i \in S(\text{unvisit})$

    $\text{cnt} = \text{cnt} + 1$

  end while

end for

2.5.3 Initial Population Construction

Instead of planning the path by randomly picking up POIs and randomly setting the take-off time, well-designed single-period paths are used as the construction blocks for the initial population, and the take-off time for each flight is randomly configured with constraints. For one aircraft, its single-period paths are connected as shown in Fig. 2.4. The path for each flight is randomly selected from the results of single-period path planning except the path for the last flight: the selection of a single-period path for the last flight is constrained by (2.19). The number of additional wait time slots is determined such that constraints (2.17) and (2.18) are satisfied.

As mentioned in Section 2.5.1, the number of wait time slots affects the take-off time. There are many ways to determine the number of wait time slots for the initial population. For example, a random number l_a can be generated from a discrete uniform distribution $\mathcal{U}\{0, (T_a^{\mathrm{dmax}} - T_a^{\mathrm{dmin}})/T_{\mathrm{h}}\}$, where $t_{b1aw} = l_a T_{\mathrm{h}}$ is the wait time. Another way is to generate a random number l_a from the rounding result of a truncated normal distribution with zero mean and $(T_a^{\mathrm{dmax}} - T_a^{\mathrm{dmin}})/(mT_{\mathrm{h}})$ standard deviation conditioned on $0 \leq l_a \leq (T_a^{\mathrm{dmax}} - T_a^{\mathrm{dmin}})/T_{\mathrm{h}}$, where m is a positive constant.

The number of wait time slots l_a is scheduled in the initial population construction and rescheduled in the mutation operator. During our simulations, it was observed that mission plans with less additional wait time achieve better objective values. Therefore, generating l_a from the truncated normal distribution is used in our approach, because the value of m can be tuned so that more mission plans with small wait time can be generated in the population.

Because of the single-period path planning and initial population construction, the initial mission plans satisfy constraints (2.13) to (2.19), (2.21) and (2.22). Individuals

in the initial population do not necessarily satisfy the revisit constraint (2.20). The handling of (2.20) is described as follows.

2.5.4 Infeasibility Handling

In the EA, the feasibility w.r.t. constraints (2.15) to (2.19), (2.21) and (2.22) is checked in the genetic operators and infeasible paths or mission plans are discarded. The feasibility w.r.t. constraints (2.13) and (2.14) is guaranteed in single-period path planning and will not change by the genetic operators. The revisit constraint (2.20) is transformed into an objective function in the EA for the following reasons:

1) It is possible that the given assets cannot satisfy the revisit constraint (2.20) for all POIs, i.e., there is no feasible solution.

2) The scale of the problem (i.e., the number of POIs, the time horizon of the mission plan, etc.) is too large to verify the existence of feasible solutions w.r.t. constraint (2.20).

If the actual revisit time interval $\mathrm{rvt}(i,\varepsilon)$ (see (2.6)) is greater than the required revisit time T_r, the violated time is $\mathrm{rvt}(i,\varepsilon) - T_\mathrm{r}$. The sum of violated revisit time is to be minimized by path planning, take-off time scheduling and aircraft cooperation. Let f_3 denote the negative value of the sum of the violated revisit time:

$$f_3 = -\sum_{i\in S}\sum_{\varepsilon\in K_i} \max\left(\mathrm{rvt}(i,\varepsilon) - T_\mathrm{r}, 0\right). \tag{2.34}$$

The objective is expressed as:

$$\max_{P}\ f_3. \tag{2.35}$$

Therefore, constraint (2.20) is transformed into (2.35) in the EA. The multi-objective function is $\max_P \ (f_1, f_2, f_3)$.

2.5.5 Multi-objective Evolution

The MOEA is based on the NSGA-II framework [8]. As shown in Fig. 2.3, the following evolutionary process is performed in each generation: 1) calculate the fitness functions (2.9), (2.11) and (2.34), and select parents that are a fit for reproduction, 2) perform crossover and mutation operators on the selected parents to produce offspring, and calculate the fitness of offspring, 3) form the intermediate chromosome, which is a concatenation of the current population and the offspring population, and 4) non-domination sort the intermediate chromosome, replacing the unfit individuals with the fit individuals to maintain a constant population size.

Selection: Many selection techniques, such as tournament selection [8] and roulette wheel selection [35], are available. A binary tournament selection is used in this paper. Constrained NSGA-II (constraint-handling approach) is utilized to rank the solutions [8].

Crossover: Crossover happens with a probability p_{c}. If crossover does not happen, mutation is conducted. From the first flight to the last one, the crossover is performed by exchanging parents' single-period paths (same aircraft, same base station) with a certain probability $p_{\mathrm{c}}^{\mathrm{s}}$ successively, whereas the take-off time of each parent does not change. The time tables of the offspring are updated as the crossover carrying through. If an offspring's actual mission time exceeds T_{m} after a crossover of two single-period paths, this crossover is undone and the single-period paths are reverted to their original (last) positions in the chromosome.

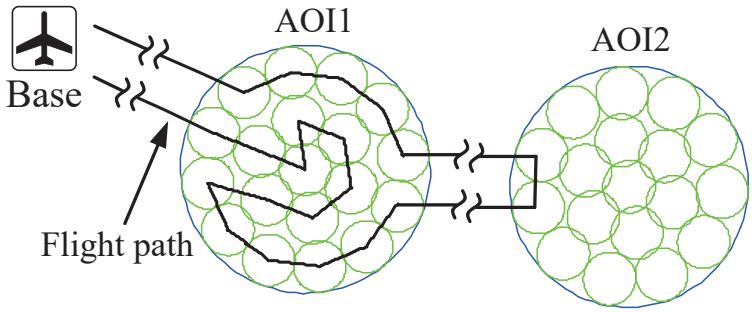


Figure 2.5: An example of single-period path. Visited POIs in AOI2 will be removed by *Remove*, and the remaining flight time after *Remove*, especially the travel time between AOI1 and AOI2, is used to revisit POIs in AOI1 when *Insert* is implemented.

Mutation: The custom mutation operator described in Algorithm 2 is enforced. Three moves are designed for mutation, namely *Remove*, *Insert*, and *Reschedule*.

The goal of *Remove* is, by giving up some visited POIs in a single-period path, to make time for possible revisits of POIs. For instance, the result of *Remove* is that the POIs in AOI2 will not be visited in Fig. 2.5, and the remaining flight time, especially the travel time between AOI1 and AOI2, will be used to revisit POIs in AOI1 (enabled by *Insert*). Line 7 in Algorithm 2 indicates that POIs in the single-period path of aircraft a's w-th flight (denoted as sp_{aw}), which belong to AOI e but do not constitute the whole POI set in AOI e, are removed. Thus, the spared time can be used to revisit other POIs in the path.

The revisit to POIs is enabled by *Insert*. Some successive POIs within the same AOI in a single-period path are selected as a path segment and inserted into this path, as shown in lines 12 and 13 in Algorithm 2. The length of the path segment can be randomly generated from 1 to $|S_e|$. The locus, where the gene segment is inserted to, should avoid the repeat of the same POI in two successive loci.

The take-off time is changed, i.e., the number of T_{h} is generated again (see Section

Algorithm 2 The custom mutation operator

Input:

A chromosome (i.e., mission plan) P contains
multi-period paths mp,
single-period paths sp, and
take-off time schedule T_0.
Remove probability p_{rm}
Insert probability p_{in}
Reschedule probability p_{rs}

Output:

The mutated chromosome P^* contains
the mutated multi-period paths mp^*,
the mutated single-period paths sp^*, and
the mutated take-off time schedule ${T_0}^*$.

for $a \in A$ **do**
  for $w \in W_a$ **do**
    Generate a random number $r \sim \mathcal{U}(0, 1)$
    if $r < p_{\text{rm}}$ **then**
      for $e \in E$ **do**
        if $\exists\{\forall i \in \text{sp}_{aw}\} \subsetneq S_e$ **then**
          Remove $\{\forall i \in \text{sp}_{aw}\} \subsetneq S_e$
          break
        end if
      end for
    else if $p_{\text{rm}} \leq r < p_{\text{rm}} + p_{\text{in}}$ **then**
      Randomly choose an AOI e and a gene segment (GS) within this AOI in a single-period path: $\text{GS} \subseteq S_e \subseteq \text{sp}_{aw}$
      Insert GS after a locus to sp_{aw}
    else
      Reschedule t_{b1aw}
    end if
    Update sp^*_{aw}, mp^*_a and corresponding time tables
    if (travel time of $\text{sp}^*_{aw} > T^{\text{f}}_a$ ||
      mission time of $\text{mp}^*_a > T_{\text{m}}$) **then**
      Undo this mutation move; revert to the original paths and time tables
    end if
  end for
end for

2.5.3).

If an offspring violates the maximum flight time constraints or mission time constraints after mutation, the mutation move is undone and the paths are reverted to the original (last) feasible ones, as shown in lines 18 to 20. These three moves can be viewed as heuristic operators embedded in the EA, which take full advantage of the peculiarities of the proposed problem.

2.5.6 Decision Making

When the evolution terminates, a set of solutions are generated and the objective values of these solutions are known. Then, we can see if there exist feasible solutions (i.e., $f_3 = 0$) or not w.r.t. constraint (2.20). A weighted-sum model can be used to integrate the objectives into a single index [33]:

$$\max_{P} \ \boldsymbol{\omega}^T \hat{\boldsymbol{f}}, \tag{2.36}$$

where $\boldsymbol{\omega} = [\omega_1, \omega_2, \omega_3]^T$, $\omega_1 + \omega_2 + \omega_3 = 1$, $0 \leq \omega_1, \omega_2, \omega_3 \leq 1$, and $\hat{\boldsymbol{f}} = \left[\hat{f}_1, \hat{f}_2, \hat{f}_3\right]^T$. The superscript ^{T} denotes matrix transpose and $\hat{f}_n$ denotes the normalized f_n, where $n = 1, 2, 3$. The selection of weights depends on the user's preferences and priorities. The highest-ranking solution with the greatest weighted-sum value will be selected as the mission plan.

If there is no feasible solution w.r.t. constraint (2.20) in the final solution set, i.e., $f_3 < 0$ for all solutions, the solution with the minimum revisit violation can be selected (by setting $w_3 \gg w_1 + w_2$) if the user emphasizes the revisit constraint. If feasible solutions exist, the mission plan will be selected from these feasible solutions,

where $w_1 \geq w_2$, because promptly updating information collected from different POIs is more important than the total number of visited POIs.

The transportation cost (total travel distance) is evaluated as:

$$f_4 = \sum_{a\in A}\sum_{w\in W_a}\sum_{i\in V}\sum_{j\in V}\sum_{\kappa\in F_{iaw}}\sum_{\mu\in F_{jaw}} d_{ij}x_{i\kappa j\mu aw}. \tag{2.37}$$

If multiple highest-ranking solutions with the same weighted-sum value are found, the one with the minimum transportation cost can be selected as the mission plan.

Note that by adding f_4 to the objective functions $\max_P (f_1, f_2, f_3, -f_4)$, minimizing transportation cost joins the optimization as the fourth objective and the proposed algorithm becomes more complex. As the number of objectives increases, the NSGA-II framework may not work efficiently, and the many-objective evolutionary algorithm NSGA-III can be implemented [7, 17]. In this paper, $\max_P \ -f_4$ is not used in the EA.

2.6 Simulations

2.6.1 Mission Scenario

There is no benchmark problem in the literature for the comparison of path planning and scheduling algorithms [11, 23]. Two simulation environments are used in this section for performance evaluation.

In the first simulation scenario, three rectangular regions are simulated as AOIs (see Fig. 2.6), the sizes of which are 200×160 km^2, 80×80 km^2, and 160×240 km^2. Two aircraft are assigned to two base stations, which are located at $(0, 80)$ km and

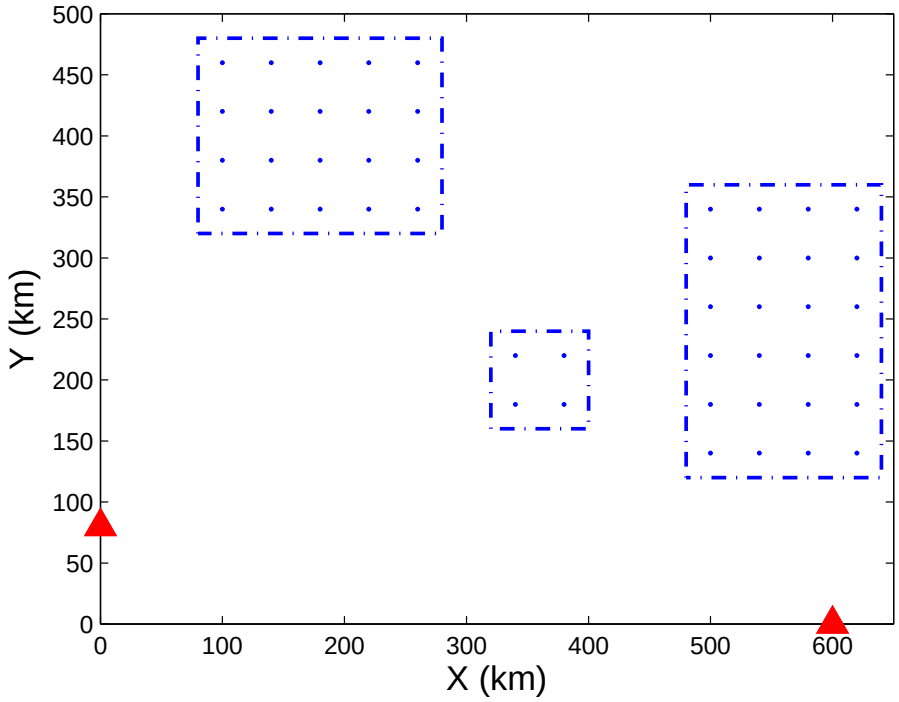


Figure 2.6: The first simulation scenario: locations of AOIs, POIs, and base stations.

$(600, 0)$ km, respectively. Aircraft are of the same type. Aircraft fly at a constant speed of 612 km/h with 4-hour maximum flight time. The minimum downtime is 1 hour and the maximum downtime is 1.5 hours. Dots inside the AOIs in the figure are POIs, which are the result of AOI segmentation. The size of each cell is 40×40 km^2. Thus, there are 48 cells in total. The mission lasts for 24 hours. Each POI in the AOIs must be revisited within 8 hours.

The mission scenario of the second simulation is based on a real-world case study, which will be shown later in this section.

2.6.2 Termination Condition

Since 1) it is not feasible to compute the global optimal solution of the proposed problem [6, 30], and 2) the NSGA-II framework has the ability to find a diverse set of solutions and converge near the true Pareto-optimal set [8], we specify a termination condition such that the solution at termination is close to the optimal one. The weighted-sum model (2.36) is implemented at the end of each generation. The best weighted-sum value of objectives is used to determine the termination of the evolution. If the maximum difference of the best (largest) weighted-sum fitness among N_t successive iterations/generations drops below a given threshold, the EA is terminated.

In the following, different values of crossover and mutation probabilities in two MOOs solution strategies are used to investigate the convergence of the path planning and scheduling problem. The values of crossover probability p_{c} are 0.3, 0.5, and 0.7; the corresponding mutation probability is $(1 - p_{\mathrm{c}})$.

The proposed algorithm based on NSGA-II can be viewed as a non-dominated sorting MOO strategy. Except for the proposed algorithm, a weighted-sum MOO strategy, which integrates the objectives into one fitness value in each fitness evaluation as shown in (2.36), is applied. The weighting vector in these two MOOs is $\boldsymbol{\omega} = [0.25, 0.15, 0.6]^T$.

The difference between the proposed non-dominated sorting MOO and the weighted-sum MOO is where the weighted-sum (2.36) is implemented. The non-dominated sorting MOO finds multiple solutions in one single simulation run [8] and implements (2.36) in decision making after the evolution of solutions, whereas the weighted-sum MOO implements (2.36) in fitness evaluation along with the evolution whenever a new individual is generated.

The crossover probability of each single-period path is $p_{\mathrm{c}}^{\mathrm{s}} = 0.5$. The probabilities of the moves in the mutation operator are $p_{\mathrm{rm}} = 0.2$, $p_{\mathrm{in}} = 0.4$ and $p_{\mathrm{rs}} = 0.4$. In constructing the chromosome, $T_{\mathrm{h}} = 1/6$ hours, which means that the wait time after the minimum downtime is an integer multiple of T_{h}. The truncated normal distribution with $m = 1.5$ is used to determine the number of wait time slots. While evaluating the *revisit-unrepeated* metric, the sliding window size is $T_{\mathrm{c}} = 4$ hours and the step length $\delta = 1$ hour. Hence, there are $(T_{\mathrm{m}} - T_{\mathrm{c}})/\delta + 1 = 21$ evaluation intervals. In this simulation, the population size is set to be 1024. The number of generations in each simulation is 150. The best weighted-sum fitness in each generation is plotted in Fig. 2.7. The results in the following simulations are based on 100 Monte Carlo runs.

It can be seen from Fig. 2.7 that the integrated fitness value reaches a plateau in each simulation with different MOO solution strategies and crossover/mutation probabilities, which shows the convergence of the proposed algorithm. Based on the convergence performance of the proposed algorithm, in the following simulations, the termination condition threshold is set to be 10^{-5}, N_t is set to be 10, and the crossover probability is 0.7.

The weighted-sum model for the objectives can be the other solution strategy for the proposed MOO. Because of avoiding the non-dominated sorting, the weighted-sum MOO strategy requires slightly less computation time in each evolution/generation than the non-dominated sorting MOO strategy. However, the weighted-sum model for MOO is deficient in attaining a diverse set of solutions [21]. The final solution by the weighted-sum MOO is greatly influenced by the selection of the weights in (2.36). If the user's preferences change, the EA using the weighted-sum MOO strategy is to

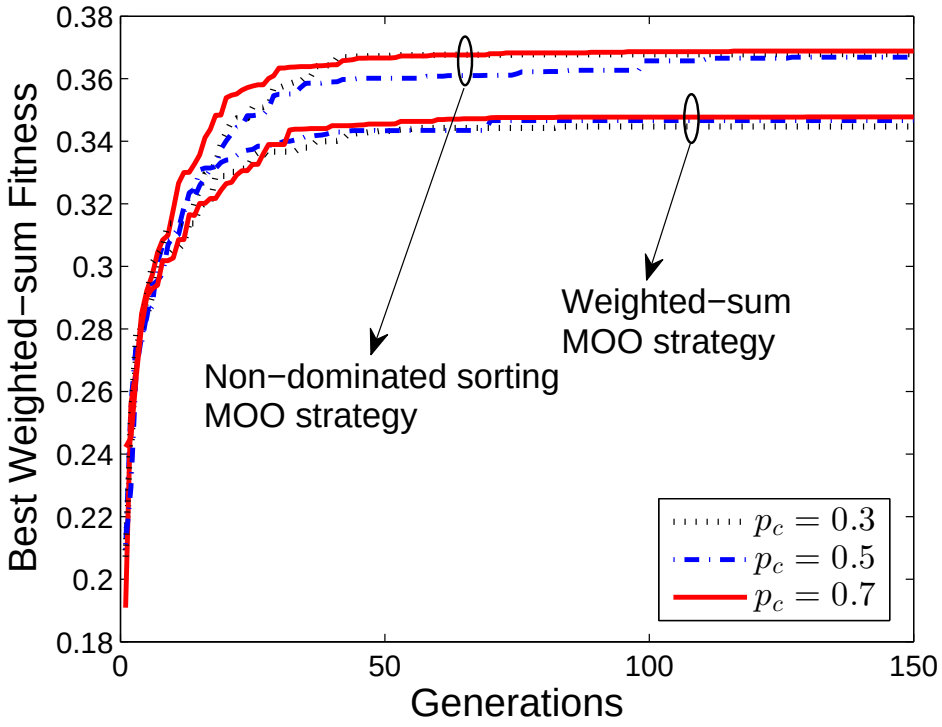


Figure 2.7: Best weighted-sum fitness: convergence of the proposed algorithm.

run again using the new weighting vector.

Our proposed non-dominated sorting MOO strategy enables the attainment of a diverse set of mission plans [8]. The final mission plan can be determined by the weighting vector in decision making, which saves computation time if the user's preferences change or multiple mission plans are needed.

2.6.3 Computation Time

The proposed solution procedure is done offline ahead of the mission. Time is mainly spent on two parts: initial population generation and multi-objective evolution. Single-period paths are planned by solving a series of OPs in the initial population generation. The OP is NP-hard (non-deterministic polynomial-time hard) [30] and these OPs are solved by heuristic methods [3], as described in Section 2.5.2. The number of OPs to solve depends on the topology of mission scenario, the capacity of aircraft and the selection of the time limits T_a^{lmt}. The NSGA-II framework is used to generate a diverse set of mission plans. The computational complexity of NSGA-II is $\mathcal{O}(MN^2)$, where N is the population size and M is the number of objectives [8].

This coverage path planning and scheduling problem is solved using MATLAB on a Core$^{\text{TM}}$ i7 2.5 GHz CPU with 16 GB RAM. In the above simulation, the population size is 1024, and the number of objectives is 3. The average computation time of initial population generation is 44 seconds. Using the non-dominated sorting MOO strategy, the average computational time of convergence to the final solution by the evolutionary process is 377 seconds. Using the weighted-sum MOO strategy, the average computational time of convergence to the final solution by the evolutionary process is 274 seconds. Note that the proposed algorithm is for offline path planning and scheduling, therefore the computation time is modest.

2.6.4 Objectives

In order to illustrate the advantage of using both *revisit-unrepeated* and *revisit-total* as objectives, $\max_P (f_1, f_2, f_3)$ is compared with $\max_P (f_1, f_3)$ and $\max_P (f_2, f_3)$. The values of f_1, f_2 and f_3 are evaluated as (2.9), (2.11) and (2.34) respectively, where the

Table 2.2: Results for the Problem with Different Objective Functions

Objective function	f_1	f_2
$\max_P\ (f_1, f_2, f_3)$	**41.4**	463.3
$\max_P\ (f_1, f_3)$	41.3	398.8
$\max_P\ (f_2, f_3)$	32.8	**465.7**

unit of f_3 is hour.

Feasible solutions ($f_3 = 0$) are generated in all cases, because revisit of each POI within 8 hours is not a strong constraint given the mission scenario. It can be seen from Table 2.2 that large values of f_1 and f_2 are attained by the proposed objectives $\max_P\ (f_1, f_2, f_3)$. However, when maximization of f_2 is not used as an objective, the value of f_2 is much smaller than that of $\max_P\ (f_1, f_2, f_3)$. Conversely, when maximization of f_1 is not specified as an objective, the largest f_2 is achieved, but f_1 is much smaller than that of $\max_P\ (f_1, f_2, f_3)$. Hence, the multi-objective function used in this paper guarantees prompt update of information from different POIs and a large number of covered cells (visited POIs), which in turn makes a persistent and effective surveillance mission plan.

2.6.5 Mutation Operator

The mutation operator is especially designed for this problem to enable area revisiting and take-off time scheduling. To illustrate the advantage of the custom mutation operator, the proposed EA is compared with the following two methods. Method-1 is an EA based on the proposed framework but without three custom mutation moves. In Method-2, which is based on the path planning method in [19], a single-period path is generated for each aircraft, and then this single-period path is copied to generate the

Table 2.3: Results for the Problem using Different Methods

Method	f_1	f_2
The propose EA	**41.4**	**463.3**
Method-1	36.9	365.5
Method-2	22	330

multi-period path with the minimum downtime between two successive single-period paths. Note that aircraft in the paths generated by Method-2 conducts a search in strips [19]; revisiting and take-off time scheduling are not enabled in Method-2.

Feasible solutions ($f_3 = 0$) are generated in these experiments. It can be seen from Table 2.3 that the proposed EA with custom mutation operator achieves the largest values of f_1 and f_2 comparing to the other two methods without problem-specific designs for revisiting and scheduling.

2.6.6 Revisit Violation Handling

Feasible solutions are generated when the requirement of revisit time is 8 hours. In this simulation, the mission scenario and configuration remain consistent with the previous ones, except the required revisit time changes to 4 hours. The proposed algorithm is compared with the following two methods. Method-3 is an EA based on the proposed framework but without *Reschedule* and without additional wait time at bases (aircraft land for the minimum downtime between successive flights). Method-4 is an EA based on the proposed framework but with the objectives $\max_P (f_1, f_2)$, in which the revisit constraint (2.20) is removed.

The sum of revisit violation time ($-f_3$) in the proposed EA is smaller than that in Method-3 mainly because the scheduling of take-off time by *Reschedule* enhances the temporal cooperation of aircraft so that the take-off time is adjusted to minimize $-f_3$

Table 2.4: Results for the Problem When the Required Revisit Time is 4 Hours

Method	f_1	f_2	$-f_3$
The propose EA	**42.5**	445.3	**26.3**
Method-3	40	441.8	37.4
Method-4	41.4	**472.7**	107.8

in the proposed algorithm. The comparison between the proposed EA and Method-3 illustrates the advantage of *Reschedule* designed in the mutation operator. Although the value of f_2 in Method-4 is the greatest among these three experiments, the revisit violation is much greater than that in the proposed EA. The proposed algorithm minimizes the sum of revisit violation time when there is no feasible solution w.r.t. constraint (2.20).

2.6.7 A Real-world Case Study

A more complex surveillance scenario that emerges from a real-world problem is set up, as shown in Fig. 2.8. Four water areas (circular areas in Fig. 2.8) are the AOIs. The radiuses of the AOIs are 55 km, 80 km, 110 km, and 55 km. Three aircraft are scheduled to perform a 48-hour mission. There is one type-1 aircraft and two type-2 aircraft. There is one base station (B1) for type-1 aircraft and two base stations (B2 and B3) for type-2 aircraft. Only one aircraft is assigned to one base station.

For type-1 aircraft, the maximum flight time is 10 hours, the minimum downtime is 2 hours and the maximum downtime is 3 hours. For type-2 aircraft, the maximum flight time is 5 hours, the minimum downtime is 1 hour and the maximum downtime is 1.5 hours. The speed of type-1 aircraft is 749 km/h and that of type-2 aircraft is 666 km/h. Each POI in an AOI must be revisited within 8 hours. Cells are circular areas with diameter 44 km. AOIs are segmented into 47 cells.

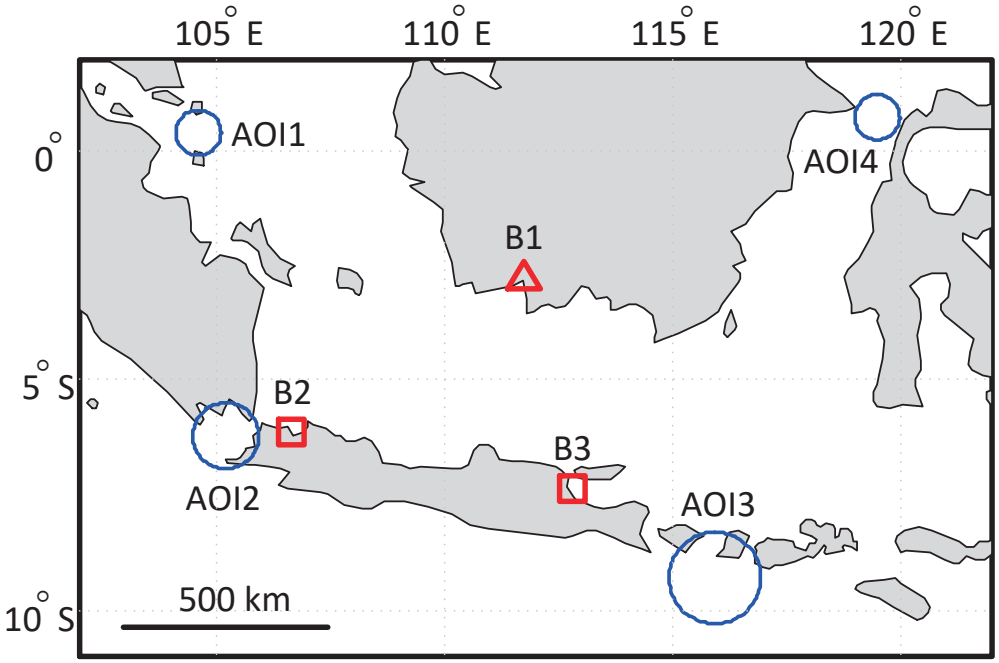


Figure 2.8: The second simulation scenario: locations of AOIs and base stations.

The parameters used in the proposed algorithm are the same as those in Section 2.6.2. The population size is 1024. Feasible solutions ($f_3 = 0$) are generated for this mission scenario. The values of f_1 and f_2 in the best mission plan are 29.7 and 899.2, respectively. The path planning and scheduling takes 1242 seconds on average, based on the computer hardware described in the first simulation scenario. Note that this computation is done offline ahead of the missions.

2.7 Conclusions

In this paper, a novel approach was described for the offline mission planning for multiple aerial vehicles to perform surveillance over disjoint areas. The formulation was

presented for this multi-period coverage path planning and temporal scheduling problem, in which repeated visit of the same area in a path was considered. This revisiting feature is common in surveillance applications, but distinguishes the formulations and solution techniques of those in the vehicle routing problem and its variations. Objective functions were developed to promptly update information collected by covering AOIs and to ensure persistent surveillance. A multi-objective evolutionary algorithm with a novel chromosome representation and custom genetic operators, which enables the revisiting of areas and the scheduling of take-off time, was proposed. The proposed algorithm was computationally efficient even in large scale problems and can be ported to a variety of surveillance coverage problems. Future research will address additional real-world constraints in airborne surveillance, for example, the selection and assignment of the base station when aircraft do not have to take off and land from/at the same base station, the management of sensor cueing along with platform path planning when sensor look direction can be adjusted.

2.8 Appendix A: No-fly Zone Avoidance

If there exist no-fly zones [9, 11] outside AOIs, they can be handled while generating single-period paths as follows. Vertices are added on the border or outside but close to the border of each no-fly zone. A penalty is defined for each edge in addition to the travel distance:

$$q_{ij} = \begin{cases} -M & \text{if the shortest path from vertex } i \text{ to vertex } j \text{ across} \\ & \text{no-fly zones} \\ 0 & \text{otherwise} \end{cases} \tag{2.38}$$

where M is a very large positive constant. The penalty for each edge is pre-computed based on whether the edge across no-fly zones or not. This penalty is included in the objective function of the single-period path planning:

$$\max_{x_{ij}} \quad \sum_{i \in S} \sum_{j \in V} \mathrm{sc}(i) x_{ij} + \sum_{i \in V} \sum_{j \in V} q_{ij} x_{ij} \tag{2.39}$$

Thus, aircraft can bypass no-fly zones in the single-period path. Once the single-period paths avoid no-fly zones, paths in the mission plan will avoid no-fly zones, because the genetic operators (Section 2.5.5) remove paths outside AOIs or add path segments inside AOIs, which will not cross no-fly zones.

Bibliography

[1] T. Bektas, "The multiple traveling salesman problem: An overview of formulations and solution procedures," *Omega*, vol. 34, no. 3, pp. 209–219, 2006.

[2] N. Bostel, P. Dejax, P. Guez, and F. Tricoire, "Multiperiod planning and routing on a rolling horizon for field force optimization logistics," in *The Vehicle Routing Problem: Latest Advances and New Challenges*, B. Golden, S. Raghavan, and E. Wasil, Eds. Boston, MA: Springer US, 2008, pp. 503–525.

[3] I.-M. Chao, B. L. Golden, and E. A. Wasil, "A fast and effective heuristic for the orienteering problem," *European Journal of Operational Research*, vol. 88, no. 3, pp. 475–489, 1996.

[4] H. Choset, "Coverage for robotics—A survey of recent results," *Annals of Mathematics and Artificial Intelligence*, vol. 31, no. 1–4, pp. 113–126, 2001.

[5] M. R. Chummun, Y. Bar-Shalom, and T. Kirubarajan, "Adaptive early-detection ML-PDA estimator for LO targets with EO sensors," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 2, pp. 694–707, Apr. 2002.

[6] B. Crevier, J.-F. Cordeau, and G. Laporte, "The multi-depot vehicle routing

problem with inter-depot routes," *European Journal of Operational Research*, vol. 176, no. 2, pp. 756–773, 2007.

[7] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: Solving problems with box constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, Aug. 2014.

[8] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[9] K. Doğançay, "UAV path planning for passive emitter localization," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 2, pp. 1150–1166, Apr. 2012.

[10] M. Drexl, "Rich vehicle routing in theory and practice," *Logistics Research*, vol. 5, no. 1–2, pp. 47–63, 2012.

[11] H. Ergezer and K. Leblebicioglu, "Path planning for UAVs for maximum information collection," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 1, pp. 502–520, Jan 2013.

[12] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1258–1276, 2013.

[13] F. Gavilan, R. Vazquez, and E. F. Camacho, "An iterative model predictive control algorithm for UAV guidance," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 3, pp. 2406–2419, July 2015.

[14] D. E. Goldberg, B. Korb, and K. Deb, "Messy genetic algorithms: Motivation, analysis, and first results," *Complex Systems*, vol. 3, pp. 493–530, 1989.

[15] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence.* MIT press, 1992.

[16] J. Hu, L. Xie, J. Xu, and Z. Xu, "Multi-agent cooperative target search," *Sensors*, vol. 14, no. 6, pp. 9408–9428, 2014.

[17] H. Jain and K. Deb, "An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, Part II: Handling constraints and extending to an adaptive approach," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 602–622, Aug. 2014.

[18] P. J. Jones, "Cooperative area surveillance strategies using multiple unmanned systems," Ph.D. dissertation, School of Elect. & Comp. Eng., Georgia Tech, Atlanta, GA, 2009.

[19] O. Karasakal, "Minisum and maximin aerial surveillance over disjoint rectangles," *TOP*, vol. 24, no. 3, pp. 705–724, 2016.

[20] S. M. LaValle, *Planning Algorithms.* New York, NY, USA: Cambridge University Press, 2006.

[21] R. T. Marler and J. S. Arora, "The weighted sum method for multi-objective optimization: New insights," *Structural and Multidisciplinary Optimization*, vol. 41, no. 6, pp. 853–862, 2010.

[22] C. E. Miller, A. W. Tucker, and R. A. Zemlin, "Integer programming formulation of traveling salesman problems," *Journal of the ACM*, vol. 7, no. 4, pp. 326–329, Oct. 1960.

[23] A. Moitra, R. Mattheyses, V. DiDomizio, L. Hoebel, R. Szczerba, and B. Yamrom, "Multivehicle reconnaissance route and sensor planning," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 3, pp. 799–812, July 2003.

[24] J. R. Montoya-Torres, J. L. Franco, S. N. Isaza, H. F. Jiménez, and N. Herazo-Padilla, "A literature review on the vehicle routing problem with multiple depots," *Computers & Industrial Engineering*, vol. 79, pp. 115–129, 2015.

[25] R. Pitre, X. Li, and R. Delbalzo, "UAV route planning for joint search and track missions—An information-value approach," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 3, pp. 2551–2565, July 2012.

[26] S. Ragi and E. K. P. Chong, "UAV path planning in a dynamic environment via partially observable Markov decision process," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 4, pp. 2397–2412, Oct. 2013.

[27] T. Sawik, "A note on the Miller-Tucker-Zemlin model for the asymmetric traveling salesman problem," *Bulletin of the Polish Academy of Sciences Technical Sciences*, vol. 64, no. 3, pp. 517–520, 2016.

[28] B. D. Song, J. Kim, and J. R. Morrison, "Rolling horizon path planning of an

autonomous system of UAVs for persistent cooperative service: MILP formulation and efficient heuristics," *Journal of Intelligent & Robotic Systems*, vol. 84, no. 1, pp. 241–258, 2016.

[29] K. Tan, T. Lee, K. Ou, and L. Lee, "A messy genetic algorithm for the vehicle routing problem with time window constraints," *Proceedings of the 2001 Congress on Evolutionary Computation*, vol. 1, pp. 679–686, 2002.

[30] H. Tang and E. Miller-Hooks, "A tabu search heuristic for the team orienteering problem," *Computers & Operations Research*, vol. 32, no. 6, pp. 1379–1407, June 2005.

[31] P. Toth and D. Vigo, Eds., *The Vehicle Routing Problem*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2001.

[32] F. Tricoire, M. Romauch, K. F. Doerner, and R. F. Hartl, "Heuristics for the multi-period orienteering problem with multiple time windows," *Computers & Operations Research*, vol. 37, no. 2, pp. 351–367, 2010.

[33] G.-H. Tzeng and J.-J. Huang, *Multiple Attribute Decision Making: Methods and Applications*. Boca Raton, FL: CRC press, Taylor & Francis Group, 2011.

[34] P. Vansteenwegen, W. Souffriau, and D. V. Oudheusden, "The orienteering problem: A survey," *European Journal of Operational Research*, vol. 209, no. 1, pp. 1–10, 2011.

[35] T. Vidal, T. G. Crainic, M. Gendreau, N. Lahrichi, and W. Rei, "A hybrid genetic algorithm for multi-depot and periodic vehicle routing problems," *Operations Research*, vol. 60, no. 3, pp. 611–624, May 2012.

The following chapter is a reproduction of a peer-reviewed article submitted to the IEEE:

Yinghui Wang, R. Tharmarasa, T. Kirubarajan, Bumsoo Kim, Rahim Jassemi-Zargani Multisensor Joint Path Planning and Scanning for Mobile 3D Search-and-Track, Submitted to *IEEE Transactions on Aerospace and Electronic Systems*, June 2018.

Chapter 3

Multisensor Joint Path Planning and Scanning for Mobile 3D Search-and-Track

3.1 Abstract

In this paper we propose a joint path planning and scanning (JPPS) solution for a fleet of unmanned aerial vehicles (UAVs) equipped with a set of sensors and cooperatively carrying out search-and-track (SAT) in a three-dimensional (3D) space with a number of targets. The 3D search space can be a mobile object with a large volume, which is modeled as a cuboid and tracked, followed and scanned by the UAVs. This search space is divided into a number of small cubes and it is assumed that the sensor on each UAV scans a fixed number of such cubes in each period of its operation. To provide insights for 3D path planning and scanning, the optimal 3D sensor geometry for target localization is analyzed with the objective to minimize the estimation uncertainty

under constraints on sensor altitude, sensor-to-sensor and sensor-to-target distances for active and passive sensors, respectively. For the 3D SAT mission, the objective function incorporates information from the search space centroid, detected targets as well as possible information from the yet-to-be-detected targets, which are included in the form of possible targets in the cubes. It is required that sensors cooperatively scan cubes and track detected targets from different angles to decrease the miss detection probability and enhance the tracking performance. In computing the information for a particular sensor-target geometry, the target detection probabilities, UAV kinematic, communication connectivity, collision and target fire avoidance are jointly considered. A weighted-sum multiobjective mixed-integer nonlinear optimization is formulated and a solution technique based on genetic algorithm is applied to determine the scan regions and path selection simultaneously. In the simulations, the proposed algorithm is compared with two independent path planning and scanning solution methods, which use pre-defined 3D search patterns for scanning and solve a nonlinear programming problem for path planning. The proposed JPPS algorithm outperforms the decoupled methods in terms of the overall SAT performance.

3.2 Introduction

In recent years, UAVs have been widely employed in civil and military applications such as search and rescue, environmental monitoring and battlefield surveillance. With the advent of affordable UAV systems, a cooperative group of small UAVs becomes a major contributor to intelligent and effective data collection. Moreover, modern sensor suites mounted on UAVs enable salient features of the target to be closely monitored and inspected. For example, it is possible to use a fleet of UAVs

to address threats associated with cargo-container-borne dangerous materials, which moves the scanning, detection and intervention away from the Customs area at a coastal port city to water areas far from the shore [10].

Considerable effort has been dedicated to UAV path planning for target searching. Pattern templates such as a ladder pattern are commonly used in search missions [1], in which the environment is always a bounded region either with or without obstacles or forbidden regions inside. The surveillance region is often divided into cells and each cell is associated with a probability or confidence of target existence, which constitutes a probability map for the region [4]. A vision-based cooperative search path planning algorithm, which optimizes the collective coverage area and the detection performance using a probability map update model, is proposed in [16].

The objective function in optimal sensor placement can be applied to optimal path planning and the analytical solutions for optimal sensor-to-target geometries reveal important insights into path planning for target tracking [22, 35]. The optimal geometries for a two-dimensional (2D) time-of-arrival (TOA) localization configuration is analyzed and the UAV path planning, which is based on minimizing the area of estimation confidence region, is simulated in [22]. Similarly, based on the Fisher information matrix (FIM), the UAV path planning for passive emitter localization in a 2D environment is presented in [9], where angle-of-arrival (AOA), time-difference-of-arrival (TDOA) and scan-based localization are considered.

In SAT missions, the goal is not only to find or detect targets, but also to track them for the remainder of the mission. A decentralized cooperative control algorithm, which maximizes the information gathered from an environment and decides the next area to scan by predicting how much information can be gained by searching that area,

is proposed in [27] for multiple cooperative UAVs to track detected targets as well as to search for the undetected ones. An information-based objective function, which integrates the conflicting objectives of target detection, target tracking and vehicle survivability into a single scalar index, is presented for UAV path optimization in [23].

Although optimal path planning has received considerable attention for detection, localization and tracking, existing algorithms are still limited regarding the assumptions about the environment, search space and sensing capability. A number of optimal sensor placement and path planning algorithms are developed in 2D space without extending to 3D scenarios [22, 23, 27]. Although 3D sensing model and 3D UAV dynamics are considered in [16], the search area is a 2D plane. Recently, a comprehensive analysis of optimal sensor-to-target geometries is provided in [35] for 3D AOA target localization with no restriction on the number of sensors. However, it fails to consider that the sensor placement region is always constrained in real applications. For example, in an airborne sensor placement scenario, the target is on the ground and the sensor altitude has an upper bound smaller than the sensor-to-target safety distance. Therefore, the sensor cannot be placed right above the target (i.e., elevation at $\pi/2$). Hence, it is worth studying the optimal sensor geometry and path planning under constraints on sensor altitude, sensor-to-sensor distance and sensor-to-target distance in 3D.

In SAT missions, the region of interest is usually assumed to be known and stationary [16, 27]. However, there are cases that the search space itself can be viewed as a mobile target, e.g., the search space may be a container ship and targets to be searched and tracked are concealed onboard [10]. Note that the path planning for

SAT missions in a mobile search space has not been investigated in the literature to the best of our knowledge. Search-and-track of a mobile space leads to the problem of tracking the search space, determining regions to scan and tracking detected targets.

In this paper, we consider an SAT scenario where the 3D mobile search space is to be tracked, followed and scanned, with possible targets within this space are to be searched and detected targets tracked. Two sensors on each UAV are used: one sensor (referred to as S1) tracks the surveillance space centroid, while the other sensor (referred to as S2) searches for possible targets and tracks detected targets within the search space. To inspect and monitor the surveillance space from different angles, as well as to avoid missed detections and to enhance tracking performance, at least two UAVs with sensor S2 are required to scan the same cubes or track detected targets cooperatively. The UAVs exchange the information about scanned cubes, obtained tracks and their current kinematic states.

The optimal 3D sensor placement for target localization by cooperative UAVs is analyzed to provide useful insights for path planning and scanning in this 3D SAT missions. Both active sensors with range, azimuth and elevation measurements (for short, active sensors) and passive sensors with AOA (i.e., azimuth and elevation) measurements (for short, passive sensors) are investigated [5]. These two sensor modes can be used for both sensors S1 and S2.

The objective function of optimal sensor placement minimizes the trace of Cramér-Rao lower bound (CRLB), which is defined as the inverse of the FIM and provides a mean square error bound on the performance of any unbiased estimator of an unknown parameter vector [2, 12, 35]. The bound is referred to as the posterior CRLB (PCRLB) if this parameter vector is stochastic [12], which is used for the optimal

path planning in this paper. The PCRLB gives a measure of the achievable optimum performance and this bound can be calculated predictively [13, 29]. Constraints, such as the maximum and minimum sensor altitude, minimum sensor-to-target distance for safety issues, maximum sensor-to-sensor distance for communication connectivity, and minimum sensor-to-sensor distance for collision avoidance, are considered for the optimal sensor configuration. Meanwhile, the UAV dynamics is added to the constraints in path planning.

When the optimal 3D sensor geometries for target localization under the above-mentioned constraints are known, the target tracking performance by UAVs can be roughly known by looking at the relative geometry relationship among the UAVs and the target. The optimal 3D sensor geometries provide an evaluation benchmark for online path planning: the UAVs should form a geometry similar to the optimal one while tracking the target. Additionally, with the optimal 3D sensor geometries, the following can be achieved in the SAT missions. When the target (e.g., the surveillance space centroid) state estimation is known, the best initial locations for UAVs, for example, the base stations where the UAVs take off, can be selected based on the optimal sensor geometry and the predicted target position. When the UAVs are approaching the surveillance space centroid but not close enough to inspect the space, they can fly to the positions which form the optimal sensor geometry with respect to (w.r.t.) the surveillance space centroid without applying a step-by-step online path planning but a simple geometry computation based on the predicted target position and the known optimal sensor-to-sensor and sensor-to-target geometry. Therefore, the optimal 3D sensor geometries for target localization provide useful insights for path planning and scanning in this 3D SAT missions.

For the SAT mission, a JPPS strategy based on genetic algorithm (GA) [14] is developed to solve a receding horizon optimization problem [30, 23]. This proposed algorithm selects paths and determines cubes to scan for all sensors by maximizing the predictive information in the Fisher sense from 1) the surveillance space centroid, 2) detected targets, and 3) the yet-to-be-detected targets, which are included in the form of possible targets in the cubes [27]. The above three objectives are integrated into one single metric by a weighted-sum model [33].

The proposed JPPS is compared with an approach that takes the decisions on path selection and cubes to scan independently. Three-dimensional search patterns, which consider the coverage induced by the sensor look direction and the coverage induced by the sensing platform movement jointly, are used.

This paper makes the following contributions:

1) A comprehensive analysis of optimal 3D sensor configuration and path planning for target localization by cooperative sensors from different angles with restrictions on the sensor altitude, sensor-to-sensor and sensor-to-target distance is provided for active and passive sensors. Different constraints on sensor placement are given based on real-world applications. The optimization problem is formulated to minimize the trace of the inverse of FIM and is solved by interior point methods. Extensive examples of the optimal sensor geometry and path planning for 3D localization are shown in the simulation studies.

2) A weighted-sum multiobjective optimization formulation for 3D mobile surveillance space search-and-track, which integrates the tracking performance of the search space, detected targets within this space as well as the yet-to-be-detected targets by different sensors, is presented. Note that the search space is a 3D volume and mobile,

which distinguishes our problem and the proposed JPPS strategy from those approaches for 2D search-and-track within the fixed surveillance region. Cubes to scan and path selection for the future time steps are determined jointly by the JPPS. Meanwhile, cooperative 3D path planning with constraints is considered in our approach. A JPPS solution technique based on genetic algorithm, which requires modest computational resources and outperforms the independent path planning and scanning methods, is developed.

The remainder of the paper is structured as follows. The optimal 3D sensor geometries with constraints are formulated for active and passive modes in Section 3.3. Section 3.4 presents the optimization formulation for this cooperative SAT problem in a 3D mobile surveillance space. The solution technique is developed in Section 3.5. Optimal sensor configurations and illustrative numerical examples are presented for target localization in Section 3.6. Simulation results on path planning and scanning are given for the proposed JPPS in Section 3.7. The conclusions are discussed in Section 3.8.

3.3 Optimal Sensor Placement with Constraints for 3D Localization

Consider the 3D localization problem where the target at a fixed location $\boldsymbol{p} = [x, y, z]^\intercal$ is to be located by N_s sensors at $\boldsymbol{s}_i = [x_i, y_i, z_i]^\intercal$, $i = 1, 2, \ldots, N_s$, where $^\intercal$ denotes matrix transpose. With no loss of generality, the target is assumed to be at the origin of the Cartesian coordinate system, i.e., $\boldsymbol{p} = [0, 0, 0]^\intercal$. Sensors in passive and active modes are investigated.

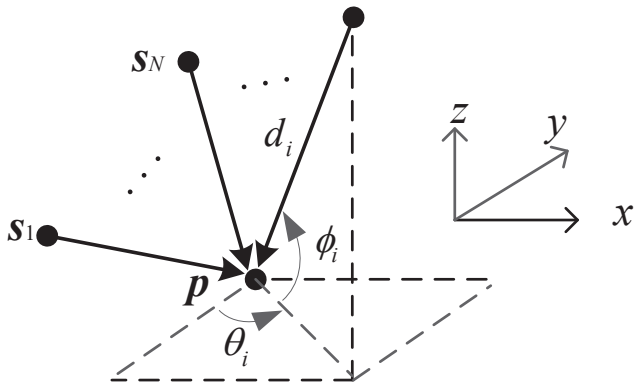


Figure 3.1: 3D localization using range, azimuth and elevation.

3.3.1 Passive Sensors with AOA Measurements

In 3D AOA localization, the target location $\boldsymbol{p}$ is estimated from azimuth and elevation angular measurements $\{\theta_i, \phi_i\}$ [35, 9]. These measurements are assumed to be corrupted by additive white Gaussian noise with zero mean

$$\tilde{\theta}_i = \theta_i + n_i,\ n_i \sim \mathcal{N}(0, \sigma_i^2), \tag{3.1}$$

$$\tilde{\phi}_i = \phi_i + e_i,\ e_i \sim \mathcal{N}(0, \gamma_i^2), \tag{3.2}$$

where $-\pi < \tilde{\theta}_i \leq \pi$, $-\pi/2 < \tilde{\phi}_i < \pi/2$, σ_i^2 and γ_i^2 are sensor-dependent noise variances.

The azimuth and elevation angles of the target collected by sensor i are

$$\theta_i = \tan^{-1} \frac{y - y_i}{x - x_i} \tag{3.3}$$

and

$$\phi_i = \sin^{-1} \frac{z - z_i}{\|\boldsymbol{p} - \boldsymbol{s}_i\|}, \tag{3.4}$$

respectively, where $\tan^{-1}$ is the 4-quadrant arctangent, $\|\cdot\|$ denotes the Euclidean norm, and $d_i = \|\boldsymbol{p} - \boldsymbol{s}_i\|$ is the sensor-to-target distance.

Under the Gaussian noise assumption, the FIM for the 3D AOA localization is [35]

$$\boldsymbol{\Phi}_{\mathrm{P}} = \boldsymbol{J}_{\mathrm{P}}^{T} \boldsymbol{\Sigma}_{\mathrm{P}}^{-1} \boldsymbol{J}_{\mathrm{P}} \tag{3.5a}$$

$$= \underbrace{\sum_{i=1}^{N_s} \frac{1}{d_i^2 \sigma_i^2 \cos^2 \phi_i} \boldsymbol{u}_i \boldsymbol{u}_i^{\intercal}}_{\boldsymbol{\Phi}_1} + \underbrace{\sum_{i=1}^{N_s} \frac{1}{d_i^2 \gamma^2} \boldsymbol{v}_i \boldsymbol{v}_i^{\intercal}}_{\boldsymbol{\Phi}_2}, \tag{3.5b}$$

where $\boldsymbol{J}_{\mathrm{P}}$ is the Jacobian of AOA measurement errors evaluated at the true target location, $\boldsymbol{\Sigma}_{\mathrm{P}}$ is the AOA sensor measurement covariance matrix, $\boldsymbol{u}_i$ is the unit vector orthogonal to the 2D azimuth vector, $\boldsymbol{v}_i$ is the unit vector orthogonal to the 3D range vector

$$\boldsymbol{u}_i = \begin{bmatrix} -\sin\theta_i \\ \cos\theta_i \\ 0 \end{bmatrix}, \ \boldsymbol{v}_i = \begin{bmatrix} -\sin\phi_i \cos\theta_i \\ -\sin\phi_i \sin\theta_i \\ \cos\phi_i \end{bmatrix}. \tag{3.6}$$

In (3.5), the matrix $\boldsymbol{\Phi}_1$ represents the azimuth angle FIM as it corresponds to the 2D projection of the localization problem onto the xy-plane

$$\boldsymbol{\Phi}_1 = \sum_{i=1}^{N_s} \frac{1}{d_i^2 \sigma_i^2 \cos^2 \phi_i} \begin{bmatrix} \sin^2\theta_i & -\frac{1}{2}\sin 2\theta_i & 0 \\ -\frac{1}{2}\sin 2\theta_i & \cos^2\theta_i & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{3.7}$$

Note that a constraint that $\phi_i \neq \pm 90^\circ$ comes with (3.7). The azimuth angle is not available for a sensor right above or under the target. The matrix $\boldsymbol{\Phi}_2$ in (3.5) is the

elevation angle FIM, which can be expanded as

$$\boldsymbol{\Phi}_2 = \sum_{i=1}^{N_s} \frac{1}{d_i^2 \gamma_i^2} \begin{bmatrix} \sin^2\phi_i \cos^2\theta_i & \frac{1}{2}\sin^2\phi_i \sin 2\theta_i & -\frac{1}{2}\sin 2\phi_i \cos\theta_i \\ \frac{1}{2}\sin^2\phi_i \sin 2\theta_i & \sin^2\theta_i \sin^2\theta_i & -\frac{1}{2}\sin 2\phi_i \sin\theta_i \\ -\frac{1}{2}\sin 2\phi_i \cos\theta_i & -\frac{1}{2}\sin 2\phi_i \sin\theta_i & \cos^2\phi_i \end{bmatrix}. \tag{3.8}$$

3.3.2 Active Sensors with Range and Angular Measurements

In addition to AOA measurements as in (3.3) and (3.4), sensor-to-target distance is observed for sensors in active mode

$$d_i = \|\boldsymbol{p} - \boldsymbol{s}_i\|. \tag{3.9}$$

The range measurement is assumed to be corrupted by additive white Gaussian noise with zero mean

$$\tilde{d}_i = d_i + m_i,\ m_i \sim \mathcal{N}(0, \eta_i^2), \tag{3.10}$$

where η_i^2 is the sensor-dependent noise variance.

Under the Gaussian noise assumption, the FIM for the 3D target localization problem with measurements $\{d_i, \theta_i, \phi_i\}$ is

$$\boldsymbol{\Phi}_{\mathrm{A}} = \boldsymbol{J}_{\mathrm{A}}^T \boldsymbol{\Sigma}_{\mathrm{A}}^{-1} \boldsymbol{J}_{\mathrm{A}} \tag{3.11a}$$

$$= \boldsymbol{\Phi}_1 + \boldsymbol{\Phi}_2 + \underbrace{\sum_{i=1}^{N} \frac{1}{\eta_i^2} \boldsymbol{q}_i \boldsymbol{q}_i^{\intercal}}_{\boldsymbol{\Phi}_3}, \tag{3.11b}$$

where $\boldsymbol{J}_{\mathrm{A}}$ is the Jacobian of range, azimuth and elevation measurement errors evaluated at the true target location, $\boldsymbol{\Sigma}_{\mathrm{A}}$ is the sensor measurement covariance matrix,

and $\boldsymbol{q}_i$ is the 3D range vector

$$\boldsymbol{q}_i = \begin{bmatrix} \cos\phi_i \cos\theta_i \\ \cos\phi_i \sin\theta_i \\ \sin\phi_i \end{bmatrix}. \tag{3.12}$$

The matrix $\boldsymbol{\Phi}_3$ is the range FIM, which can be expanded as

$$\boldsymbol{\Phi}_3 = \sum_{i=1}^{N_s} \frac{1}{\eta_i^2} \begin{bmatrix} \cos^2\phi_i \cos^2\theta_i & \frac{1}{2}\cos^2\phi_i \sin 2\theta_i & \frac{1}{2}\sin 2\phi_i \cos\theta_i \\ \frac{1}{2}\cos^2\phi_i \sin 2\theta_i & \cos^2\theta_i \sin^2\theta_i & \frac{1}{2}\sin 2\phi_i \sin\theta_i \\ \frac{1}{2}\sin 2\phi_i \cos\theta_i & \frac{1}{2}\sin 2\phi_i \sin\theta_i & \sin^2\phi_i \end{bmatrix}. \tag{3.13}$$

3.3.3 Objective Function for Optimal Sensor Placement

In this paper, minimizing the trace of the inverse of FIM is used as the objective function for localization [35, 9]. It is assumed that sensors on different UAVs have the same working mode and parameters. The FIM is denoted as $\boldsymbol{\Phi}$, which can be $\boldsymbol{\Phi}_{\mathrm{P}}$ in (3.5) or $\boldsymbol{\Phi}_{\mathrm{A}}$ in (3.11), respectively. Minimizing the objective function is given by

$$\min \mathrm{tr}(\boldsymbol{\Phi}^{-1}). \tag{3.14}$$

Note that the objective can also be expressed as maximizing the trace of $\boldsymbol{\Phi}$, i.e., $\max \mathrm{tr}(\boldsymbol{\Phi})$ [23]. Here, minimization is used to be consistent with most (constrained) nonlinear programming solvers [31, 32].

The optimal sensor position for 3D AOA target localization has been researched in [35] with no restriction on the number of sensors. The optimal sensor position is determined by reducing $\boldsymbol{\Phi}_{\mathrm{P}}$ to a diagonal matrix: both $\boldsymbol{\Phi}_1$ and $\boldsymbol{\Phi}_2$ are diagonal [35].

From (3.7), (3.8) and (3.13), we can see that, when $\mathbf{\Phi}_1$ and $\mathbf{\Phi}_2$ are diagonal, $\mathbf{\Phi}_3$ is also diagonal. Therefore, given the sensor-to-target distance d_i and noise standard deviations, the angle configurations in [35] can be applied to active sensors with range and angular measurements. However, the constraints on sensor placement region, sensor-to-target distance and sensor-to-sensor distance are not included in [35], which are necessary for real applications. In the following, constraints for sensor placement are explicitly discussed.

3.3.4 Constraints for Optimal Sensor Placement

Restricted by the motion capability of the sensing platform, the sensor is constrained by a minimum altitude $h_i^{\min}$ and a maximum altitude $h_i^{\max}$.

$$h_i^{\min} \leq z_i \leq h_i^{\max},\ i = 1, 2, ..., N_s. \tag{3.15}$$

In some cases, under the constraint of (3.15), all sensors or a subset of them must be at the same altitude.

$$z_i = z_j,\ i, j \in \{1, 2, ..., N_s\},\ i \neq j. \tag{3.16}$$

Constraint (3.16) is a special case of (3.15) when $h_i^{\min} = h_i^{\max} = h_j^{\min} = h_j^{\max}$.

A minimum sensor-to-target distance $d_{\text{s-p}}$ is required for safety reasons, for example, to avoid possible fire from the target.

$$d_i \geq d_{\text{s-p}},\ i = 1, 2, ..., N_s. \tag{3.17}$$

A minimum sensor-to-sensor distance $d_{\text{s-s}}^{\min}$ can be placed for collision avoidance.

$$\|\boldsymbol{s}_i - \boldsymbol{s}_j\| \geq d_{\text{s-s}}^{\min},\ i,j \in \{1,2,...,N_s\},\ i \neq j. \tag{3.18}$$

A maximum sensor-to-sensor distance $d_{\text{s-s}}^{\max}$ is enforced for communication connectivity.

$$\|\boldsymbol{s}_i - \boldsymbol{s}_j\| \leq d_{\text{s-s}}^{\max},\ i,j \in \{1,2,...,N_s\},\ i \neq j. \tag{3.19}$$

The above constraints can be used alone or in combination. The problem of (3.14) under the constraints from (3.15) to (3.19) is a constrained nonlinear optimization, which can be solved by interior point methods [32]. There are many solvers available for constrained nonlinear optimization [7, 31, 32]. Methods to solve optimal sensor placement is not the focus of this paper. The optimal sensor configurations and illustrative numerical examples are provided in Section 3.6.

3.4 Path Planning and Scanning for Mobile 3D SAT

3.4.1 Problem Description

The surveillance space is a large 3D volume with known dimensional information, which is modeled as a cuboid with a fixed length, width and height. The search space can be stationary or mobile, with its centroid being tracked by sensor S1 and the entire space being scanned by sensor S2. No obstacle is assumed inside or out of the search space. The search space is divided into a number of small cubes with a fixed

length, as shown in Fig. 3.2.

It is assumed that the sensor S2 on each UAV can scan M such small cubes around the predicted positions of possible targets in each period of its operation by changing the look direction [28, 33]. The UAVs have already formed into groups before the SAT mission and the UAVs in the same group scan the same cubes and track the same detected targets using sensor S2. The number of UAVs is N_s, the number of groups is N_g and the number of UAVs in the gth group is N_s^g, where $\sum_{g=1}^{N_g} N_s^g = N_s$. There is an unknown number of targets within the search space, which can be stationary or mobile w.r.t. the coordination of the search space. Undetected targets are to be searched and detected targets are to be tracked. The SAT mission ends when the surveillance space has been scanned for N_{end} times, $N_{\text{end}} \geq 1$.

UAVs are required to efficiently and cooperatively gather information from the surveillance space centroid, detected targets as well as possible information from the yet-to-be-detected targets, which are included in the form of possible targets in the cubes. The information obtained from sensors S1 and S2 is maximized as in [27, 23], where the trace of the PCRLB is evaluated [29]. The centroid of each cube, whose location can be computed based on the dimensions of the search space and the estimated location of search space centroid, is used to guide sensor S2's look direction on each UAV. Paths are selected and scan decisions are made to achieve the optimal SAT performance given the estimated states of search space centroid, detected targets, unscanned cubes, and the states of the UAVs.

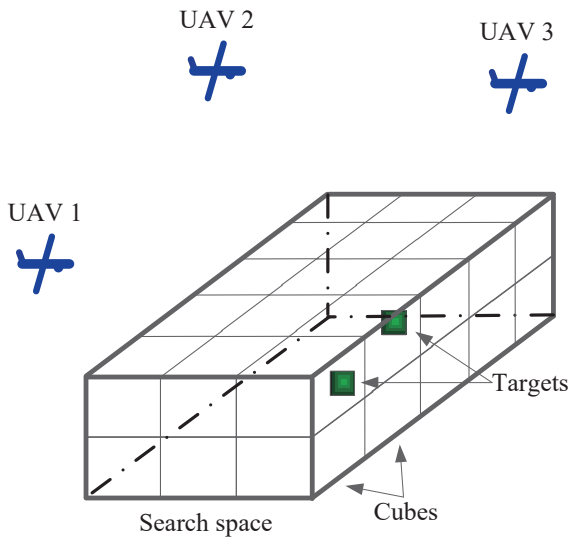


Figure 3.2: An illustration of the 3D search space, cubes and detected targets for SAT by three UAVs.

3.4.2 UAV Kinematic Model

The UAV path planning is constrained by the UAV kinematic capabilities. For example, in the horizontal plane, a fan-shaped flyable area can be formed for a UAV due to the constraints on its turning capability [17]. Different UAV kinematic models can be found in UAV path planning literature: in [25], it is assumed that UAVs can move in any direction with unit velocity; in [16], the UAV motion model only deals with the waypoints of agents at discrete-time steps and the true dynamics of UAVs are not discussed; in [17], UAVs move along continuous trajectories with constant speed and constraints on turning; in [24], the altitude of a UAV is a constant and the control variables are the forward acceleration and the bank angle. Note that the above UAV kinematic models only consider motion in the 2D plane.

In this paper, the 3D discrete time airplane model from [11] is used for UAVs. The UAV airspeed is assumed to be a constant V. The path planning decision variables

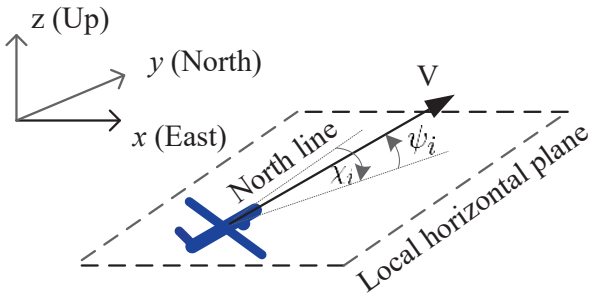


Figure 3.3: UAV kinematic model.

for the ith UAV at time k are the bank angle $\vartheta_i(k)$ and the flight path angle $\psi_i(k)$. The heading angle of the ith UAV at time k is expressed as $\chi_i(k)$ and the location is $\boldsymbol{s}_i(k) = [x_i(k), y_i(k), z_i(k)]^\intercal$. Under the assumption of constant V, $\vartheta_i(k)$ and $\psi_i(k)$ during the time interval from time k to $k+1$, the heading angle at time $k+1$ is

$$\chi_i(k+1) = \frac{gT\tan\left(\vartheta_i(k)\right)}{V} + \chi_i(k), \tag{3.20}$$

where g is the acceleration due to gravity and T is the length of the time interval.

For formula compactness, let

$$\kappa_i(k) = \frac{gT\tan\left(\vartheta_i(k)\right)}{V}. \tag{3.21}$$

The UAV waypoint update is modelled as [11]

$$\begin{aligned}\boldsymbol{s}_i(k+1) = \boldsymbol{s}_i(k) + \\ \begin{bmatrix} \frac{VT}{\kappa_i(k)}\cos\left(\psi_i(k)\right)\left[\sin(\kappa_i(k)+\chi_i(k)) - \sin(\chi_i(k))\right] \\ \frac{VT}{\kappa_i(k)}\cos\left(\psi_i(k)\right)\left[\cos(\chi_i(k)) - \cos(\kappa_i(k)+\chi_i(k))\right] \\ VT\sin\left(\psi_i(k)\right) \end{bmatrix}.\end{aligned} \tag{3.22}$$

In this manner, the future state $[\boldsymbol{s}_i(k+1), \chi_i(k+1)]^\intercal$ of the ith UAV can be predicted based on its previous state and the present control inputs $[\vartheta_i(k), \psi_i(k)]^\intercal$.

The bank angle $\vartheta_i(k)$ and the flight path angle $\psi_i(k)$ satisfy the following constraints

$$|\vartheta_i(k)| \leq \Theta, \ \forall i \in \{1, 2, ..., N_s\}, \tag{3.23}$$

$$|\psi_i(k)| \leq \Psi, \ \forall i \in \{1, 2, ..., N_s\}, \tag{3.24}$$

where Θ is the maximum bank angle and Ψ is the maximum flight path angle. Note that the rate of turn and radius of turn are constrained given (3.23).

3.4.3 Objectives for Path Planning and Scanning

The computation of the kinematic information matrix follows [29]. The objective function is the combination of information gathered on three aspects: surveillance space centroid, detected targets and possible information from cubes, which may contain possible targets. The UAV path planning (geometries of UAVs) affects the tracking performance of the search space centroid; the path selection and the scan decision jointly affect the tracking performance of the detected targets and the information gathered from cubes.

Let o represent the search space centroid, $\boldsymbol{C}$ denote the set of cubes, and $\boldsymbol{D}(k)$ stand for the set of detected targets. It is assumed that the detected targets are widely-separated targets.

For the search space centroid o, the information by sensor S1 on the ith UAV at

time k can be expressed as [29]

$$\begin{aligned} \boldsymbol{\Phi}_{i,o}(k) &= \mathbb{E}\left[\boldsymbol{J}_{k,1}^{i}(o)^{T} Q_{k,1}^{o}(i) \boldsymbol{\Sigma}_{k,1}^{-1} \boldsymbol{J}_{k,1}^{i}(o)\right], \\ & i \in \{1, 2, \ldots, N_s\}, \end{aligned} \tag{3.25}$$

where $Q_{k,1}^{o}(i)$ is the Information Reduction Matrix (IRM) of sensor S1 on the ith UAV, $\boldsymbol{J}_{k,1}^{i}(o)$ is the Jacobian of measurement errors evaluated at the state of o by sensor S1, $\boldsymbol{\Sigma}_{k,1}$ is sensor S1's measurement covariance matrix, and $\mathbb{E}$ denotes the expectation operator.

Similarly, for a target $t \in \boldsymbol{D}(k)$, the information by sensor S2 on the ith UAV at time k can be expressed as

$$\begin{aligned} \boldsymbol{\Phi}_{i,t}(k) &= \mathbb{E}\left[\boldsymbol{J}_{k,2}^{i}(t)^{T} Q_{k,2}^{t}(i) \boldsymbol{\Sigma}_{k,2}^{-1} \boldsymbol{J}_{k,2}^{i}(t)\right], \\ & i \in \{1, 2, \ldots, N_s\}. \end{aligned} \tag{3.26}$$

where $Q_{k,2}^{t}(i)$ is the IRM of sensor S2 on the ith UAV, $\boldsymbol{J}_{k,2}^{i}(t)$ is the Jacobian of measurement errors evaluated at the state of t by sensor S2, and $\boldsymbol{\Sigma}_{k,2}$ is sensor S2's measurement covariance matrix.

For possible targets inside the τth cube, $\tau \in \boldsymbol{C}$, the information is

$$\begin{aligned} \boldsymbol{\Phi}_{i,\tau}(k) &= \mathbb{E}\left[\boldsymbol{J}_{k,2}^{i}(\tau)^{T} Q_{k,2}^{\tau}(i) \boldsymbol{\Sigma}_{k,2}^{-1} \boldsymbol{J}_{k,2}^{i}(\tau)\right], \\ & i \in \{1, 2, \ldots, N_s\}. \end{aligned} \tag{3.27}$$

where $Q_{k,2}^{\tau}(i)$ is the IRM of sensor S2 on the ith UAV, $\boldsymbol{J}_{k,2}^{i}(\tau)$ is the Jacobian of measurement errors evaluated at the location of cube τ by sensor S2. Formulations (3.25) to (3.27) can be used for passive or active sensor mode. For a detailed explanation of

IRM calculations, the reader is referred to [29].

The search space centroid o is always tracked by sensor S1. However, sensor S2 has a limited scan capability: scan decision is made at each UAV among all the cubes and detected targets. Define a scan decision variable $\alpha_{i,\tau}(k)$

$$\alpha_{i,\tau}(k) = \begin{cases} 1 & \text{if sensor S2 on UAV } i \text{ will scan cube } \tau \text{ at time } k+1 \\ 0 & \text{otherwise} \end{cases}, \quad (3.28)$$

where $\tau \in \boldsymbol{C}$ and $i \in \{1, 2, \ldots, N_s\}$. Similarly, define a scan decision variable $\beta_{i,t}(k)$

$$\beta_{i,t}(k) = \begin{cases} 1 & \text{if sensor S2 on UAV } i \text{ will scan detected target } t \text{ at time} \\ & k+1 \\ 0 & \text{otherwise} \end{cases}, \quad (3.29)$$

where $t \in \boldsymbol{D}(k)$ and $i \in \{1, 2, \ldots, N_s\}$.

Sensor S2 on UAV i is able to scan M cubes or detected targets each time, which is a constraint on scan decision vairables. That is,

$$\sum_{\tau \in \boldsymbol{C}} \alpha_{i,\tau}(k) + \sum_{t \in \boldsymbol{D}(k)} \beta_{i,t}(k) = M, \ \forall i \in \{1, 2, \ldots, N_s\}. \quad (3.30)$$

The UAVs in the same group scan the same cubes and detected targets cooperatively

$$\begin{aligned} &\alpha_{i,\tau}(k) = \alpha_{j,\tau}(k), \ \beta_{i,t}(k) = \beta_{j,t}(k), \\ &\forall i, j \in \{1, 2, \ldots, N_s^g\}, \ i \neq j, \ \forall g \in \{1, 2, \ldots, N_g\}. \end{aligned} \quad (3.31)$$

The total number of scanned cubes and detected targets in the kth step is

$$\sum_{i=1}^{N_s}\left(\sum_{\tau\in\boldsymbol{C}}\alpha_{i,\tau}(k)+\sum_{t\in\boldsymbol{D}(k)}\beta_{i,t}(k)\right)=MN_g. \tag{3.32}$$

An importance factor $\delta_\tau(k)$ is defined for each cube [23]. This importance factor is a design parameter that is necessary for scan decision, because all cubes are not equally important as the SAT mission evolves. The importance factor depends on whether the cube is recently scanned: for example, a cube may be less valuable if it has been scanned recently. The most recent time that the τth cube being scanned can be denoted as

$$k_\tau=\max\left\{k'|\alpha_{i,\tau}(k')==1,\ k'=1,2,...,k\right\}. \tag{3.33}$$

This importance factor $\delta_\tau(k)$ can be formulated as a function of $(k-k_\tau)$.

Similarly, an importance factor $\rho_t(k)$ is defined for each detected target. This importance factor depends on the uncertainty about the state of the detected target and/or the specific measurement from the detected target if additional classification sensors/methods are used on the UAV. For example, a target with a large uncertainty region may be more valuable; a target with unique features may be more important; an evasive target may be more important than a stationary one w.r.t. the coordinate of the search space [23].

The above importance factors are updated after searching and tracking the cubes and detected targets at each time step. Therefore, $\delta_\tau(k)$ and $\rho_t(k)$ are used for time $k+1$.

The predictive information from the search space centroid by sensor S1 at time k

is defined as [13]

$$G_o(k) = \mathbb{E}\left[\sum_{i=1}^{N_s} \mathrm{tr}(\tilde{\boldsymbol{\Phi}}_{i,o}(k+1))\right], \tag{3.34}$$

where $\tilde{\boldsymbol{\Phi}}_{i,o}(k+1)$ is the predicted information given the current states of UAVs and current state of the search space centroid.

The predictive information from cubes by sensor S2 at time k is defined as

$$G_c(k) = \mathbb{E}\left[\sum_{i=1}^{N_s}\sum_{\tau\in\boldsymbol{C}} \alpha_{i,\tau}(k)\delta_\tau(k)\mathrm{tr}(\tilde{\boldsymbol{\Phi}}_{i,\tau}(k+1))\right]. \tag{3.35}$$

The predictive information from detected targets by sensor S2 at time k is

$$G_d(k) = \mathbb{E}\left[\sum_{i=1}^{N_s}\sum_{t\in\boldsymbol{D}(k)} \beta_{i,t}(k)\rho_t(k)\mathrm{tr}(\tilde{\boldsymbol{\Phi}}_{i,t}(k+1))\right]. \tag{3.36}$$

The objective function for the JPPS at time k is defined as

$$G(k) = \boldsymbol{w}(k)^\intercal \begin{bmatrix} G_o(k) \\ G_c(k) \\ G_d(k) \end{bmatrix}, \tag{3.37}$$

where $\boldsymbol{w}(k)$ is the weighting vector at time k. Details about $\boldsymbol{w}(k)$ will be discussed at the end of this section.

At time k, based on the state of UAVs, the estimated search space centroid, the estimated states of detected targets, and the approximate locations of cubes, the path planning and scanning for the next time step $k+1$ is made

$$\arg_{\{\vartheta_i(k),\psi_i(k),\alpha_{i,\tau}(k),\beta_{i,t}(k)\}} \max G(k),$$

$$\forall\ i \in \{1, 2, \ldots, N_s\}\,,\ \tau \in \boldsymbol{C},\ t \in \boldsymbol{D}(k). \tag{3.38}$$

Note that here we use the maximization of the trace of the information matrix rather than minimizing the trace of information matrix inverse for consistency with our GA based solution technique, because the fitness function is always maximized in GA [30, 33]. The path planning is constrained by the UAV kinematic constraints (3.23) and (3.24). Additionally, UAV path planning can be constrained by one or more constraints from (3.15) to (3.19). The scan decision is constrained by (3.30) and (3.31).

To guarantee a feasible solution, constraints (3.15) to (3.19) are transformed into penalty functions and included in the objective function [15].

The penalty function of constraint (3.15) is expressed as

$$\begin{aligned}\mathcal{P}_1(k) = -\,\mathcal{U} \sum_{i=1}^{N_s} \big[&\max\left(h_i^{\min} - \tilde{z}_i(k+1), 0\right) \\ &+ \max\left(\tilde{z}_i(k+1) - h_i^{\max}, 0\right)\big]\,,\end{aligned} \tag{3.39}$$

where $\mathcal{U}$ is a very large positive constant and $\tilde{z}_i(k+1)$ is the predicted altitude for the ith UAV according to the path selection decision variables. Since constraint (3.16) is a special case of constraint (3.15), the penalty function of constraint (3.16) shares the same formula with that of (3.15).

The penalty function of constraint (3.17) is expressed as

$$\mathcal{P}_2(k) = -\mathcal{U} \sum_{i=1}^{N_s} \left[\max\left(d_{\text{s-p}} - \tilde{d}_i(k+1), 0\right)\right], \tag{3.40}$$

where $\tilde{d}_i(k+1)$ is the predicted sensor-to-target distance given the path selection

decision variables and the predicted target location.

The penalty function of constraint (3.18) is expressed as

$$\mathcal{P}_3(k) = -\mathcal{U} \sum_{i=1}^{N_s} \sum_{\substack{j=1 \\ i \neq j}}^{N_s} \left[\max \left(d_{\text{s-s}}^{\min} - \| \tilde{\boldsymbol{s}}_i(k+1) - \tilde{\boldsymbol{s}}_j(k+1) \|, 0 \right) \right], \tag{3.41}$$

where $\tilde{\boldsymbol{s}}_i(k+1)$ is the predicted location of UAV i at time $k+1$ given the path selection decision variables.

The penalty function of constraint (3.19) is expressed as

$$\mathcal{P}_4(k) = -\mathcal{U} \sum_{i=1}^{N_s} \sum_{\substack{j=1 \\ i \neq j}}^{N_s} \left[\max \left(\| \tilde{\boldsymbol{s}}_i(k+1) - \tilde{\boldsymbol{s}}_j(k+1) \| - d_{\text{s-s}}^{\max}, 0 \right) \right]. \tag{3.42}$$

Define a vector of binary variables $\boldsymbol{b} = [b_1, b_2, b_3, b_4]^\intercal$, which indicates whether a constraint from (3.15) to (3.19) is included in the objective function or not. For example, $\boldsymbol{b} = [1, 0, 0, 0]^\intercal$ means only the altitudes of UAVs are constrained.

After adding the penalty functions, the objective function is expressed as

$$\mathcal{F}(k) = G(k) + \boldsymbol{b}^\intercal \left[\mathcal{P}_1(k), \mathcal{P}_2(k), \mathcal{P}_3(k), \mathcal{P}_4(k) \right]^\intercal. \tag{3.43}$$

The optimization formulation is $\max \mathcal{F}(k)$.

In this paper, the weighting vector in (3.37) is formulated as

$$\boldsymbol{w}(k) = \left[\frac{w_o(k)}{\bar{G}_o(k)}, \frac{w_c(k)}{\bar{G}_c(k)}, \frac{w_d(k)}{\bar{G}_d(k)} \right]^\intercal, \tag{3.44}$$

where

$$\bar{G}_o(k) = \mathbb{E}\left[\sum_{i=1}^{N_s} \mathrm{tr}(\mathbf{\Phi}_{i,o}(k))\right], \tag{3.45}$$

$$\bar{G}_c(k) = \mathbb{E}\left[\sum_{i=1}^{N_s}\sum_{\tau\in\boldsymbol{C}} \alpha_{i,\tau}(k-1)\delta_\tau(k-1)\mathrm{tr}(\mathbf{\Phi}_{i,\tau}(k))\right], \tag{3.46}$$

$$\bar{G}_d(k) = \mathbb{E}\left[\sum_{i=1}^{N_s}\sum_{t\in\boldsymbol{D}(k)} \beta_{i,t}(k-1)\rho_t(k-1)\mathrm{tr}(\mathbf{\Phi}_{i,t}(k))\right], \tag{3.47}$$

which are the informations gathered based on current UAV positions, measurements and importance factors, respectively. The values in the denominator in (3.44) scale the predictive information.

By changing the values in $\boldsymbol{w}(k)$, different objectives can be emphasized based on mission preference, the detection and tracking results of the search space centroid and detected targets, as well as on the scan results of cubes. For example, more weight $w_c(k)$ can be put onto $G_c(k)$ if UAVs scan the space but detect no target; more weight $w_d(k)$ can be put onto $G_d(k)$ if all cubes are scanned at least once and some targets are detected; more weight $w_o(k)$ can be put on $G_o(k)$ if the uncertainty about the state of search space centroid is large. If no target is detected (i.e., $\bar{G}_d(k)$ does not exist), the corresponding weight is configured as zero.

3.4.4 Receding Horizon Control

The above path planning and scanning can be extended to a receding horizon control [8]. By predicting the states of search space centroid and detected targets for the next K steps, the UAV path and sensor look direction are determined accordingly to maximize the objective function.

$$\max \sum_{\xi=k}^{k+K-1} \mathcal{F}(\xi). \tag{3.48}$$

Note that only the decision variables for the next step are used.

Since the hard constraints on UAV paths (3.15) to (3.19) are transformed into soft constraints in the objective function, as shown in (3.43), the generated path is possible to violate these constraints. Receding horizon control with a large K value will decrease the overall value of the constraint violation because of the long-term vision for path planning. Smoother trajectories will be generated for UAVs with $K > 1$ receding horizon control comparing to the path planning and scanning with $K = 1$ when the predicted motion models of search space centroid and the detected targets are accurate. However, the computational cost increases exponentially with the value of K.

3.5 Solution Technique

From Section 3.4 we can see that the path control inputs are continuous and the scan decision variables are discrete. Therefore, (3.48) is a mixed-integer nonlinear programming (MINLP) [3]. MINLP includes both nonlinear programming (NLP) and mixed-integer linear programming (MILP) as subproblems, which is an NP-hard combinatorial problem and its solution typically requires searching enormous search trees [3]. Details on MINLP models and solution methods can be found in [3].

Since the proposed MINLP problem must be solved in real time for online path planning and scanning, it is desirable to obtain a near optimal solution quickly than

to wait for the global optimal solution. In this paper, a genetic algorithm based cooperative control is developed. Note that other heuristics for solving MINLPs, such as particle swarm optimization (PSO) [26], tabu search (TS) [34], and simulated annealing (SA) [3], can also be used.

The use of GA to solve this MINLP is inspired by [30]. GA is a class of learning algorithms based on a parallel search for an optimal solution [14]. The parallel searches, which are performed synchronously in time steps, are called generations. In each generation, a certain number of path decisions and scan decisions called individuals are maintained. The whole set of individuals in a generation is referred to as the population. The main idea in GA is to preserve and create variations of individuals that seem most promising (i.e., with high fitness values) and remove the others. Denote the number of generations as N_{gen} and the population size N_{pop}. The number of cubes within the search space is $N_c = |\boldsymbol{C}|$. The number of detected targets at time k is $N_d^k = |\boldsymbol{D}(k)|$.

The individual (also known as the chromosome), which represents the decision variables in the proposed problem, is formed as follows: the first N_c variables represent the cubes, which take 1 if the corresponding cube will be scanned in the next step and 0 otherwise; the next N_d^k variables represent detected targets, which take 1 if the corresponding target will be scanned and tracked in the next step and 0 otherwise; and the rest of the variables are allocated for UAV path selection variables, which are the bank angles and flight path angles. Therefore, the length of individuals is $N_{\text{ind}} = N_c + N_d^k + 2N_s$ if path planning and scanning variables are generated for all UAVs.

To further decrease the computational cost, in our solution technique, an order

is defined for groups of UAVs to scan the search space: groups of UAVs scan the cubes and detected targets at different times and measurements are shared after the operation. Therefore, for the gth group, the length of individual becomes $N_{\text{ind}}^g = N_c + N_d^k + 2N_s^g$. The path planning and scanning algorithm is designed for a group of UAVs, which can be processed on a central UAV in the group or on each UAV so that each acts as a central node.

In order to handle the continuous values in GA, the bank angles and flight path angles are mapped to continuous values between 0 and 1

$$\bar{\vartheta}_i(k) = \frac{\vartheta_i(k) + \Theta}{2\Theta}, \text{ and } \bar{\psi}_i(k) = \frac{\psi_i(k) + \Psi}{2\Psi}, \tag{3.49}$$

$$0 \leq \bar{\vartheta}_i(k), \bar{\psi}_i(k) \leq 1. \tag{3.50}$$

Therefore, constraints (3.23) and (3.24) are satisfied when generating individuals. These continuous variables are divided into evenly spaced numbers in the GA and randomly selected when generating the chromosome. They are converted back to real values of angles when evaluating the fitness function and when the GA terminates.

Individuals of the initial population are generated at random under the constraint that the sum of the first $N_c + N_d^k$ variables be M, as in (3.30).

The fitness function (3.43) is evaluated in each time step. Constraints (3.23), (3.24), (3.30) and (3.31) are met when generating the individuals in the initial population and infeasible individuals to these constraints are repaired during the evolution.

Selection of parents to reproduce is based on the fitness of individuals. Individuals with high fitness values are most likely to be selected, whereas those with low fitness will be discarded. Many selection techniques, such as tournament selection

and roulette wheel selection, are available [14]. A binary tournament selection is used in this paper.

In the crossover operation, a locus is randomly chosen and the subsequences before and after that locus between two chromosomes (parents) are exchanged to create two new individuals (offsprings). Crossover is normally performed with a probability P_c, i.e., a fraction P_c of the new individuals are formed by crossover and the remaining fraction $1 - P_c$ are copied.

In the mutation operation, each variable of the new individual is changed with a certain probability P_m; binary variables (scan decision) are flipped and continuous variables (path selection) are replaced by a random value between 0 and 1.

Since crossover and mutation may produce individuals that violate constraint (3.30), a repair operator is designed to rectify infeasible individuals. If the summation of the first $N_c + N_d^k$ variables in the chromosome is greater than M, a number of 1 bits will be randomly selected and changed to 0 so that the summation of the binary variables is M. Therefore, constraint (3.30) is satisfied in each generation.

In each generation, the individuals are ranked by sorting their fitness values. A small proportion of the fittest individuals are copied into the next generation. These top scored individuals that are preserved unchanged through elitism remain eligible for selection as parents when breeding the remainder of the next generation. Elitism prevents losing the few best-found solutions and may improve the performance of GA [30].

In this paper, a cooperative control is proposed for UAVs in the same group. Each UAV performs the following tasks in each time step, which is also shown in Fig. 3.4. The UAV uses sensor S1 to scan the measurement of search space centroid, updates

the state and state covariance of the search space centroid, and transmits tracks for track-to-track fusion [21]. The fusion output of the search space centroid location is used to compute the approximate locations of cube centroids to direct sensor S2 scanning these regions of interests. The search space centroid state is used to predict the locations of the search space centroid and cube centroids for the next time step. Based on the locations of the cube centroids and detected targets, UAVs scan M cubes and/or detected targets according to the scan decision made at time $k-1$. The tracks of detected targets are updated and transmitted for track-to-track fusion. Meanwhile, the status of the cubes, which includes whether a cube is scanned, when the cube is scanned and the measurements in the cube, is updated. Based on the tracking and detection results, the importance factors of cubes and detected targets are updated and the states of detected targets are predicted. The path planning and scanning for the next time step is made by solving (3.48) based on the predicted states of the search space centroid, detected targets, cube centroids and importance factors.

3.6 Simulation for Optimal Sensor Configuration

In this simulation, one stationary target at $[0,0,0]^\intercal$ is to be localized by multiple sensors working in the same mode. Two modes, active and passive as discussed in Section 3.3, are used for the sensors. The difference between the simulation in this paper and those in [35] is that constraints (3.15) to (3.19) are now included. A stationary target is used here to illustrate the optimal adaptation capability of our algorithm and to provide geometric insights. A mobile target does not change the operation of the algorithm.

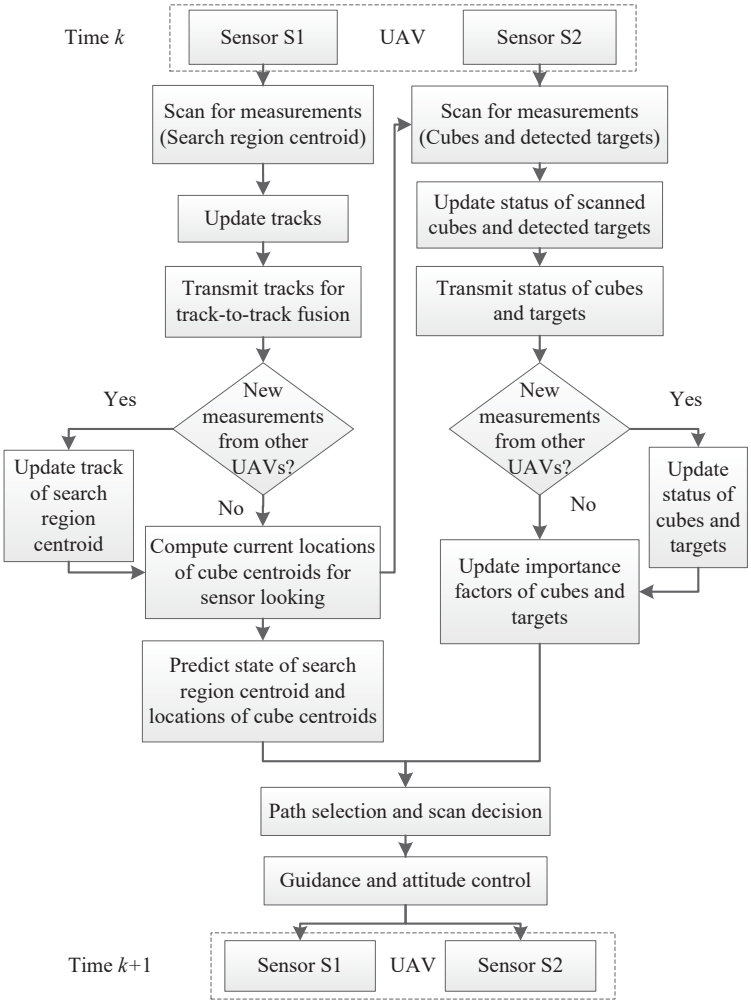


Figure 3.4: One cycle of the proposed cooperative control framework.

3.6.1 Optimal Sensor Configuration for Two Passive Sensors

Example 1. The sensor-to-target distance is fixed for both sensors: $d_{\text{s-p}} = 100$m. Let the sensor-dependent noise variances $\sigma_1 = \sigma_2 = 1°$, and $\gamma_1 = \gamma_2 = 1°$. The minimum sensor-to-sensor distance is $d_{\text{s-s}}^{\min} = 15$m. Different constraints on the minimum altitude and the maximum altitude are used.

The maximum sensor altitude is fixed at 95m and the optimal sensor configurations

are computed for different minimum sensor altitudes $h_1^{\min} = h_2^{\min} = \{0, 5, 10, ..., 95\}$m. The difference in the optimal azimuth angles of the two sensors $|\theta_1 - \theta_2|$ with different minimum sensor altitudes, along with the difference in the optimal elevation angles of the two sensors $|\phi_1 - \phi_2|$, is shown in Fig. 3.5. The optimal elevation angles of sensor 1 and sensor 2 are also shown in Fig. 3.5.

When $h_i^{\min} = 0$m, the optimal sensor configuration is obtained when $|\theta_1 - \theta_2| = 90^\circ$ and $\{\phi_1, \phi_2\} \in \{\{0,\ \arcsin(0.95)\}, \{\arcsin(0.95), 0\}\}$, which means that one sensor is placed on the plane of the minimum altitude (0m) while the other sensor is located on the plane of the maximum altitude (95m).

When $h_i^{\min} = 30$m, the optimal sensor configuration is obtained when $|\theta_1 - \theta_2| = 101.32^\circ$ and $\{\phi_1, \phi_2\} \in \{\{\arcsin(0.3), \arcsin(0.95)\}, \{\arcsin(0.95), \arcsin(0.3)\}\}$, which means that one sensor is placed on the plane of the minimum altitude (30m) while the other sensor is located on the plane of the maximum altitude (95m).

However, the above optimal configuration where one sensor is located on the minimum altitude plane and the other one is located on the maximum altitude plane is not always optimal. When $h_i^{\min} = 60$m, the optimal sensor configuration is obtained when $|\theta_1 - \theta_2| = 114.3^\circ$ and $\{\phi_1, \phi_2\} \in \{\{\arcsin(0.6), \arcsin(0.9)\}, \{\arcsin(0.9), \arcsin(0.6)\}\}$.

The minimum sensor altitude is fixed at 0m and the optimal sensor configurations are computed for different maximum sensor altitudes $h_1^{\max} = h_2^{\max} = \{0, 5, 10, ..., 95\}$m. The difference in the optimal azimuth angles and the difference in the optimal elevation angles of the two sensors (i.e., $|\theta_1 - \theta_2|$ and $|\phi_1 - \phi_2|$) are shown in Fig. 3.6. The optimal elevation angles of sensor 1 and sensor 2 are also shown in Fig. 3.6.

In Fig. 3.6, from $h_i^{\max} = 0$m to $h_i^{\max} = 65$m, the two sensors in the optimal configuration are located on the same altitude plane, which is the plane of maximum

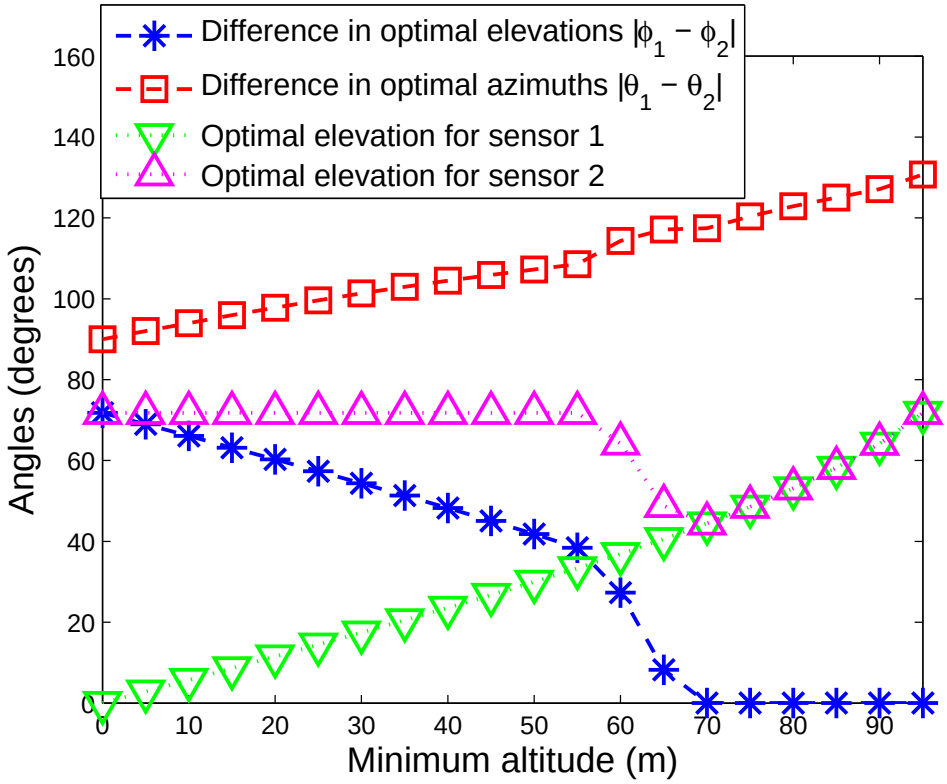


Figure 3.5: The difference in the optimal azimuth angles and the difference in the optimal elevation angles as the minimum altitude changes from 0m to 95m. ($d_{\text{s-p}} = 100\text{m}$, $d_{\text{s-s}}^{\min} = 15\text{m}$, $h_1^{\max} = h_2^{\max} = 95\text{m}$, $\sigma_1 = \sigma_2 = 1°$, and $\gamma_1 = \gamma_2 = 1°$.)

altitude. From $h_i^{\max} = 80\text{m}$ to $h_i^{\max} = 95\text{m}$, in the optimal configuration, one sensor is located on the minimum altitude plane (0m) while the other one is located on the maximum altitude plane. When the maximum altitude is 70m, one sensor is located on the altitude plane of 60m while the other one is located on the altitude plane of 68.2m. When the maximum altitude is 75m, one sensor is located on the altitude plane of 75m while the other one is located on the altitude plane of 34.9m.

In this scenario, no matter what the minimum/maximum altitude is, the UAVs will separate in azimuth to an angle greater than or equal to 90° to locate the target.

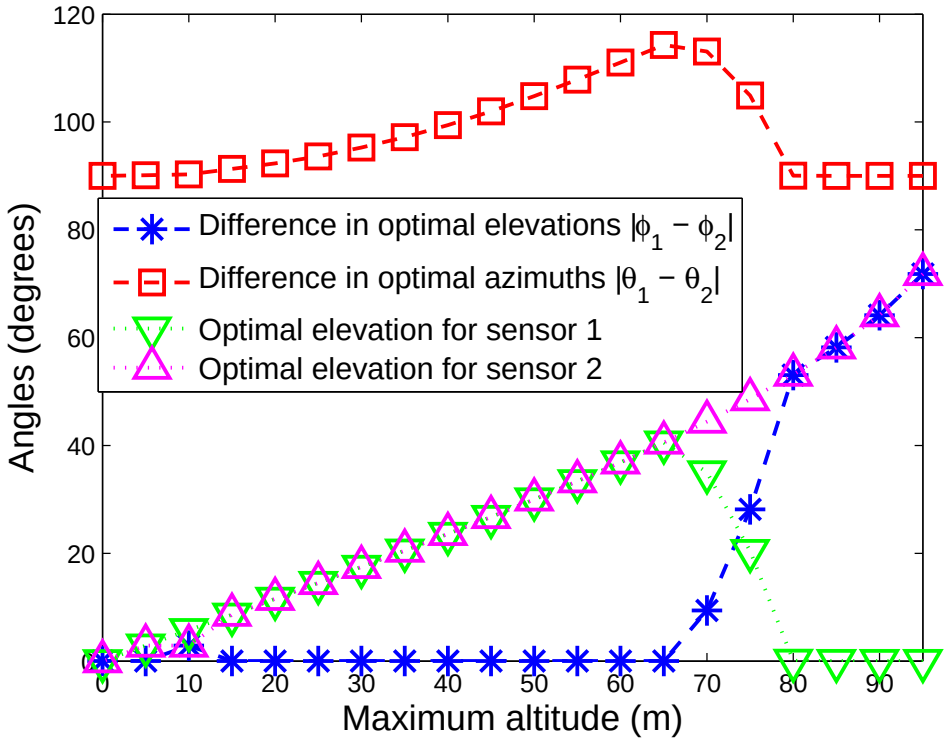


Figure 3.6: The difference in the optimal azimuth angles and the difference in the optimal elevation angles as the maximum altitude changes from 95m to 0m. ($d_{\text{s-p}} = 100\text{m}$, $d_{\text{s-s}}^{\min} = 15\text{m}$, $h_1^{\min} = h_2^{\min} = 0\text{m}$, $\sigma_1 = \sigma_2 = 1°$, and $\gamma_1 = \gamma_2 = 1°$.)

Example 2. The sensor-to-target distance and the minimum sensor-to-sensor distance are the same as those in Example 1. With $\sigma_1 = \sigma_2 = 0.1°$ and $\gamma_1 = \gamma_2 = 2°$, the optimal sensor configuration is computed with different constraints on the altitude.

The maximum sensor altitude is fixed at 95m and the optimal sensor configurations are computed for different minimum sensor altitudes $h_1^{\min} = h_2^{\min} = \{0, 5, 10, ..., 95\}\text{m}$. The difference in the optimal azimuth angles and the difference in the optimal elevation angles of the two sensors (i.e., $|\theta_1 - \theta_2|$ and $|\phi_1 - \phi_2|$) are shown in Fig. 3.7. The optimal elevation angles of the two sensors are the same and the two sensors are located on the plane of the minimum altitude. The difference in the optimal azimuth

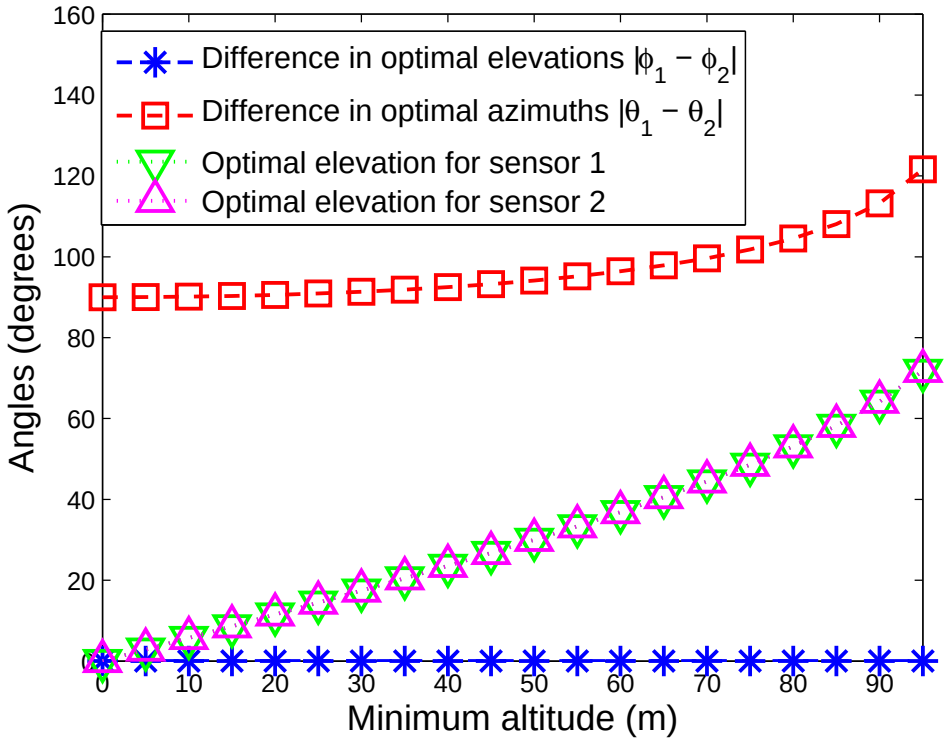


Figure 3.7: The difference in the optimal azimuth angles and the difference in the optimal elevation angles as the minimum altitude changes from 0m to 95m. ($d_{\text{s-p}} = 100$m, $d_{\text{s-s}}^{\min} = 15$m, $h_1^{\max} = h_2^{\max} = 95$m, $\sigma_1 = \sigma_2 = 0.1°$, and $\gamma_1 = \gamma_2 = 2°$.)

angles increases as the minimum altitude increases. In this scenario, the UAVs will fly at the same altitude and separate in azimuth to an angle greater than 90°.

The minimum sensor altitude is fixed at 0m and the optimal sensor configurations are fixed for different maximum sensor altitudes $h_1^{\max} = h_2^{\max} = \{0, 5, 10, ..., 95\}$m. The difference in the optimal azimuth angles and the difference in the optimal elevation angles of the two sensors (i.e., $|\theta_1 - \theta_2|$ and $|\phi_1 - \phi_2|$) are shown in Fig. 3.8. The optimal sensor configuration is obtained from $|\theta_1 - \theta_2| = 90°$ and $\phi_1 = \phi_2 = 0$, which is the optimal sensor placement in 2D [35].

Note that in UAV path planning, the minimum altitude is usually greater than 0,

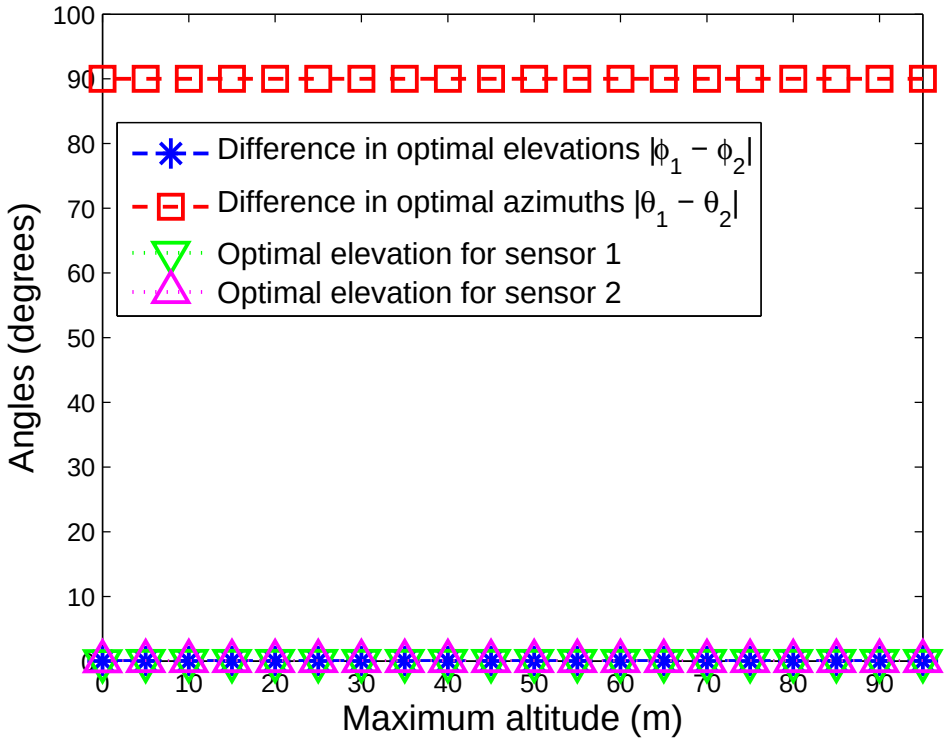


Figure 3.8: The difference in the optimal azimuth angles and the difference in the optimal elevation angles as the maximum altitude changes from 95m to 0m. ($d_{\text{s-p}} = 100\text{m}$, $d_{\text{s-s}}^{\min} = 15\text{m}$, $h_1^{\min} = h_2^{\min} = 0\text{m}$, $\sigma_1 = \sigma_2 = 0.1°$, and $\gamma_1 = \gamma_2 = 2°$.)

which corresponds to the configurations in Fig. 3.7.

Example 3. The simulation configurations are kept the same as those in Example 1 except that one constraint is added: the maximum sensor-to-sensor distance is $d_{\text{s-s}}^{\max} = 100\text{m}$. In Fig. 3.9, the maximum altitude is 95m and the minimum altitude changes from 0m to 95m. When the minimum altitude is no greater than 50m, one sensor is located at the altitude of 95m and the other one at 50m, with the difference in the optimal azimuth angles being 84.6°. Comparing Fig 3.9 with Fig. 3.5, when the minimum altitude is no greater than 50m, the difference in the optimal azimuth angles does not increase and the altitudes of both sensors cannot be lower than 50m

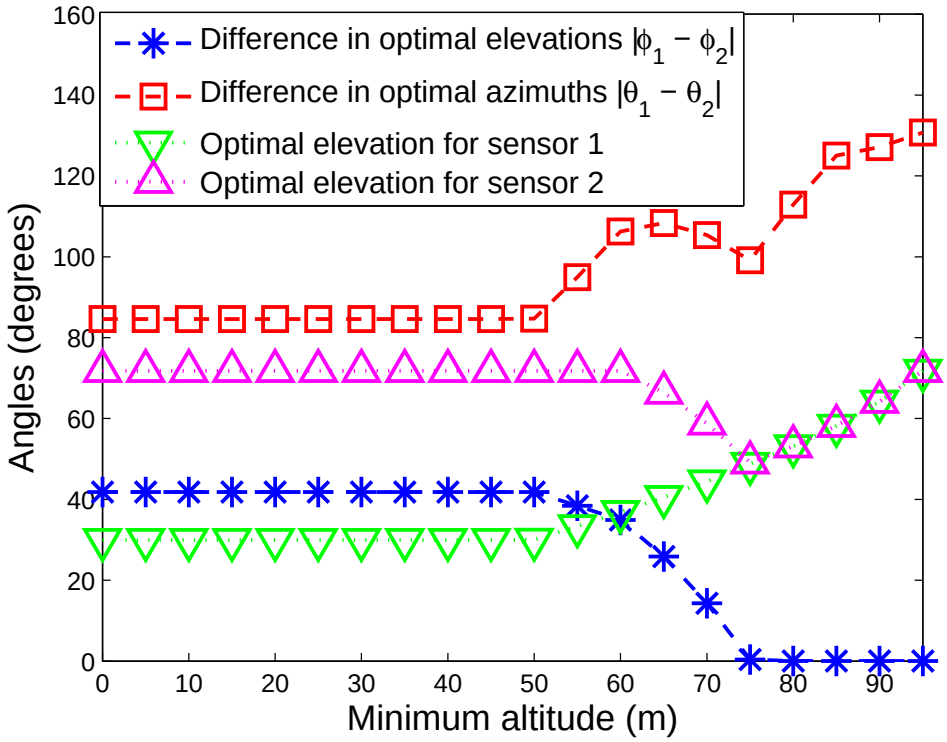


Figure 3.9: The difference in the optimal azimuth angles and the difference in the optimal elevation angles as the minimum altitude changes from 0m to 95m. ($d_{\text{s-p}} = 100\text{m}$, $d_{\text{s-s}}^{\min} = 15\text{m}$, $d_{\text{s-s}}^{\max} = 100\text{m}$, $h_1^{\max} = h_2^{\max} = 95\text{m}$, $\sigma_1 = \sigma_2 = 1°$, and $\gamma_1 = \gamma_2 = 1°$.)

because of the maximum sensor-to-sensor distance constraint. When the minimum altitude is no greater than 80m, the sensor-to-sensor distance in the optimal sensor configurations is 100m. When the minimum altitude is no less than 85m, the optimal sensor configurations are the same as those in Fig. 3.5 and the sensor-to-sensor distances are smaller than 100m.

The differences in the angles of the two sensors and the optimal elevation angles are shown in Fig. 3.10, when the minimum altitude is fixed to 0m, the maximum altitude changes from 0m to 95m and the maximum sensor-to-sensor distance requirement is

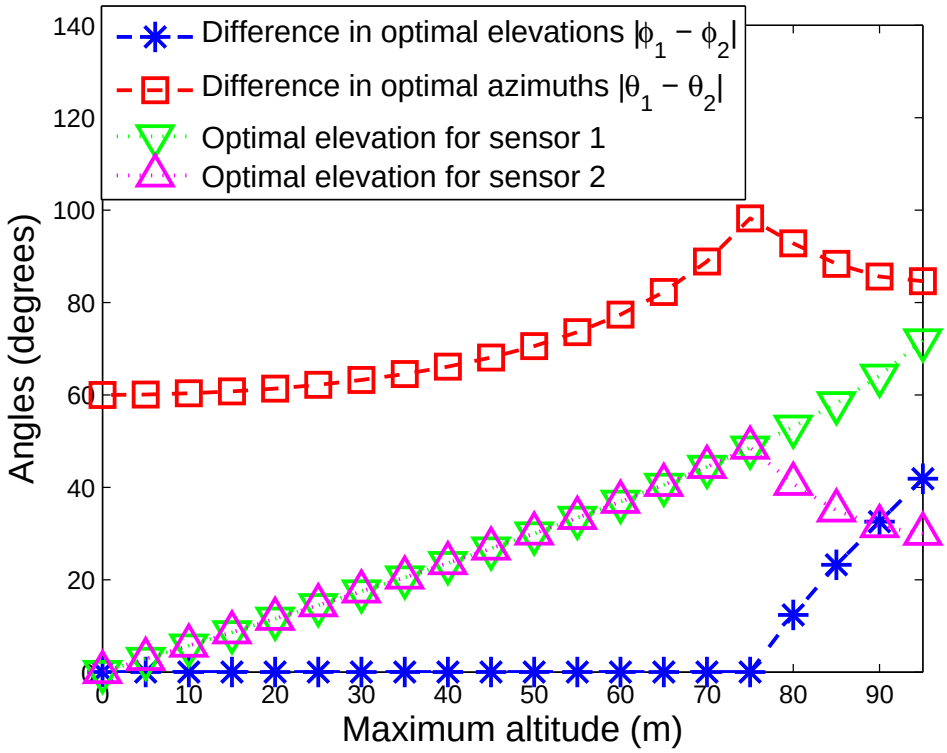


Figure 3.10: The difference in the optimal azimuth angles and the difference in the optimal elevation angles as the maximum altitude changes from 95m to 0m. ($d_{\text{s-p}} = 100\text{m}$, $d_{\text{s-s}}^{\min} = 15\text{m}$, $d_{\text{s-s}}^{\max} = 100\text{m}$, $h_1^{\min} = h_2^{\min} = 0\text{m}$, $\sigma_1 = \sigma_2 = 1^\circ$, and $\gamma_1 = \gamma_2 = 1^\circ$.)

100m.

3.6.2 Optimal Path Planning for Two Passive Sensors

In this subsection, we verify the optimal geometries in the previous examples using minimum mean-squared error (MMSE) estimation [2]. The target is located at $\boldsymbol{p} = [0, 0, 0]^{\mathsf{T}}\text{m}$. The sensors can move in any direction in 3D space. The sensor speed is $V = 4$ m/s. The constrained nonlinear programming problem in Section 3.3 is solved by the interior point algorithm [6].

Example 4. Let $\sigma_1 = \sigma_2 = 1^\circ$, $\gamma_1 = \gamma_2 = 1^\circ$, $d_{\text{s-s}}^{\min} = 15\text{m}$, and $h_1^{\max} = h_2^{\max} = 95\text{m}$

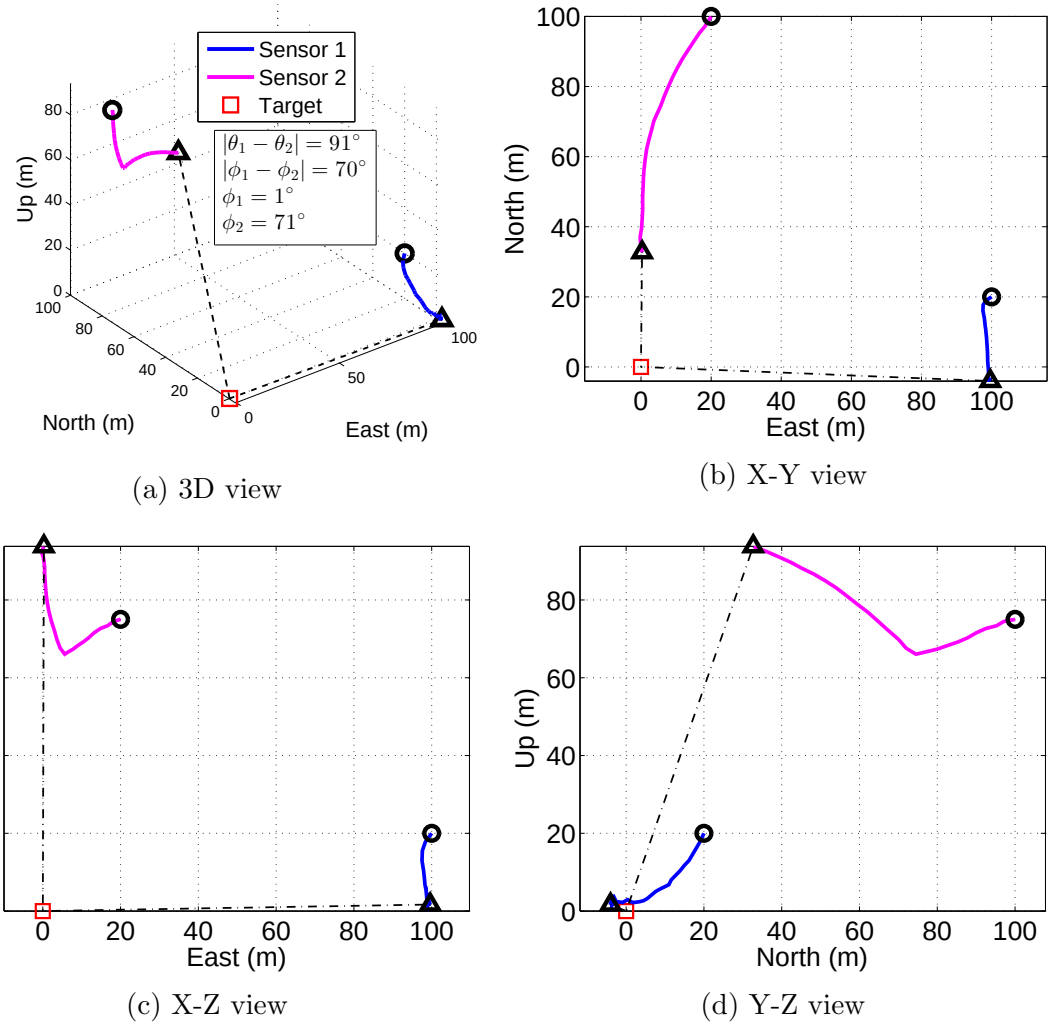
as in Example 1. The minimum sensor altitude is $h_1^{\min} = h_2^{\min} = 0$m. The minimum sensor-to-target distance is constrained: $d_1 = d_2 \geq d_{\text{s-p}} = 100$m. Two sensors start from initial positions $[100, 20, 20]^\intercal$m and $[20, 100, 75]^\intercal$m, respectively. The sensor trajectories are shown in Fig. 3.11, which are averaged over 100 Monte Carlo runs. The final sensor geometry matches that in Fig. 3.5 when the minimum altitude is 0m.

Example 5. Next we consider the setup in Example 2, i.e., $\sigma_1 = \sigma_2 = 0.1^\circ$, $\gamma_1 = \gamma_2 = 2^\circ$, $d_{\text{s-s}}^{\min} = 15$m, and $h_1^{\max} = h_2^{\max} = 95$m. The minimum sensor altitude is $h_1^{\min} = h_2^{\min} = 75$m. The minimum sensor-to-target distance is constrained: $d_1 = d_2 \geq d_{\text{s-p}} = 100$m. Two sensors start from initial positions $[110, 20, 90]^\intercal$m and $[-20, 110, 90]^\intercal$m, respectively. The sensor trajectories are shown in Fig. 3.12, which are averaged over 100 Monte Carlo runs. The final sensor geometry matches that in Fig. 3.8 when the minimum altitude is 75m.

3.6.3 Optimal Sensor Configuration for Two Active Sensors

Example 6. The simulation configurations are kept the same as those in Example 1 with $\eta_1 = \eta_2 = 5$m. The maximum sensor altitude is fixed at 95m and the optimal sensor configurations are computed for different minimum sensor altitudes $h_1^{\min} = h_2^{\min} = \{0, 5, 10, ..., 95\}$m. The differences in the optimal azimuth angles and the difference in the optimal elevation angles of the two sensors are shown in Fig. 3.13.

When $h_i^{\min} = 0$m, the optimal sensor configuration is obtained when $|\theta_1 - \theta_2| = 90^\circ$ and $\{\phi_1, \phi_2\} \in \{\{0, \arcsin(0.95)\}, \{\arcsin(0.95), 0\}\}$, which means that one sensor is placed on the plane of the minimum altitude (0m) while the other sensor is located on the plane of the maximum altitude (95m). This is the same as the configuration



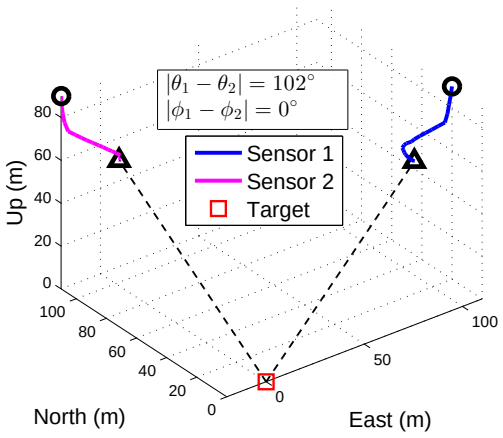
(a) 3D view

(b) X-Y view

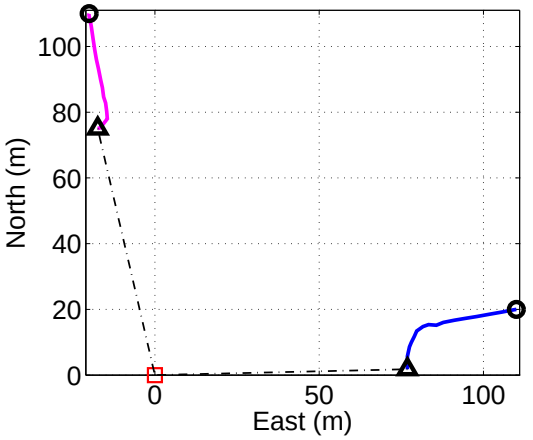
(c) X-Z view

(d) Y-Z view

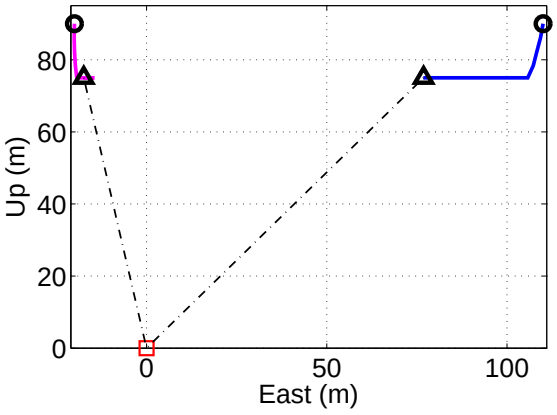
Figure 3.11: Optimal sensor trajectories for two passive sensors when $d_1 = d_2 \geq 100$m, $d_{\text{s-s}}^{\min} = 15$m, $h_1^{\max} = h_2^{\max} = 95$m, $h_1^{\min} = h_2^{\min} = 0$m, $\sigma_1 = \sigma_2 = 1^\circ$, and $\gamma_1 = \gamma_2 = 1^\circ$. The black '$\circ$' symbols mark the sensor initial positions and the black '$\triangle$' symbols mark the final positions.



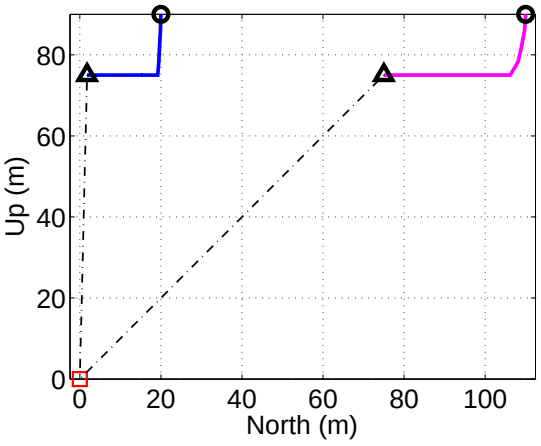
(a) 3D view



(b) X-Y view



(c) X-Z view



(d) Y-Z view

Figure 3.12: Optimal sensor trajectories for two passive sensors when $d_1 = d_2 \geq 100$m, $d_{\text{s-s}}^{\min} = 15$m, $h_1^{\max} = h_2^{\max} = 95$m, $h_1^{\min} = h_2^{\min} = 70$m, $\sigma_1 = \sigma_2 = 0.1°$, and $\gamma_1 = \gamma_2 = 2°$. The black '$\circ$' symbols mark the sensor initial positions and the black '$\triangle$' symbols mark the final positions.

for two passive sensors.

When $h_i^{\min} = 30$m, the optimal sensor configuration is obtained when $|\theta_1 - \theta_2| = 98.76^\circ$ and $\{\phi_1, \phi_2\} \in \{\{\arcsin(0.3), \arcsin(0.95)\}, \{\arcsin(0.95), \arcsin(0.3)\}\}$, which means that one sensor is placed on the plane of the minimum altitude (30m) while the other sensor is located on the plane of the maximum altitude (95m).

However, the above optimal configuration where one sensor is located on the minimum altitude plane and the other one is located on the maximum altitude plane is not always optimal. When $h_i^{\min} = 75$m, the optimal sensor configuration is obtained from $|\theta_1 - \theta_2| = 113.5^\circ$ and $\phi_1 = \phi_2 = \arcsin(0.75)$.

The minimum sensor altitude is fixed at 0m and the optimal sensor configurations are computed for different maximum sensor altitudes $h_1^{\max} = h_2^{\max} = \{0, 5, 10, ..., 95\}$m. The difference in the optimal azimuth angles and the difference in the optimal elevation angles are shown in Fig. 3.14.

In Fig. 3.14, from $h_i^{\max} = 0$m to $h_i^{\max} = 70$m ($i = 1, 2$), the two sensors in the optimal configuration are located on the same altitude plane, which is the plane of maximum altitude. From $h_i^{\max} = 85$m to $h_i^{\max} = 95$m ($i = 1, 2$), in the optimal configuration, one sensor is located on the minimum altitude plane (0m) and the other one is located on the maximum altitude plane. When the maximum altitude is 75m, one sensor is located on the altitude plane of 75m and the other one is located on the altitude plane of 56.18m. When the maximum altitude is 80m, one sensor is located on the altitude plane of 80m and the other one is located on the altitude plane of 16.75m.

Example 7. The simulation configuration is the same as Example 2 with $\eta_1 = \eta_2 = 5$m. The maximum sensor altitude is fixed at 95m and the optimal sensor

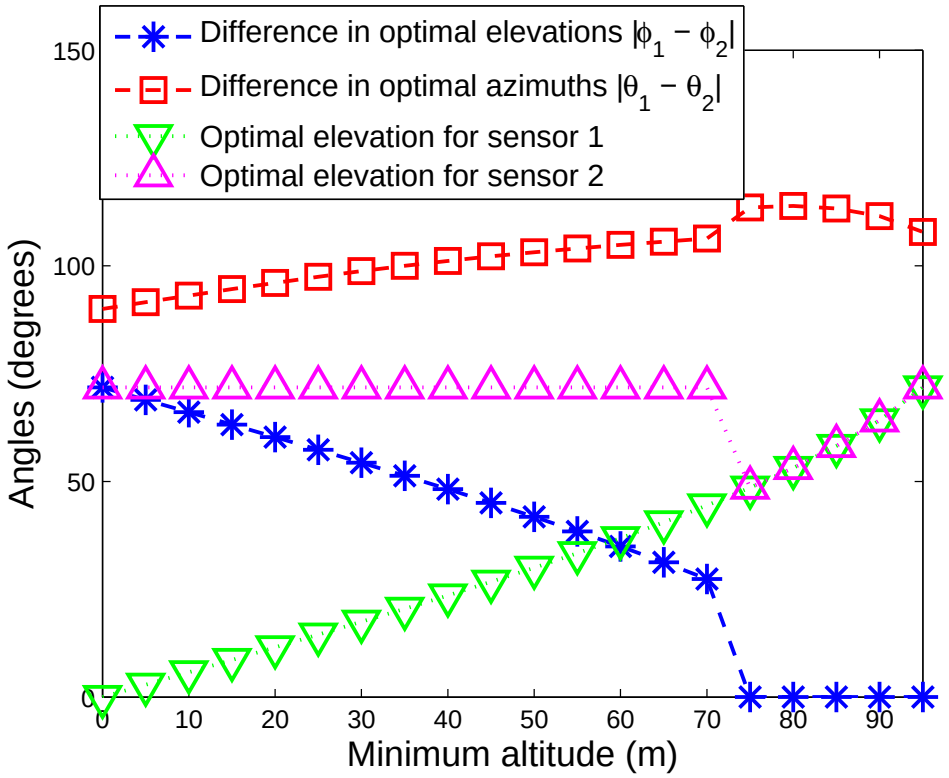


Figure 3.13: The difference in the optimal azimuth angles and the difference in the optimal elevation angles as the minimum altitude changes from 0m to 95m. ($d_{\text{s-p}} = 100$m, $d_{\text{s-s}}^{\min} = 15$m, $h_1^{\max} = h_2^{\max} = 95$m, $\sigma_1 = \sigma_2 = 1°$, $\gamma_1 = \gamma_2 = 1°$, and $\eta_1 = \eta_2 = 5$m.)

configurations are computed for different minimum sensor altitudes $h_1^{\min} = h_2^{\min} = \{0, 5, 10, ..., 95\}$m. The difference in the optimal azimuth angles and the difference in the optimal elevation angles are shown in Fig. 3.15. The optimal elevation angles of the two sensors are the same and the two sensors are located on the plane of the minimum altitude. The difference in the optimal azimuth angle increases as the minimum altitude increases.

In this scenario, the UAVs fly at the same altitude and separate in azimuth to an angle from 90° to 91.82°.

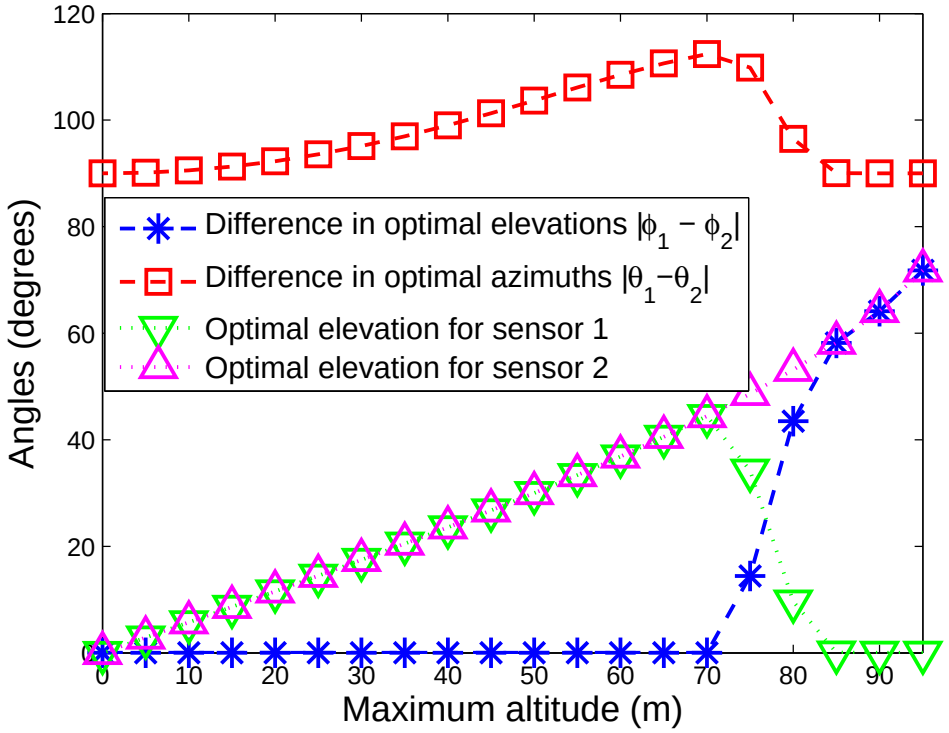


Figure 3.14: The difference in the optimal azimuth angles and the difference in the optimal elevation angles as the maximum altitude changes from 95m to 0m. ($d_{\text{s-p}} = 100\text{m}$, $d_{\text{s-s}}^{\min} = 15\text{m}$, $h_1^{\min} = h_2^{\min} = 0\text{m}$, $\sigma_1 = \sigma_2 = 1°$, $\gamma_1 = \gamma_2 = 1°$, and $\eta_1 = \eta_2 = 5\text{m}$.)

The minimum sensor altitude is fixed at 0m and the optimal sensor configurations are computed for different maximum sensor altitudes $h_1^{\max} = h_2^{\max} = \{0, 5, 10, ..., 95\}\text{m}$. The differences in the optimal azimuth angles and the difference in the optimal elevation angles are shown in Fig. 3.16. The optimal sensor configuration is obtained from $|\theta_1 - \theta_2| = 90°$ and $\phi_1 = \phi_2 = 0$, which is the optimal sensor placement in 2D [35].

Note that in UAV path planning, the minimum altitude is usually greater than 0, which corresponds to the configurations in Fig. 3.15.

Example 8. The simulation configurations are kept the same as those in Example 6

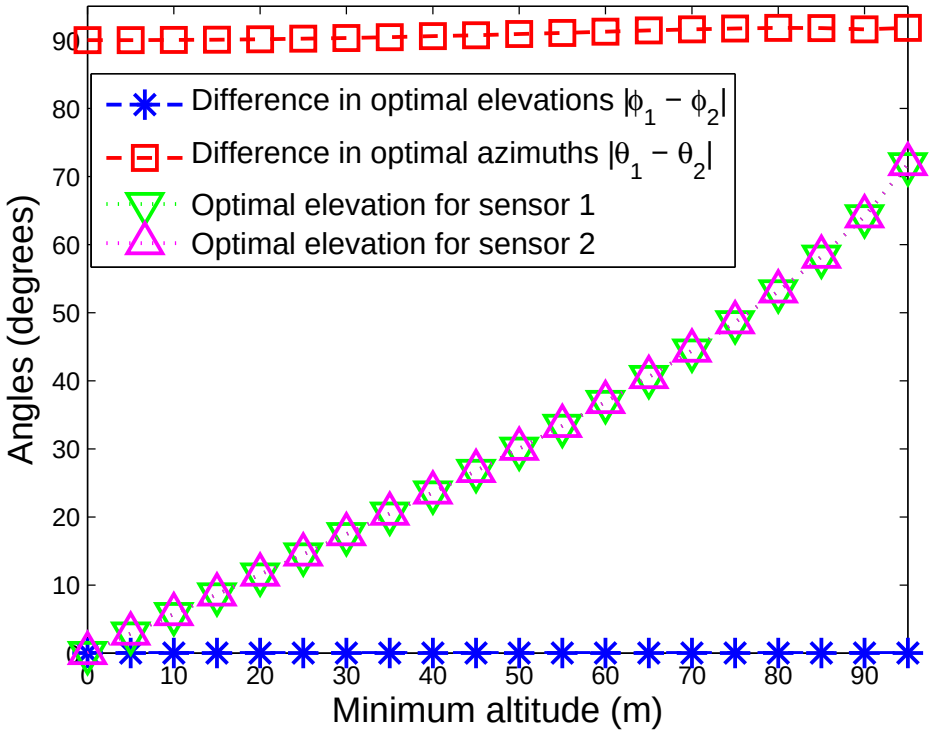


Figure 3.15: The difference in the optimal azimuth angles and the difference in the optimal elevation angles as the minimum altitude changes from 0m to 95m. ($d_{\text{s-p}} = 100\text{m}$, $d_{\text{s-s}}^{\min} = 15\text{m}$, $h_1^{\max} = h_2^{\max} = 95\text{m}$, $\sigma_1 = \sigma_2 = 0.1°$, $\gamma_1 = \gamma_2 = 2°$, and $\eta_1 = \eta_2 = 5\text{m}$.)

except that one constraint is added: the maximum sensor-to-sensor distance is 100m. In Fig. 3.17, the maximum altitude is 95m and the minimum altitude changes from 0m to 95m. When the minimum altitude is no greater than 50m, one sensor is located at the altitude of 95m and the other one at 51.34m, while the difference of optimal azimuth angles is 87.37°.

The angle differences and the optimal elevation angles are shown in Fig. 3.18, when the minimum altitude is fixed at 0m, the maximum altitude changes from 0m to 95m, and the sensor-to-sensor distances of all optimal configurations in different

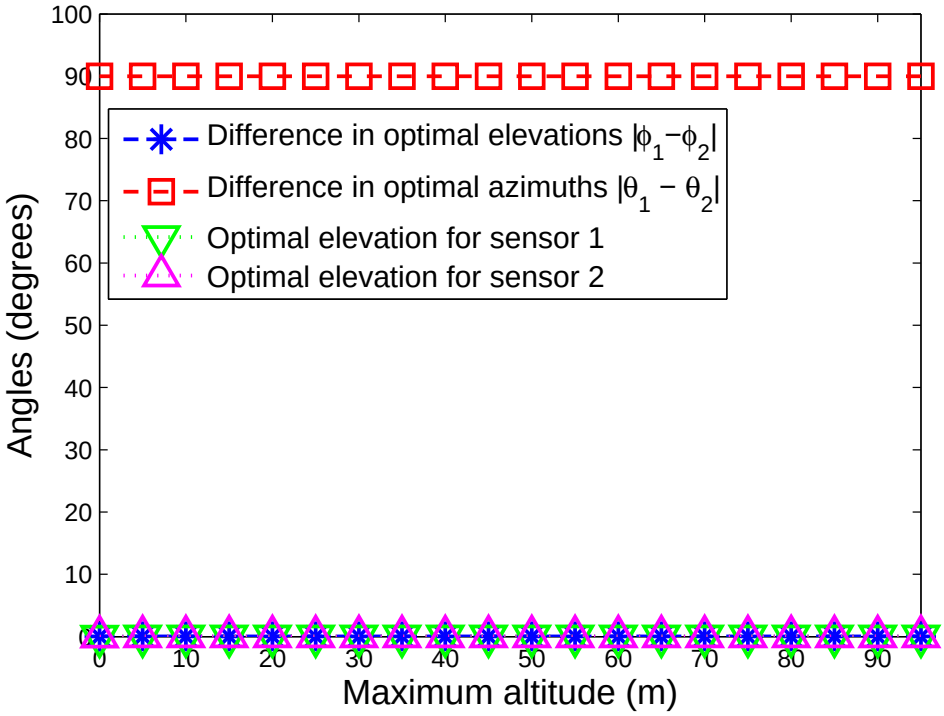


Figure 3.16: The difference in the optimal azimuth angles and the difference in the optimal elevation angles as the maximum altitude changes from 95m to 0m. ($d_{\text{s-p}} = 100\text{m}$, $d_{\text{s-s}}^{\min} = 15\text{m}$, $h_1^{\min} = h_2^{\min} = 0\text{m}$, $\sigma_1 = \sigma_2 = 0.1°$, $\gamma_1 = \gamma_2 = 2°$, and $\eta_1 = \eta_2 = 5\text{m}$.)

constraints are 100m.

3.6.4 Optimal Path Planning for Two Active Sensors

Example 9. Let $\sigma_1 = \sigma_2 = 1°$, $\gamma_1 = \gamma_2 = 1°$, $\eta_1 = \eta_2 = 5\text{m}$, $d_{\text{s-s}}^{\min} = 15\text{m}$, $h_1^{\max} = h_2^{\max} = 95\text{m}$, and $h_1^{\min} = h_2^{\min} = 65\text{m}$. The minimum sensor-to-target distance is constrained: $d_1 = d_2 \geq d_{\text{s-p}} = 100\text{m}$. The initial positions of the two sensors are $\boldsymbol{s}_1 = [100, 20, 80]^{\intercal}\text{m}$ and $\boldsymbol{s}_2 = [20, 100, 80]^{\intercal}\text{m}$, respectively. The sensor trajectories are shown in Fig. 3.19. The final sensor geometry matches that in Fig. 3.14: one sensor is located at the altitude of 95m, the other one is located at the altitude of 65m and

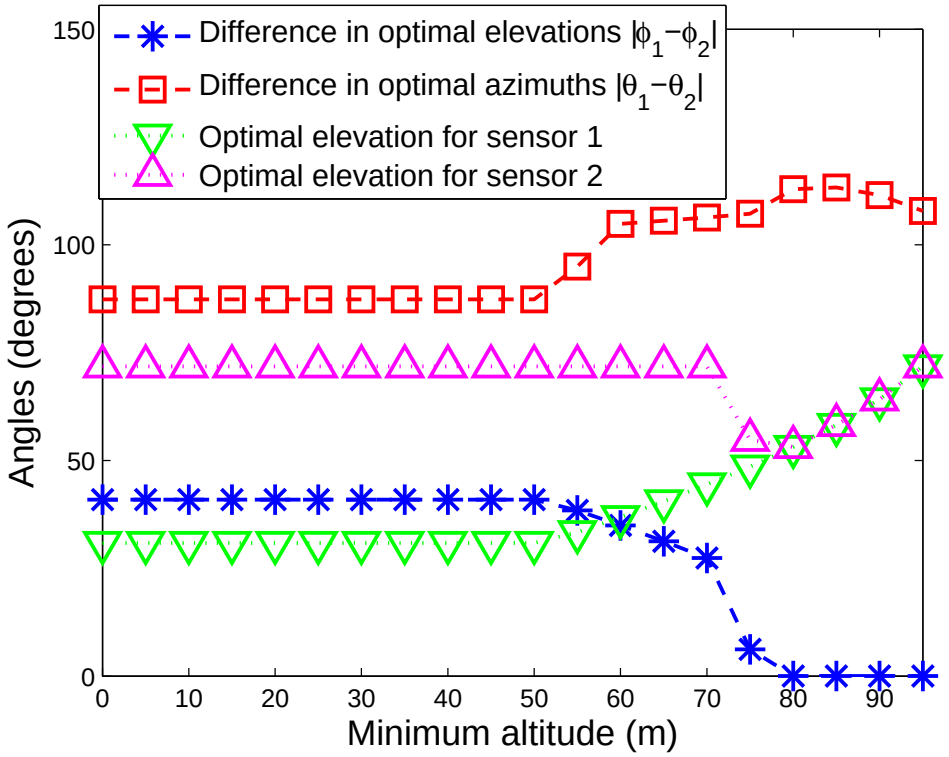


Figure 3.17: The difference in the optimal azimuth angles and the difference in the optimal elevation angles as the minimum altitude changes from 0m to 95m. ($d_{\text{s-p}} = 100$m, $d_{\text{s-s}}^{\min} = 15$m, $d_{\text{s-s}}^{\max} = 100$m, $h_1^{\max} = h_2^{\max} = 95$m, $\sigma_1 = \sigma_2 = 1°$, $\gamma_1 = \gamma_2 = 1°$, and $\eta_1 = \eta_2 = 5$m.)

the azimuth difference between the two active sensors is 105.6°.

3.6.5 Optimal Sensor Configuration for Three Passive Sensors

Example 10. The sensor-to-target distance is fixed for three passive sensors: $d_{\text{s-p}} = 100$m. Let $\sigma_1 = \sigma_2 = 1°$, and $\gamma_1 = \gamma_2 = 1°$. The minimum sensor-to-sensor distance is $d_{\text{s-s}}^{\min} = 15$m. Different constraints on the minimum altitude and the maximum altitude are used.

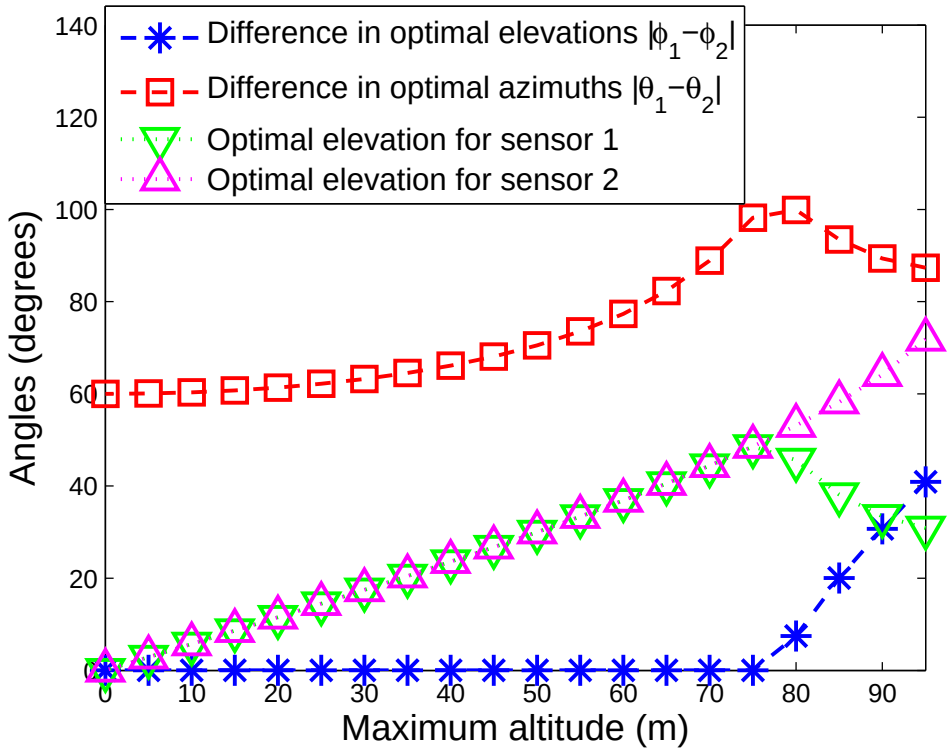
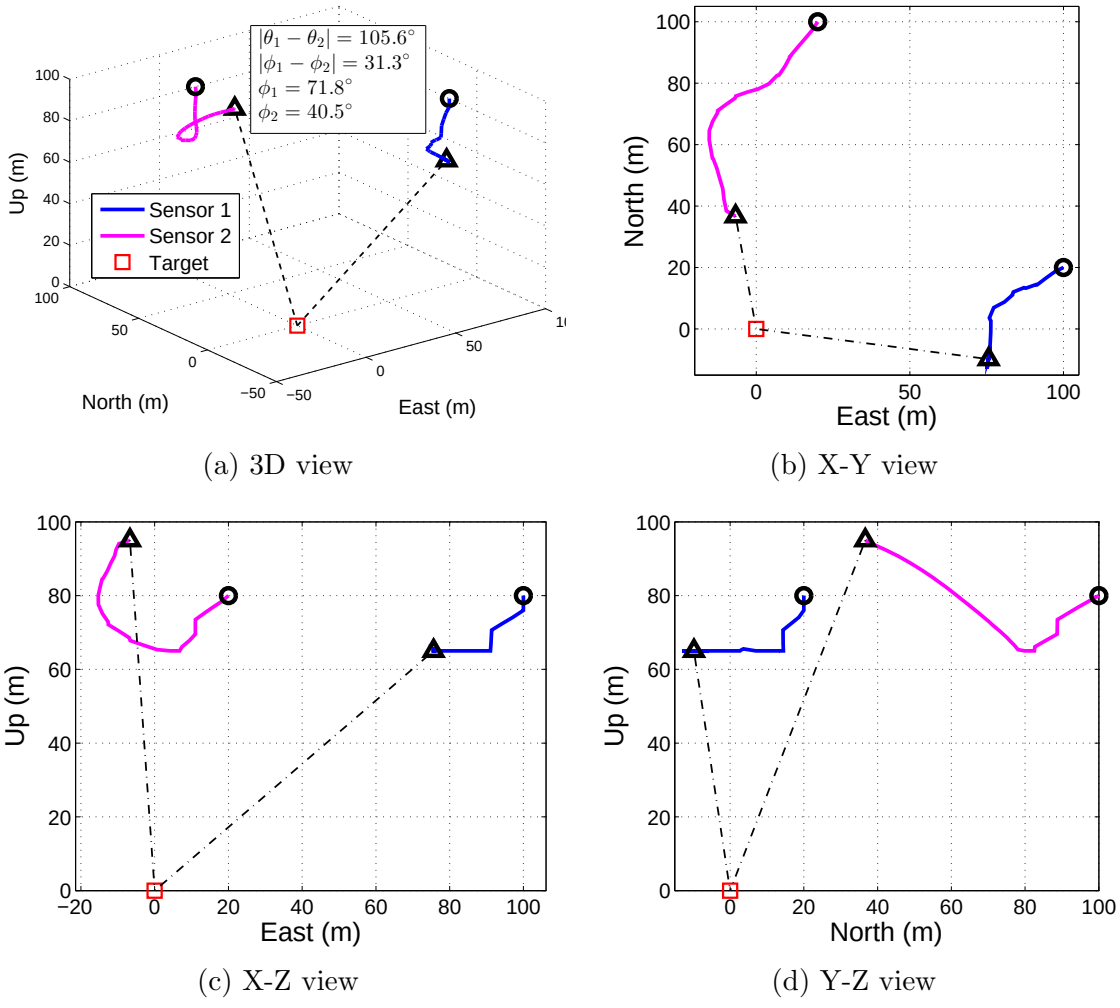


Figure 3.18: The difference in the optimal azimuth angles and the difference in the optimal elevation angles as the maximum altitude changes from 95m to 0m. ($d_{\text{s-p}} = 100\text{m}$, $d_{\text{s-s}}^{\min} = 15\text{m}$, $d_{\text{s-s}}^{\max} = 100\text{m}$, $h_1^{\min} = h_2^{\min} = 0\text{m}$, $\sigma_1 = \sigma_2 = 1°$, $\gamma_1 = \gamma_2 = 1°$, and $\eta_1 = \eta_2 = 5\text{m}$.)

The maximum sensor altitude is fixed at 95m and the optimal sensor configurations are computed for different minimum sensor altitude constraints $h_{1,2,3}^{\min} = \{0, 30, 60, 90\}\text{m}$. The optimal sensor configurations are shown in Fig. 3.20.

Note that in the following examples, the azimuth angle of sensor 1 (or UAV 1) is fixed to 0°. However, the FIM is invariant to xy-plane rotations [35], and the relative positions between the sensors and between the sensor and the target do not change.

When $h_{1,2,3}^{\min} = 0\text{m}$, the optimal sensor configuration is obtained when $\{\theta_1 =$



(a) 3D view

(b) X-Y view

(c) X-Z view

(d) Y-Z view

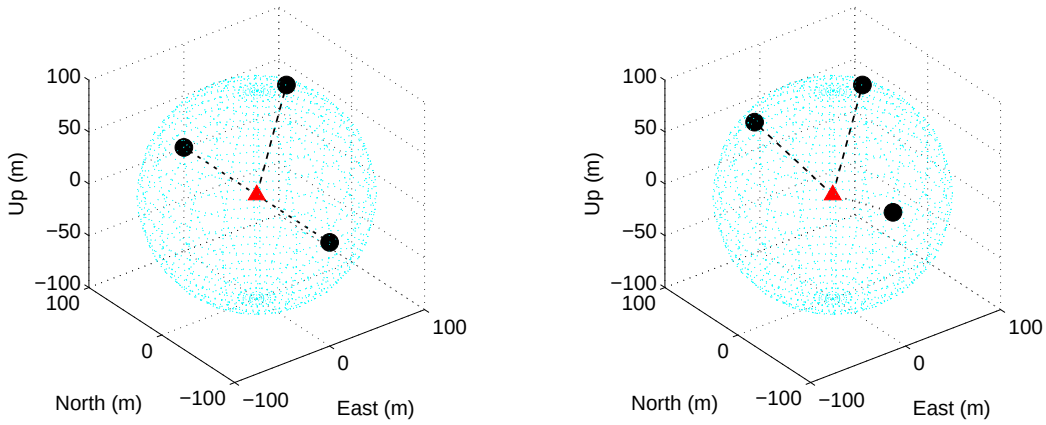
Figure 3.19: Optimal sensor trajectories for two active sensors when $d_1 = d_2 \geq 100$m, $d_{\text{s-s}}^{\min} = 15$m, $h_1^{\max} = h_2^{\max} = 95$m, $h_1^{\min} = h_2^{\min} = 65$m, $\sigma_1 = \sigma_2 = 1°$, $\gamma_1 = \gamma_2 = 1°$, and $\eta_1 = \eta_2 = 5$m. The black '$\circ$' symbols mark the sensor initial positions and the black '$\triangle$' symbols mark the final positions.

$0°,\ \phi_1 = \arcsin(0.95)\}$, $\{\theta_2 = 90°, \phi_2 = 0\}$, and $\{\theta_3 = -90°, \phi_3 = 0\}$, where two sensors are located at the minimum altitude (0m) and the third one is located at the maximum altitude (95m). When $h^{\min}_{1,2,3} = 30$m, the optimal sensor configuration is obtained when $\{\theta_1 = 0°, \phi_1 = \arcsin(0.95)\}$, $\{\theta_2 = 95.7°, \phi_2 = \arcsin(0.3)\}$, and $\{\theta_3 = -95.7°, \phi_3 = \arcsin(0.3)\}$, where two sensors are located at the minimum altitude (30m) and the third one is located at the maximum altitude (95m). When $h^{\min}_{1,2,3} = 60$m, the optimal configuration is obtained when $\{\theta_1 = 0°, \phi_1 = \arcsin(0.95)\}$, $\{\theta_2 = 99.8°, \phi_2 = \arcsin(0.6)\}$, and $\{\theta_3 = -99.8°, \phi_3 = \arcsin(0.6)\}$, where two sensors are located at the minimum altitude (60m) and the third one is located at the maximum altitude (95m). When $h^{\min}_{1,2,3} = 90$m, the optimal sensor configuration is obtained when $\{\theta_1 = 0°, \phi_1 = \arcsin(0.9)\}$, $\{\theta_2 = 120°, \phi_2 = \arcsin(0.9)\}$ and $\{\theta_3 = -120°, \phi_3 = \arcsin(0.9)\}$, where the three sensors are located at the same altitude (90m).

Example 11. The sensor-to-target distance is fixed for three passive sensors: $d_{\text{s-p}} = 100$m. Let $\sigma_1 = \sigma_2 = \sigma_3 = 0.1°$, and $\gamma_1 = \gamma_2 = \gamma_3 = 1°$. The minimum sensor-to-sensor distance is $d^{\min}_{\text{s-s}} = 15$m. Different constraints on the minimum altitude and the maximum altitude are used.

First, the maximum sensor altitude is fixed at 95m and the optimal configurations are computed for different minimum sensor altitude constraints $h^{\min}_{1,2,3} = \{0, 30, 60, 90\}$m. The optimal sensor configurations are shown in Fig. 3.21.

When $h^{\min}_{1,2,3} = 0$m, the optimal sensor configuration is obtained when $\{\theta_1 = 0°,\ \phi_1 = 0\}$, $\{\theta_2 = 90°, \phi_2 = 0\}$, and $\{\theta_3 = -90°, \phi_3 = 0\}$, where the three sensors are located at the minimum altitude (0m). When $h^{\min}_{1,2,3} = 30$m, the optimal sensor configuration is obtained when $\{\theta_1 = 0°, \phi_1 = \arcsin(0.3)\}$, $\{\theta_2 = 120°, \phi_2 = \arcsin(0.3)\}$,



(a) $h_{1,2,3}^{\min} = 0\text{m}$ (b) $h_{1,2,3}^{\min} = 30\text{m}$

(c) $h_{1,2,3}^{\min} = 60\text{m}$ (d) $h_{1,2,3}^{\min} = 90\text{m}$

Figure 3.20: Optimal sensor configurations for different minimum altitude constraints on three passive sensors when $d_{\text{s-p}} = 100\text{m}$, $d_{\text{s-s}}^{\min} = 15\text{m}$, $h_{1,2,3}^{\max} = 95\text{m}$, $\sigma_{1,2,3} = 1^\circ$, $\gamma_{1,2,3} = 1^\circ$. The '•' symbols mark the sensor positions and the '▲' symbols mark the target positions.

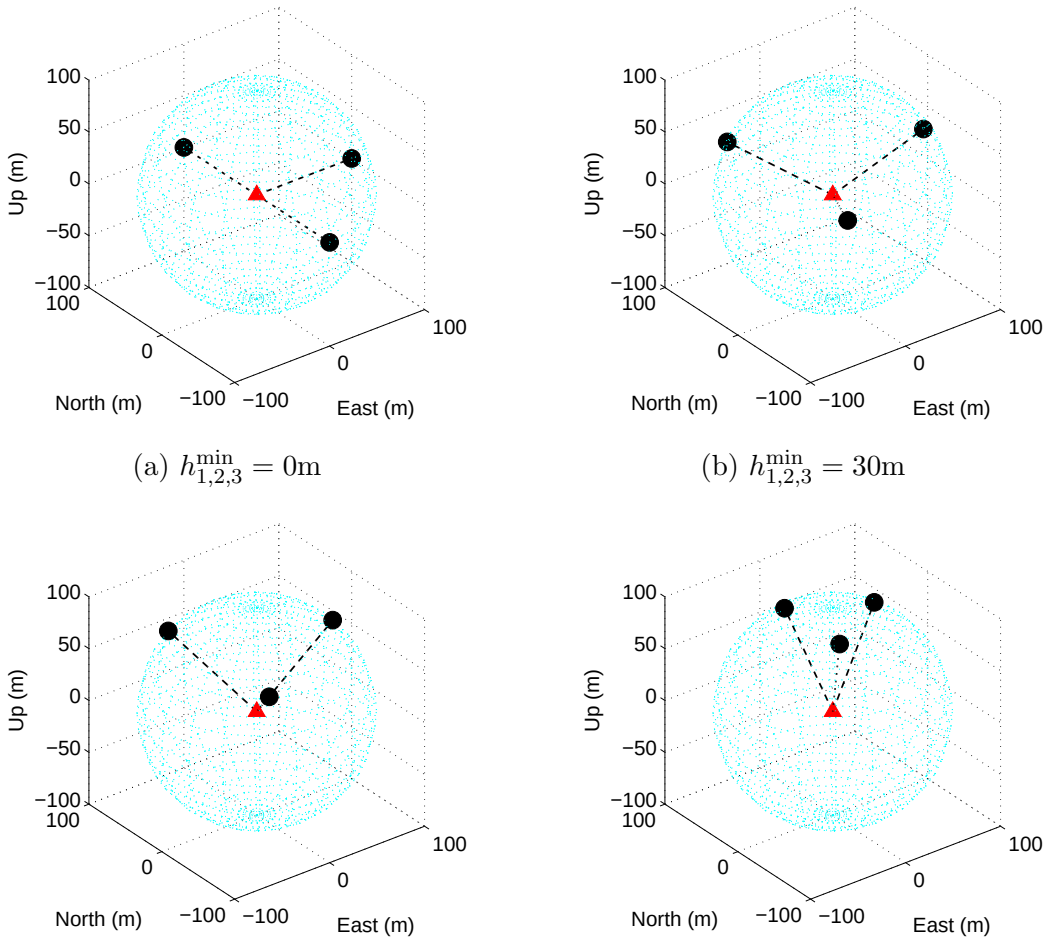
and $\{\theta_3 = -120°, \phi_3 = \arcsin(0.3)\}$, where the three sensors are located at the minimum altitude (30m). When $h^{\min}_{1,2,3} = 60$m, the optimal sensor configuration is obtained when $\{\theta_1 = 0°, \phi_1 = \arcsin(0.6)\}$, $\{\theta_2 = 120°, \phi_2 = \arcsin(0.6)\}$, and $\{\theta_3 = -120°$, $\phi_3 = \arcsin(0.6)\}$, where the three sensors are located at the minimum altitude (60m). When $h^{\min}_{1,2,3} = 90$m, the optimal sensor configuration is obtained when $\{\theta_1 = 0°$, $\phi_1 = \arcsin(0.9)\}$, $\{\theta_2 = 120°, \phi_2 = \arcsin(0.9)\}$, and $\{\theta_3 = -120°, \phi_3 = \arcsin(0.9)\}$, where the three sensors are located at the same altitude (90m).

3.6.6 Optimal Path Planning for Three Passive Sensors

In this subsection, we verify the optimal geometries for three passive sensors using MMSE estimation [2]. Let $\sigma_1 = \sigma_2 = \sigma_3 = 1°$, $\gamma_1 = \gamma_2 = \gamma_3 = 1°$, $d^{\min}_{\text{s-s}} = 15$m, $h^{\max}_{1,2,3} = 95$m, $h^{\min}_{1,2,3} = 85$m. The sensor-to-target distance should be no less than 100m. The three sensors start moving from initial positions $[110, 0, 95]^\intercal$m, $[-10, 110, 95]^\intercal$m and $[-10, -110, 95]^\intercal$m, respectively. The sensor trajectories are shown in Fig. 3.22, which is the average over 100 Monte Carlo runs. Finally, these three sensors stop at the altitude of 85m and form an equilateral triangle with the target at the center, which is consistent with the results in Fig. 3.21b, 3.21c and 3.21d.

3.7 Simulation for Path Planning and Scanning

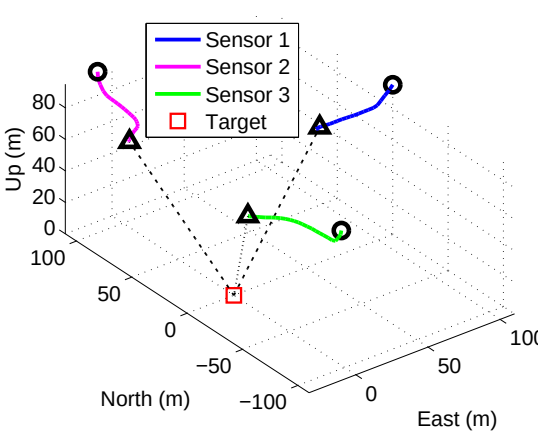
Two UAVs in the same group are to cooperatively track, follow and scan a mobile search space, search for possible targets inside this 3D space and track any detected targets. The 3D search space is divided into small 150m long cubes. Each cube is to be scanned at least once. Sensor S1 on each UAV is a radar with range, azimuth and



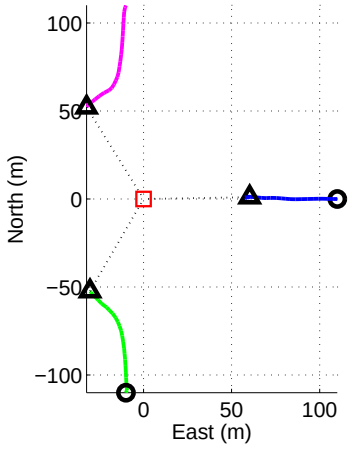
(a) $h^{\min}_{1,2,3} = 0\text{m}$ (b) $h^{\min}_{1,2,3} = 30\text{m}$

(c) $h^{\min}_{1,2,3} = 60\text{m}$ (d) $h^{\min}_{1,2,3} = 90\text{m}$

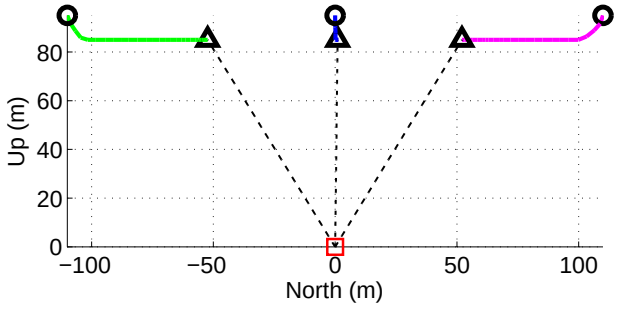
Figure 3.21: Optimal sensor configurations for different minimum altitude constraints on three passive sensors when $d_{\text{s-p}} = 100\text{m}$, $d^{\min}_{\text{s-s}} = 15\text{m}$, $h^{\max}_1 = h^{\max}_2 = 95\text{m}$, $\sigma_{1,2,3} = 0.1^\circ$, $\gamma_{1,2,3} = 1^\circ$. The '•' symbols mark the sensor positions and the '▲' symbols mark the target positions.



(a) 3D view



(b) X-Y view



(c) Y-Z view

Figure 3.22: Optimal sensor trajectories for three passive sensors when $d_{\text{s-p}} = 100\text{m}$, $\sigma_{1,2,3} = 1^\circ$, $\gamma_{1,2,3} = 1^\circ$, $d_{\text{s-s}}^{\min} = 15\text{m}$, $h_{1,2,3}^{\max} = 95\text{m}$, $h_{1,2,3}^{\min} = 85\text{m}$. The '$\circ$' symbols mark the sensor initial positions and the '$\triangle$' symbols mark the final positions.

elevation measurements, used to track the centroid of the search space. Sensor S2 on each UAV is a passive sensor with azimuth and elevation measurements, used to scan $M = 6$ cubes in each period of its operation.

For both UAVs, the speed is $V = 50$m/s, the maximum bank angle is $\Theta = 30^\circ$ and the maximum flight path angle is $\Psi = 12^\circ$. The maximum altitude is 5000m, the minimum altitude is 500m, the minimum distance between UAVs is 200m, the distance from a UAV to the centroid of the search space is to be greater than 1000m. Therefore, constraints (3.15), (3.17), (3.18), and (3.19) are enforced with $\boldsymbol{b} = [1, 1, 1, 1]^\intercal$ in (3.43). The initial locations of the UAVs are $[-1500, 1200, 660]^\intercal$m and $[-1500, -1200, 660]^\intercal$m, respectively. The initial headings of both UAVs are $[1, 0, 0]^\intercal$ (to the east).

In the simulation, the detection probabilities of sensor S1 and S2 are specified as functions of the range [27], as shown in Fig. 3.23. The Extended Kalman filter (EKF) and the Integrated Probability Data Association (IPDA) algorithm are used in target tracking with both sensors [2, 18, 20].

The 3D search space has a volume of 1000m in length, 400m in width and 250m in height. The initial location of the search space centroid is $[0, 0, 0]^\intercal$m. The search space is mobile and moves to the east at a constant speed of 36m/s. A sequence of zero-mean white Gaussian process noise with power spectral density matrix $\text{diag}\,[S_x, S_y, S_z] = \text{diag}\,[0.1, 0.1, 0.01]$ is used to generate the search space centroid trajectory [19].

The UAVs track the search space centroid without scanning the cubes until the state estimate of the search space centroid is accurate enough. The eigenvalues of the estimated search space centroid position covariance are computed, which are the lengths of the uncertainty ellipsoid's semi-axes. A threshold is set up for the largest

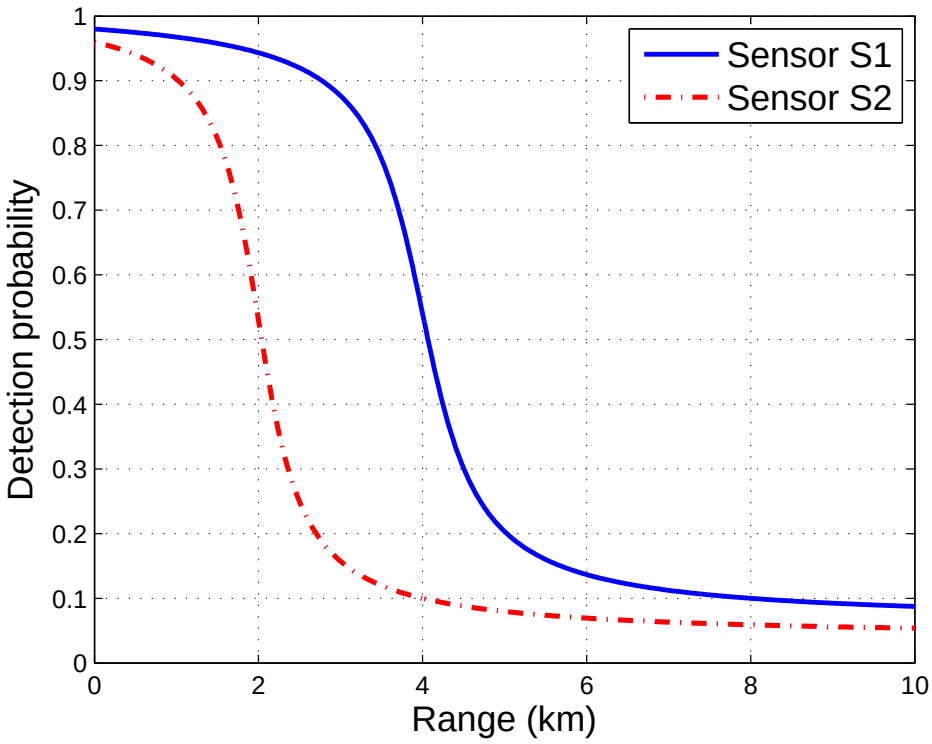


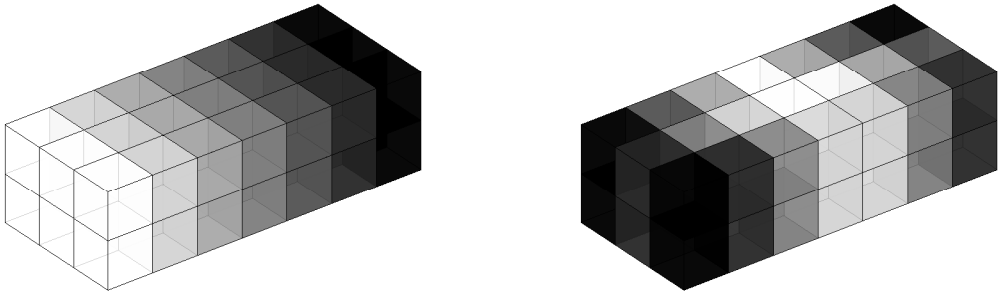
Figure 3.23: Detection probabilities assumed in the simulation.

eigenvalue among the three positional values. The UAVs start simultaneously tracking and scanning the search space once the largest eigenvalue is no greater than the threshold. In this simulation, the threshold is set to 25m. It is assumed that the approximate dimensional information about the mobile search space is known to the UAVs. From the perspective of the UAVs, taking into account the uncertainty in search area centroid estimation, the size of the estimated search space is generated based on the a priori rough knowledge of the search space dimension and the eigenvalues of the estimated search area centroid position covariance. In this simulation, the estimated 3D search space is a volume of 1050m in length, 450m in width and 300m in height, consisting of $7 \times 3 \times 2$ cubes.

Two search patterns are defined for the independent path planning and scanning in this 3D search-and-track scenario using a group of two UAVs: a 3D lawnmower-like search pattern (for short, 3D lawnmower) and a 3D spiral-like search pattern (for short, 3D spiral), as shown in Fig. 3.24. Cubes are scanned in the order of increasing shade intensity: In the 3D lawnmower search pattern, cubes are scanned from one side to the other; in the 3D spiral search pattern, cubes are scanned from the centroid to the sides.

In the following simulation, each cube inside the search space is scanned four times. One target, whose position w.r.t. the search space centroid is $[-400, 200, 100]^{\intercal}$m and does not change, exists within the space. The proposed JPPS solution is compared with the independent path planning and scanning solution methods based on the two pre-defined search patterns. The following results are averaged over 100 Monte Carlo runs.

The comparison of position PCRLB values of the search space centroid from the



(a) 3D Lawnmower (b) 3D Spiral

Figure 3.24: Scan priorities in the pre-defined 3D lawnmower search pattern and the pre-defined 3D spiral search pattern: Cubes are scanned in the order of increasing shade intensity.

start of cube scanning is shown in Fig. 3.25. It can be seen that there is not much difference between these three methods. Note that the search space centroid is being tracked for 44 time steps before scanning the search space, in order to achieve a low uncertainty region. During these 44 time steps, the PCRLB values are the same for different methods in each step and they decrease step after step, which are not shown in Fig. 3.25. The increase in PCRLBs in Fig. 3.25 is the result of UAVs adjusting their positions to avoid violating the sensor-to-sensor distance constraint.

The comparison of position PCRLB values of the targets detected by these three methods is shown in Fig. 3.26. It can be seen that the PCRLBs have similar values in the first 5 measurement steps, which is a result of the UAV kinematic constraints and the weighted-sum objective optimization: UAV states obtained by different path planning and scanning methods have the same positions and velocities before scanning the cubes and the difference in their positions and velocities will not be significant due to the UAV kinematic model within a few steps; the scan decision impacts the

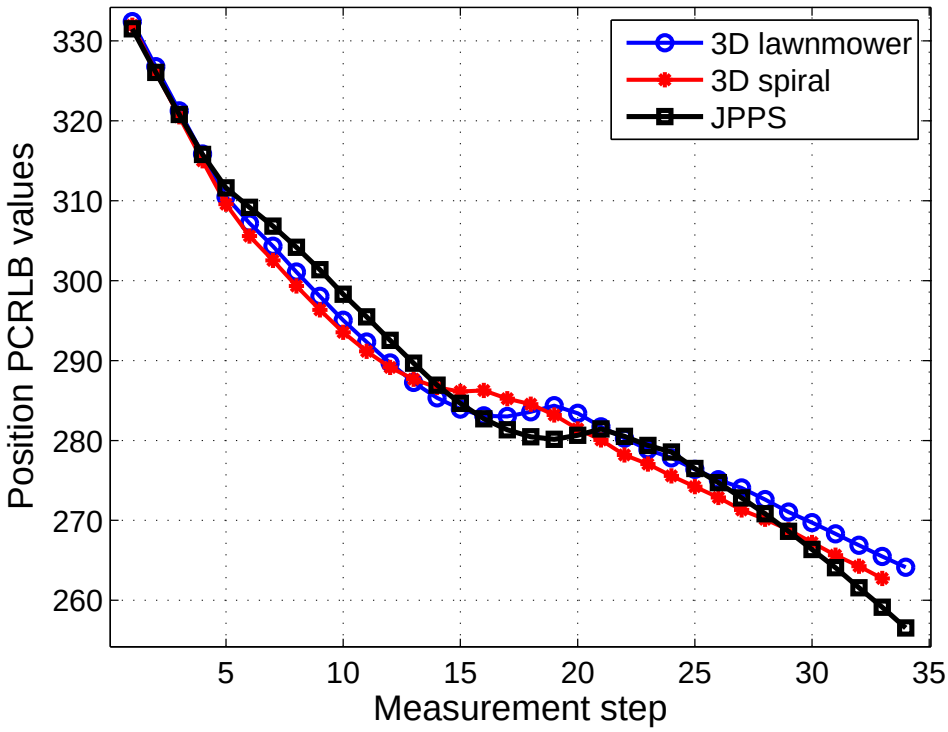


Figure 3.25: Position PCRLB values of the search space centroid using three different path planning and scanning methods.

overall performance of position PCRLB based on the corresponding weight value. The increase in PCRLBs in Fig. 3.26 is due to the same reason as that in Fig. 3.25. After the positional adjustment, UAVs using JPPS are able to continuously decrease the position PCRLB, whereas the decrease in position PCRLBs using independent path planning and scanning is not obvious.

In the following, one target, whose position w.r.t. the search space centroid is randomly generated in each Monte Carlo run, exists within the space. Once detected, the target is tracked for 10 steps by the UAVs. The comparison of position PCRLB values of the targets detected by these three methods is shown in Fig. 3.27 based on 100 Monte Carlo runs. The proposed method achieves a lower position PCRLB than

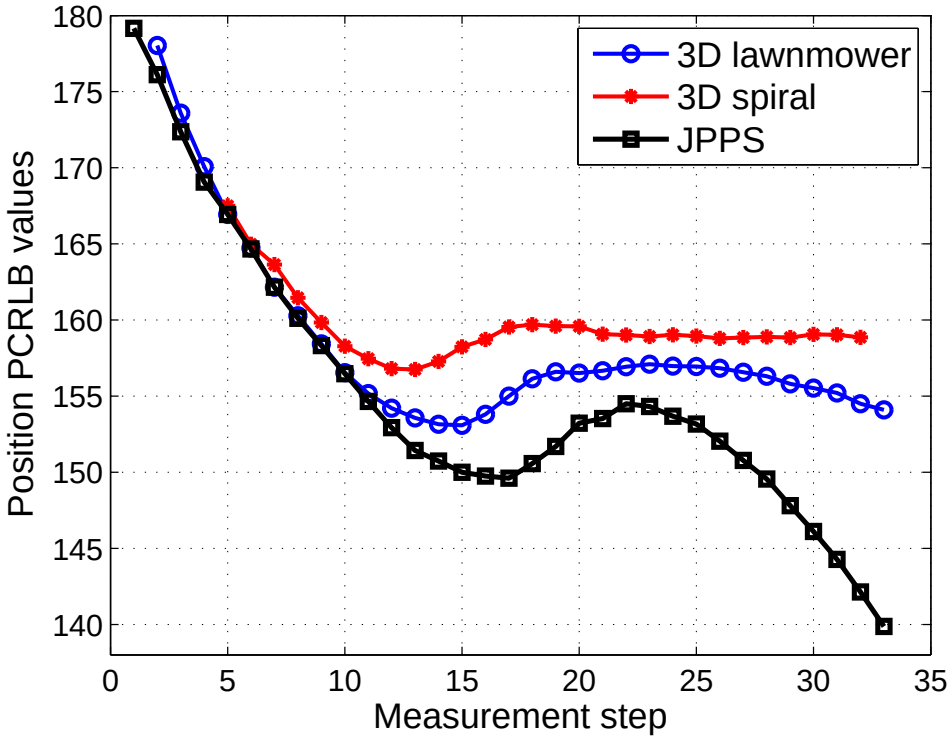


Figure 3.26: Position PCRLB values of the detected target using three different path planning and scanning methods, when the location of target w.r.t. the search space centroid is fixed in each Monte Carlo run.

the other two methods.

From the above simulations, we can see that the proposed JPPS solution has a better overall performance w.r.t. tracking the search space centroid, searching for possible targets within the space and tracking detected targets than the algorithms based on pre-defined search patterns.

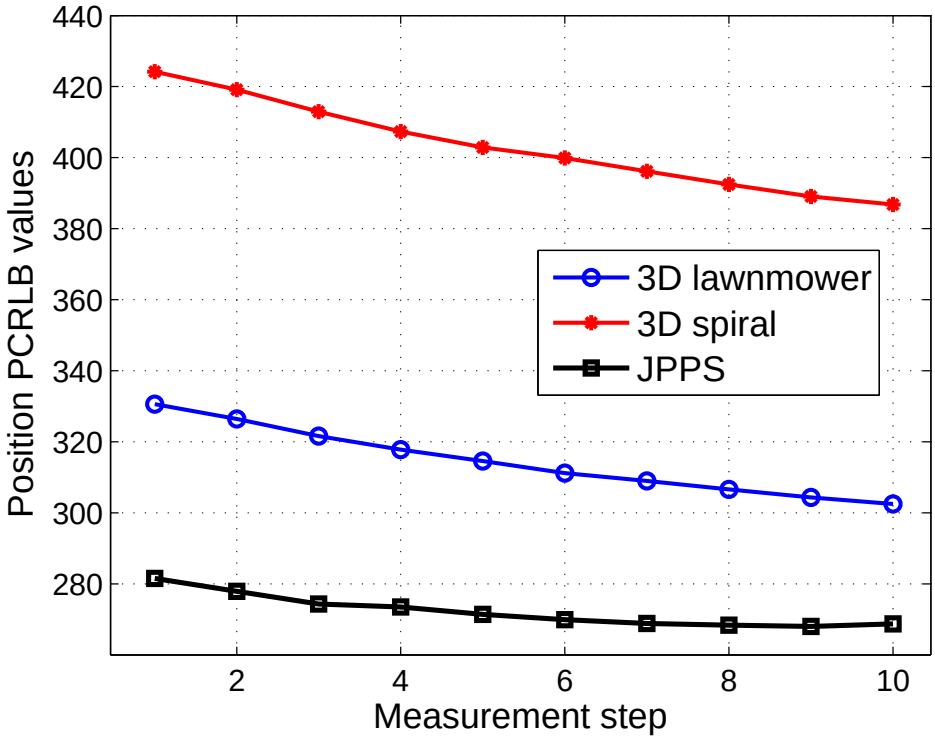


Figure 3.27: Position PCRLB values of the detected target using three different path planning and scanning methods, when the location of target w.r.t. the search space centroid is random in each Monte Carlo run.

3.8 Conclusions

In this paper, we proposed a joint path planning and scanning (JPPS) strategy based on genetic algorithm for a group of UAVs carrying out SAT over a 3D mobile space with a number of targets inside. A cooperative control framework was developed for UAVs to follow the mobile search space, track detected targets as well as search for new targets. Each UAV broadcasts its current scan, detection and tracking information as well as its current state, and then decides on path and cubes/targets to scan according to an information-based weighted-sum multiobjective function. The

UAV kinematic model, detection probabilities and the constraints on sensor altitude, sensor-to-sensor distance and sensor-to-target distance were considered in 3D path planning. The optimal sensor configuration for target localization using passive and active sensors in a 3D environment under constraints was provided with examples. In the simulation of path planning and scanning for an SAT mission, in terms of the PCRLB, the proposed JPPS method outperformed methods that make scan decisions and path selections independently.

Bibliography

[1] J. F. Araújo, P. B. Sujit, and J. B. Sousa, "Multiple UAV area decomposition and coverage," in *2013 IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA)*, Apr. 2013, pp. 30–37.

[2] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software.* John Wiley & Sons, 2004.

[3] P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, and A. Mahajan, "Mixed-integer nonlinear optimization," *Acta Numerica*, vol. 22, pp. 1–131, May 2013.

[4] L. F. Bertuccelli and J. P. How, "Robust UAV search for environments with imprecise probability maps," in *Proceedings of the 44th IEEE Conference on Decision and Control*, Dec. 2005, pp. 5680–5685.

[5] W. D. Blair, G. A. Watson, T. Kirubarajan, and Y. Bar-Shalom, "Benchmark for radar allocation and tracking in ECM," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 4, pp. 1097–1114, Oct. 1998.

[6] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004.

[7] R. H. Byrd, J. C. Gilbert, and J. Nocedal, "A trust region method based on interior point techniques for nonlinear programming," *Mathematical Programming*, vol. 89, no. 1, pp. 149–185, Nov. 2000.

[8] R. Chai, A. Savvaris, A. Tsourdos, S. Chai, and Y. Xia, "Optimal tracking guidance for aeroassisted spacecraft reconnaissance mission based on receding horizon control," *IEEE Transactions on Aerospace and Electronic Systems*, vol. PP, no. 99, pp. 1–1, Jan. 2018.

[9] K. Doğançay, "UAV path planning for passive emitter localization," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 2, pp. 1150–1166, Apr. 2012.

[10] S. Gallagher and R. Lanza, "Detection of nuclear weapons and fissile material abroad cargo containerships," Feb. 27 2007, US Patent 7,183,554.

[11] F. Gavilan, R. Vazquez, and E. F. Camacho, "An iterative model predictive control algorithm for UAV guidance," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 3, pp. 2406–2419, July 2015.

[12] M. L. Hernandez, A. Farina, and B. Ristic, "PCRLB for tracking in cluttered environments: measurement sequence conditioning approach," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 42, no. 2, pp. 680–704, Apr. 2006.

[13] M. L. Hernandez, T. Kirubarajan, and Y. Bar-Shalom, "Multisensor resource deployment using posterior Cramer-Rao bounds," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, no. 2, pp. 399–416, Apr. 2004.

[14] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. Cambridge, MA, USA: MIT press, 1992.

[15] A. Homaifar, C. X. Qi, and S. H. Lai, "Constrained optimization via genetic algorithms," *Simulation*, vol. 62, no. 4, pp. 242–253, Apr. 1994.

[16] J. Hu, L. Xie, J. Xu, and Z. Xu, "Multi-agent cooperative target search," *Sensors*, vol. 14, no. 6, pp. 9408–9428, May 2014.

[17] Y. Jin, Y. Liao, A. A. Minai, and M. M. Polycarpou, "Balancing search and target response in cooperative unmanned aerial vehicle (UAV) teams," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 36, no. 3, pp. 571–587, Jun. 2005.

[18] T. Kirubarajan and Y. Bar-Shalom, "Probabilistic data association techniques for target tracking in clutter," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 536–557, Mar. 2004.

[19] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. Part I. Dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, Oct. 2003.

[20] D. Mušicki, R. Evans, and S. Stankovic, "Integrated probabilistic data association (IPDA)," in *Proceedings of the 31st IEEE Conference on Decision and Control*, vol. 4, 1992, pp. 3796–3798.

[21] D. Mušicki, T. L. Song, H. H. Lee, X. Chen, and T. Kirubarajan, "Track-to-track fusion with target existence," *IET Radar, Sonar Navigation*, vol. 9, no. 3, pp. 241–248, Mar. 2015.

[22] N. H. Nguyen and K. Doğançay, "Optimal geometry analysis for multistatic TOA localization," *IEEE Transactions on Signal Processing*, vol. 64, no. 16, pp. 4180–4193, Aug. 2016.

[23] R. Pitre, X. Li, and R. Delbalzo, "UAV route planning for joint search and track missions—An information-value approach," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 3, pp. 2551–2565, July 2012.

[24] S. Ragi and E. K. P. Chong, "UAV path planning in a dynamic environment via partially observable Markov decision process," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 4, pp. 2397–2412, Oct. 2013.

[25] J. R. Riehl, G. E. Collins, and J. P. Hespanha, "Cooperative search by UAV teams: A model predictive approach using dynamic graphs," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 4, pp. 2637–2656, Oct. 2011.

[26] V. Roberge, M. Tarbouchi, and G. Labonte, "Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 132–141, Feb. 2013.

[27] A. Sinha, T. Kirubarajan, and Y. Bar-Shalom, "Autonomous surveillance by multiple cooperative UAVs," *Proc. SPIE*, vol. 5913, pp. 59 131V–59 131V–12, 2005.

[28] ——, "Autonomous search, tracking and classification by multiple cooperative UAVs," *Proc. SPIE*, vol. 6235, pp. 623 508–623 508–12, 2006.

[29] R. Tharmarasa, T. Kirubarajan, M. L. Hernandez, and A. Sinha, "PCRLB-based multisensor array management for multitarget tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 2, pp. 539–555, Apr. 2007.

[30] R. Tharmarasa, T. Kirubarajan, and T. Lang, "Joint path planning and sensor subset selection for multistatic sensor networks," in *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, July 2009, pp. 1–8.

[31] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, Mar. 2006.

[32] R. Waltz, J. Morales, J. Nocedal, and D. Orban, "An interior algorithm for nonlinear optimization that combines line search and trust region steps," *Mathematical Programming*, vol. 107, no. 3, pp. 391–408, July 2006.

[33] Y. Wang, T. Kirubarajan, R. Tharmarasa, R. Jassemi-Zargani, and N. Kashyap, "Multiperiod coverage path planning and scheduling for airborne surveillance," *IEEE Transactions on Aerospace and Electronic Systems*, vol. PP, no. 99, pp. 1–1, Mar. 2018.

[34] G. Wu, W. Pedrycz, H. Li, M. Ma, and J. Liu, "Coordinated planning of heterogeneous earth observation resources," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 1, pp. 109–125, Jan. 2016.

[35] S. Xu and K. Doğançay, "Optimal sensor placement for 3-D angle-of-arrival target localization," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 3, pp. 1196–1211, June 2017.

The following chapter is a reproduction of a ready to submit paper to IEEE Transactions on Aerospace and Electronic Systems:

Yinghui Wang, R. Tharmarasa, T. Kirubarajan, Anthony Damini, Martie Goulding Joint Path Planning and Radar Mode Scheduling for Multitarget Tracking in ECM, To be submitted to *IEEE Transactions on Aerospace and Electronic Systems*, August 2018.

Chapter 4

Joint Path Planning and Radar Mode Scheduling for Multitarget Tracking in ECM

4.1 Abstract

In this paper, a joint path planning and radar active-passive mode scheduling (JP-PAPS) strategy is developed in a three-dimensional (3D) contested environment using radar systems mounted on unmanned aerial vehicles (UAVs), which adopt active-passive mode switching against maneuvering targets in the presence of jammers broadcasting wide-band noise as electronic countermeasures (ECM). The radar systems on UAVs operate to track targets in the active mode or jammers in the passive mode. The estimated jammers' positions are used in active mode to suppress interference by spatial anti-jamming methods and further enhance the target tracking performance. A mixed-integer nonlinear programming in a receding horizon optimization based on

predictive posterior Cramér-Rao lower bound (PCRLB) is formulated to achieve accurate target state estimation, where the PCRLB values in the future steps for different radar modes are predicted and summed up. At each time step, UAV paths and radar operating modes are determined in the JPPAPS using a genetic algorithm to leverage the tracking of targets and jammers considering the possible target fire avoidance, UAV collision avoidance and hostile interference mitigation. Simulation results verify the superiority of the proposed algorithm in terms of the target tracking accuracy and track loss by comparing with independent path planning and active-passive mode scheduling strategies.

4.2 Introduction

In this paper, multitarget tracking (MTT) in the presence of electronic countermeasures (ECM) is studied using a group of unmanned aerial vehicle (UAVs) equipped with radar systems, which possess the capability to work either in active or passive mode. Moreover, the radar system is capable of simultaneously launching multiple beams to execute several radar tasks independently and generating nulls to reject interfering signals [36, 42]. Jammers are used as ECM to conceal protected targets while degrading and confusing radar systems [1]. In this contested environment, the radars on the UAVs should track the targets as accurately as possible by taking advantage of path planning and active-passive mode scheduling.

It is assumed that the jammers are within the radar detection range and can be tracked in passive mode. In other words, the jammer can be viewed as a target in passive mode even though it may not be as important as the actual targets under track. The estimated jammer state is used to predict the jammer's future position to

afford spatial interference suppression when the radars switch to active mode. The optimal active-passive mode scheduling is studied along with cooperative UAV path planning to minimize the estimation uncertainty corresponding to the targets in the presence of jammers. Path selection and radar mode are determined by solving a receding horizon optimization [5].

To use a phased array radar against maneuvering targets, the design and performance analysis of an interacting multiple model (IMM) estimator with an adaptive sampling policy, where large sampling intervals are allowed during benign periods and short sampling intervals are needed during maneuvering periods, were presented in [6]. The use of IMM with the probabilistic data association filter (PDAF) for the same problem in the presence of false alarm (FA) and ECM was presented in [20, 21], while countering the effects of target amplitude fluctuations, beamshape, missed detections, FAs and ECM, finite resolution, target maneuvers, and track loss. The ECM in the benchmark problem [4] includes a standoff jammer (SOJ) broadcasting wideband noise and targets attempting range gate pull off (RGPO).

Even though some of the target trajectories in our simulations are based on those in the benchmark problem [4] and an IMM estimator is used as in [6, 20], the optimization formulation in the joint path planning and active-passive mode scheduling (JP-PAPS) is the focus of our paper rather than the development of an advanced tracker or filter. Moreover, although beamforming and spatial anti-jamming techniques are considered to mitigate interference and to enhance the signal-to-interference-plus-noise ratio (SINR) when the radar works in active mode [8, 11, 22], they are mainly used for modeling the target detection probability. Radar signal processing [28] combined with sensor resource management [23] can be considered in some future work.

Considerable effort has been expanded on UAV path planning with application to target search-and-track systems [31, 33]. A number of optimal sensor placement and path planning algorithms are developed in two-dimensional (2D) space without extending to 3D scenarios [25, 26, 32]. The optimal geometries for time-of-arrival localization configuration is analyzed and the UAV path planning, which is based on minimizing the area of estimation confidence region, is simulated in [25]. Similarly, based on the Fisher information matrix (FIM), the UAV path planning for passive emitter localization is presented in [7], where angle-of-arrival, time-difference-of-arrival and scan-based localization are considered. A decentralized cooperative control algorithm, which maximizes the information gained from an environment and decides the next area to scan by predicting how much information can be gained by searching that area, is proposed for multiple cooperative UAVs to track detected targets as well as search for the undetected ones in [32]. An information-based objective function, which integrates the conflicting objectives of target detection, target tracking and vehicle survivability into a single scalar index, is presented for UAV path optimization in [26]. Constraints to the UAV dynamics, flyable region and no-fly zone are considered for the optimal path planning [7].

The Cramér-Rao lower bound (CRLB), which is the inverse of the FIM and provides a mean square error bound on the performance of any unbiased estimator of an unknown parameter vector, is frequently used in the optimization formulation for target localization [2, 12, 25, 40]. The posterior Cramér-Rao lower bound (PCRLB), which gives a measure of the achievable optimum performance and can be calculated predictively, is often used as the criterion in optimal sensor resource management problems with application to target tracking, such as optimal sensor selection, sensor

placement, power allocation and path planning [7, 13, 34, 39, 42]. Meanwhile, many path planning approaches for search-and-track missions use the value of gathered information as the objective function [26, 32].

These trackers available in the literature for target tracking, adaptive selection of sampling period, benchmark problems for MTT in the presence of ECM and optimal UAV path planning provide the motivation and serve as the foundation for our current study. To adapt to the development of sensor and sensor platform technology, joint UAV path planning and radar active-passive mode scheduling for intelligence, surveillance, target acquisition and reconnaissance missions are considered in this paper. The proposed JPPAPS is formulated as an optimization based on the predictive PCRLB using receding horizon control (RHC) [26, 34].

When the radars operate in passive mode, the jammers are tracked. The spatial signature of the jammers, e.g., the directions of arrival (DOAs), can be computed and used in the anti-jamming process to increase the SINR and the detection probability with the objective of enhancing the target tracking performance once the radars switch to active mode. However, the targets cannot be observed when the radars operate in passive mode since they do not emit any signals by themselves. When the radar is in active mode, the targets are tracked but the jammers are excluded due to beam-pointing. Therefore, there is a trade-off between radars operating in active mode versus in passive mode: in the presence of ECM, the estimation of jammer states in passive mode can enhance the target tracking results while radars operate in active mode, but the targets cannot be detected when radars are in passive mode leading to loss of track quality. Hence, one of the goals of this paper is to find the optimal solution to make decisions on radar operation mode considering the effect

of jammers on future steps. The radar mode scheduling is jointly considered with UAV path planning where the UAVs try to detect and track the targets or jammers cooperatively from different angles using different operation mode in each step.

The UAV path and radar working mode decision variables for the next H steps are considered in the objective function from different targets and jammers in future steps. The target detection probability is modeled considering the present of jammers within the targets' line of sight (LOS) and the accuracy of jammer state estimates to effectively implement spatial interference suppression. By switching between active and passive modes, the mode scheduling aims to optimize target tracking performance with the help of jammer state estimation to increase the detection probability. The UAV paths and radar modes in the predictive horizon that minimize the predictive PCRLB are selected as the decision variables. A genetic algorithm (GA) solution technique, which requires modest computational resources, is developed to generate the decisions for active vs. passive mode selection and platform path selection [14, 34]. The use of receding horizon control also enables the UAVs to have a long-term strategy for path planning, which improves collision avoidance and avoids possible target fire.

This paper makes the following contributions: An integrated optimization formulation is proposed for joint UAV path planning and radar active-passive mode scheduling. The predictive PCRLB using different radar modes given the prediction of the states of UAVs, targets and jammers in future time steps are computed and summed up. The estimation of the jammer states enables the radar system to suppress the jamming signal when the radars operate in active mode. The detection probability used in the information reduction factor (IRF) is modeled based on the estimated locations of the targets and jammers, the history of radar operating

modes and the history of track association results. The proposed JPPAPS adaptively switches between active and passive modes and designs the paths for UAVs according to the changes in the contested environment rather than iteratively switching between active and passive modes without the knowledge of the environment or switching to passive mode when the radar systems declare the existence of jammers. The proposed JPPAPS is solved using a genetic algorithm, which handles the mixed-integer nonlinear programming within the receding horizon optimization framework.

The remainder of the paper is structured as follows. Section 4.3 outlines the preliminary information needed to understand the system model including the decision variables, the criteria for the target to be corrupted by the jammer signal, the detection probability model, the UAV kinematic model and the formulation of the PCRLB. Section 4.4 presents the optimization formulation for the proposed JPPAPS strategy. The solution technique based on a genetic algorithm is developed in Section 4.5. The proposed JPPAPS strategy is compared with existing path planning and radar mode selection methods, which follow an ordered active-passive mode switching schemes and make decisions on UAV path and radar modes independently, through simulation in Section 4.6. Concluding remarks are given in Section 4.7.

4.3 Problem Description

Consider S UAVs tracking Q point targets in a 3D surveillance region with J jammers. A radar system, which possesses the capability to work in either active or passive mode, launch multiple beams pointing to different targets simultaneously and generate nulls to mitigate interference, is mounted on each UAV. The targets are tracked by the radar in active mode and the jammers are tracked in passive mode.

Set T as the measurement sampling interval for the radar system. Based on the position estimate of a target or a jammer, the DOA of the corresponding object, which can be used to steer a beam pointing to the target or generate a null to reject the interference, can be computed [8, 22, 37].

At time kT, denoted by time step k, the state vector of the qth ($q = 1, 2, ..., Q$) target is defined as $\mathbf{x}_{q,k}$, which consists of position $[x_{q,k}, y_{q,k}, z_{q,k}]^\intercal$, velocity $[\dot{x}_{q,k}, \dot{y}_{q,k}, \dot{z}_{q,k}]^\intercal$ and possibly acceleration $[\ddot{x}_{q,k}, \ddot{y}_{q,k}, \ddot{z}_{q,k}]^\intercal$, where $^\intercal$ denotes matrix transpose. The state vector of the jth ($j = 1, 2, ..., J$) jammer is expressed as

$$\mathbf{x}_{j,k} = [x_{j,k}, \dot{x}_{j,k}, \ddot{x}_{j,k}, y_{j,k}, \dot{y}_{j,k}, \ddot{y}_{j,k}, z_{j,k}, \dot{z}_{j,k}, \ddot{z}_{j,k}]^\intercal .$$

. The position of the sth ($s = 1, 2, ..., S$) UAV is $\mathbf{s}_{s,k} = [x_{s,k}, y_{s,k}, z_{s,k}]^\intercal$.

It is assumed that all radar systems on the UAVs at a time step work in the same mode. Let a binary variable u_k denote the decision on the radar mode at time step k. Then,

$$u_k = \begin{cases} 1 & \text{if radars operate in active mode at the } k\text{th step} \\ 0 & \text{if radars operate in passive mode at the } k\text{th step} \end{cases} \tag{4.1}$$

The decision variables on path planning will be discussed next with details on the UAV kinematic model used in this paper.

The following assumptions are made to simplify the problem.

Assumption 1: The number of targets and jammers is known from the radar search mode and the tracks are initialized by using the maximum likelihood probabilistic data association (ML-PDA) algorithm [16, 19, 39] or multi-frame detection methods [10,

17]. Then, filtering methods can be adopted to maintain target or jammer trajectories [21]. Based on the estimated state, the DOA of each target and jammer, which steers a beam to point at the target or generates a null to suppress the jamming signal, can be predicted and computed.

Assumption 2: The targets are widely separated in the surveillance region. In this case, MTT can be simplified as a number of simultaneous single target tracking problems. Even if there is association uncertainty due to target proximity, it does not limit the application of our framework provided seperate tracks are established. This is automatically done by standard MTT algorithms.

Assumption 3: To simplify the problem, as well as to make it general, radar equations, antenna gain patterns [4], adaptive beamforming [36] and interference mitigation techniques [11] are not included in our optimization formulation. Instead, the IRF for the single target case, which depends on the measurement noise covariance, the false alarm rate, the field of view of the sensor, and the probability of detection, is utilized in the computation of the predictive FIM [34].

Assumption 4: In real-world scenarios, the radar beamwidth increases as the beam is steered off the broadside direction. In this paper, for simplicity, the 3 dB beamwidth values at different angles are fixed when computing the probability of detection.

4.3.1 UAV Kinematic Model

The UAV path planning is constrained by the UAV kinematics capabilities. For example, in the horizontal plane, a fan-shaped flyable area can be formed for a UAV due to the constraints on its turning capability [18]. Different UAV kinematic models can be found in UAV path planning literature: in [29], it is assumed that UAVs can

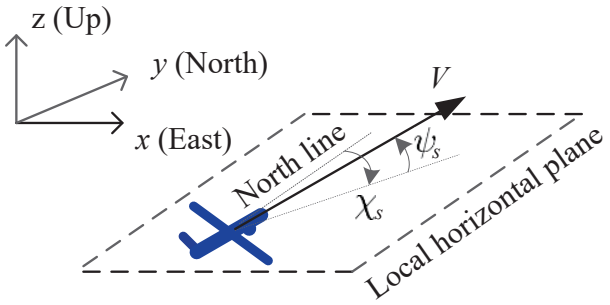


Figure 4.1: UAV kinematic model

move in any direction with unit velocity; in [15], the UAV motion model only deals with the waypoints of agents at discrete-time steps and the true dynamics of UAVs is not discussed; in [18], UAVs move along continuous trajectories with constant speed and constraints on turning; in [27], the altitude of a UAV is a constant and the control variables are the forward acceleration and the bank angle. Note that the above UAV kinematic models only consider motion in the 2D plane.

In this paper, the 3D discrete time airplane model from [9] is used for UAVs, as shown in Fig. 4.1. The UAV airspeed is assumed to be a constant V. The path planning decision variables for the sth UAV at time step k are the bank angle $\vartheta_{s,k}$ and the flight path angle $\psi_{s,k}$. The sth UAV heading angle at step k is expressed as $\chi_{s,k}$ and the location is $\mathbf{s}_{s,k} = [x_{s,k}, y_{s,k}, z_{s,k}]^\intercal$. The heading angle of the sth UAV at time step $k+1$ is

$$\chi_{s,k+1} = \frac{gT \tan(\vartheta_{s,k})}{V} + \chi_{s,k}, \tag{4.2}$$

where g is the acceleration due to gravity and T is the length of the time interval.

For formula compactness, let

$$\kappa_{s,k} = \frac{gT \tan(\vartheta_{s,k})}{V}. \tag{4.3}$$

The UAV waypoint update is modeled as [9]

$$\boldsymbol{s}_{s,k+1} = \boldsymbol{s}_{s,k} + \begin{bmatrix} \frac{VT}{\kappa_{s,k}} \cos(\psi_{s,k}) \left[\sin(\kappa_{s,k} + \chi_{s,k}) - \sin(\chi_{s,k})\right] \\ \frac{VT}{\kappa_{s,k}} \cos(\psi_{s,k}) \left[\cos(\chi_{s,k}) - \cos(\kappa_{s,k} + \chi_{s,k})\right] \\ VT \sin(\psi_{s,k}) \end{bmatrix}. \tag{4.4}$$

In this manner, a UAV's future state $[\boldsymbol{s}_{s,k+1}, \chi_{s,k+1}]^\intercal$ can be predicted based on its current state and control inputs $[\vartheta_{s,k}, \psi_{s,k}]^\intercal$.

The bank angle $\vartheta_{s,k}$ and the flight path angle $\psi_{s,k}$ satisfy the following constraints:

$$|\vartheta_{s,k}| \leq \Theta, \tag{4.5}$$

$$|\psi_{s,k}| \leq \Psi, \tag{4.6}$$

where Θ is the maximum bank angle and Ψ is the maximum flight path angle. Note that the rate of turn and radius of turn are constrained given (4.5).

The UAV state can be restricted by its motion capability. For example, the UAV may be constrained by a minimum altitude $h_{\min}$ and a maximum altitude $h_{\max}$ as

$$z_{s,k} \geq h_{\min}, \tag{4.7}$$

$$z_{s,k} \leq h_{\max}. \tag{4.8}$$

The range between the sth UAV and the qth target is defined as $D(\mathbf{s}_{s,k}, \mathbf{x}_{q,k}) = \|\mathbf{s}_{s,k} - \mathbf{x}_{q,k}\|$ and the distance between the sth UAV and the jth jammer is $D(\mathbf{s}_{s,k}, \mathbf{x}_{j,k}) = \|\mathbf{s}_{s,k} - \mathbf{x}_{j,k}\|$. A minimum UAV-to-target distance $d_{\mathrm{S}}^{\mathrm{Q}}$ is required for safety reasons,

for example, to avoid possible fire from the target. Similarly, a minimum UAV-to-jammer distance $d_{\mathrm{S}}^{\mathrm{J}}$ is defined to avoid strong jamming signal and possible fire from the jammers.

$$\begin{aligned} &D(\mathbf{s}_{s,k}, \mathbf{x}_{q,k}) \geq d_{\mathrm{S}}^{\mathrm{Q}}, \\ &\forall s = 1, 2, ..., S,\ q = 1, 2, ..., Q,\ k = 1, 2, ... \end{aligned} \tag{4.9}$$

$$\begin{aligned} &D(\mathbf{s}_{s,k}, \mathbf{x}_{j,k}) \geq d_{\mathrm{S}}^{\mathrm{J}}, \\ &\forall s = 1, 2, ..., S,\ j = 1, 2, ..., J,\ k = 1, 2, ... \end{aligned} \tag{4.10}$$

A minimum UAV-to-UAV distance $d_{\mathrm{SS}}^{\min}$ is placed for collision avoidance.

$$\begin{aligned} &D(\mathbf{s}_{s,k}, \mathbf{s}_{s',k}) \geq d_{\mathrm{SS}}^{\min}, \\ &s, s' \in \{1, 2, ..., S\},\ s \neq s',\ k = 1, 2, ... \end{aligned} \tag{4.11}$$

A maximum UAV-to-UAV distance $d_{\mathrm{SS}}^{\max}$ is enforced for communication connectivity.

$$\begin{aligned} &D(\mathbf{s}_{s,k}, \mathbf{s}_{s',k}) \leq d_{\mathrm{SS}}^{\max}, \\ &s, s' \in \{1, 2, ..., S\},\ s \neq s',\ k = 1, 2, ... \end{aligned} \tag{4.12}$$

The above constraints can be used alone or in combination.

4.3.2 Criteria for Target Corruption

Since multiple beams are launched simultaneously to observe the targets and nulls are generated to reject interfering signals [36, 42] by the radar system on a UAV, a target

return is not always corrupted in the presence of jammers and is only corrupted under the following conditions. Two angular thresholds ρ_1 and ρ_2 are used based on the beamwidth of the radar and the shape of the null generated by the spatial anti-jamming method. It is assumed that the radar beamwidth is $2\rho_1$. When any jammer is out of ρ_1 of a radar-to-target LOS, the corresponding target return will not be corrupted. When there is a jammer within ρ_1 of a radar-to-target LOS, the corresponding target return is corrupted or hidden by the jammer signal. However, with the implementation of spatial interference suppression methods given accurate DOAs of the jammers, jammer signals can be nulled. When there is a jammer within ρ_2 of a radar-to-target LOS, the spatial null generated by a spatial anti-jamming method will also filter out the return signal from targets. Therefore, spatial anti-jamming methods cannot make up for the detection probability decrease and track loss caused by jammers.

In the simulations, the impact of jamming on target detection probability is observed. Based on the above signal corruption criteria, the detection probability for the sth UAV measuring the qth target at time step k, $P_d(s,q,k)$, in this contested environment is a function of the following three parts: 1) the range between the UAV and the target $D(\mathbf{s}_{s,k}, \mathbf{x}_{q,k})$, 2) the angles between the target's LOS and all jammers' LOS, $\{\theta(\mathbf{x}_{q,k}, \mathbf{s}_{s,k}, \mathbf{x}_{j,k}),\ \forall j = 1,2,...,J\}$, and 3) the angle between the estimated jammer's LOS and the true jammer's LOS, $\theta(\mathbf{x}_{j,k}, \mathbf{s}_{s,k}, \hat{\mathbf{x}}_{j,k})$, if this jammer is within ρ_1 of the target LOS.

When $\{\theta(\mathbf{x}_{q,k}, \mathbf{s}_{s,k}, \mathbf{x}_{j,k}) < \rho_1,\ \exists j = 1,2,...,J\}$, the target return is corrupted and the detection probability decreases with the angle between the target LOS and the jammer LOS if any spatial anti-jamming process is not implemented.

When $\{\theta(\mathbf{x}_{q,k}, \mathbf{s}_{s,k}, \mathbf{x}_{j,k}) < \rho_2,\ \exists j = 1, 2, ..., J\}$, the detection probability is fixed to a small value. When $\rho_1 \leq \{\theta(\mathbf{x}_{q,k}, \mathbf{s}_{s,k}, \mathbf{x}_{j,k}),\ \exists j = 1, 2, ..., J\} < \rho_2$, with the spatial anti-jamming technique in active mode, the detection probability is incremented to make up for the loss in the detection probability due to the interference. The smaller the value $\theta(\mathbf{x}_{j,k}, \mathbf{s}_{s,k}, \hat{\mathbf{x}}_{j,k})$, the greater the increment value corresponding to $P_d(s, q, k)$ if the jth jammer is the one that within ρ_1 but out of ρ_2 of the target LOS. More details on the detection probability model used for target tracking will be described in Section 4.6.

4.3.3 FIM and PCRLB

In this subsection, only one radar on a UAV is considered at first and the UAV index "s" is omitted. The FIM for more than one UAV with respect to a target is the summation of the FIM values corresponding to all UAVs. Since the targets (and jammers) are treated separately, for brevity, the target's index "q" (and jammer's index "j") are omitted often in this and the next sections, unless doing so causes confusion.

Let $\mathbf{x}_k$ be an unknown and random state vector. The global measurement vector is denoted by $\mathbf{z}_k$. If we assume that targets are moving independently and the state equation of each target is linear, then the overall state equation is given by

$$\mathbf{x}_{k+1} = \mathbf{F}_k \mathbf{x}_k + \mathbf{v}_k, \tag{4.13}$$

where $\mathbf{F}_k$ is the state transition matrix and $\mathbf{v}_k$ is the process noise of the target. The covariance matrix of $\mathbf{v}_k$ is $\boldsymbol{\Gamma}_k$.

The measurement is given by

$$\mathbf{z}_k = \begin{cases} h_k(\mathbf{x}_k) + \boldsymbol{\omega}_k & \text{if measurement originates from target} \\ \boldsymbol{\upsilon}_k & \text{if false alarm} \end{cases} \tag{4.14}$$

where $h_k(\cdot)$ is (in general) a nonlinear function, $\boldsymbol{\omega}_k$ is a zero-mean Gaussian random variable with covariance $\boldsymbol{\Sigma}_k$ and $\boldsymbol{\upsilon}_k$ is uniformly distributed across the surveillance region. If the radar is in active mode, the measurements are usually the range, the elevation angle and the bearing angle. If the radar is in passive mode, the measurements are the elevation angle and the bearing angle [4].

The PCRLB gives a measure for the achievable optimum performance in terms of mean square error and, importantly, this bound can be calculated predictively [34, 13]. Furthermore, the PCRLB is independent of the filtering algorithm employed and is therefore not constrained by the idiosyncrasies of any particular filtering methodology. Let $\hat{\mathbf{x}}_k$ be an unbiased estimate of $\mathbf{x}_k$ based on the measurement data $\mathbf{z}_k$. The PCRLB, which is defined to be the inverse of the FIM $\mathbf{J}(\mathbf{x}_k)$, gives a lower bound of the error covariance matrix, i.e.,

$$\mathbf{C}_k = \mathbb{E}\left\{(\hat{\mathbf{x}}_k - \mathbf{x}_k)(\hat{\mathbf{x}}_k - \mathbf{x}_k)^{\intercal}\right\} \succeq \mathbf{J}^{-1}(\mathbf{x}_k), \tag{4.15}$$

where $\mathbb{E}$ denotes the expectation operator. The inequality in (4.15) means that $\mathbf{C}_k - \mathbf{J}^{-1}(\mathbf{x}_k)$ is a positive semi-definite matrix .

A recursive formula for the evaluation of the posterior FIM $\mathbf{J}(\mathbf{x}_k)$ is given by [34]

$$\mathbf{J}(\mathbf{x}_k) = \mathbf{J}_X(\mathbf{x}_k) + \mathbf{J}_Z(\mathbf{x}_k), \tag{4.16}$$

where

$$\mathbf{J}_X(\mathbf{x}_k) = \left[\boldsymbol{\Gamma}_{k-1} + \mathbf{F}_{k-1}\mathbf{J}(\mathbf{x}_{k-1})\mathbf{F}_{k-1}^{\intercal}\right]^{-1} \tag{4.17}$$

gives the prior information regarding the target states at time step k. The measurement contribution is given by

$$\mathbf{J}_Z(\mathbf{x}_k) = \mathbb{E}\left\{r_k \mathbf{H}_k^{\intercal} \boldsymbol{\Sigma}_k^{-1} \mathbf{H}_k\right\}, \tag{4.18}$$

where r_k is the IRF, $H_k(\alpha, \beta) = \frac{\partial h_k(\alpha, \mathbf{x}_k)}{\partial \mathbf{x}_k(\beta)}$ denotes the (α, β)th element of matrix $\mathbf{H}_k$. Here, we only present the PCRLB for a single sensor and widely-separated targets; for the general case (i.e., multiple sensors with measurement origin uncertainty), see [34].

The trace of the PCRLB, which is denoted as $\mathrm{tr}(\mathbf{J}^{-1}(\mathbf{x}_k))$, is used here as the scalar performance metric [7, 40]. Note that either the determinant or the maximum eigenvalue of the PCRLB can also be used to assess the performance [25, 34].

The FIM for S UAVs corresponding to the qth target is $\mathbf{J}(\mathbf{x}_{q,k}) = \sum_{s=1}^{S} \mathbf{J}(\mathbf{s}_{s,k}, \mathbf{x}_{q,k})$. The trace of the PCRLB is denoted as $\mathrm{tr}(\mathbf{J}^{-1}(\mathbf{x}_{q,k}))$.

4.4 Optimization Formulation

4.4.1 Objective Function

For all UAVs, by stacking the trace values of the PCRLB regarding all the targets, a vector is defined as follows:

$$\mathbf{G}_{Q,k} = \left[\mathrm{tr}(\mathbf{J}^{-1}(\mathbf{x}_{1,k})), \mathrm{tr}(\mathbf{J}^{-1}(\mathbf{x}_{2,k})), ..., \mathrm{tr}(\mathbf{J}^{-1}(\mathbf{x}_{Q,k}))\right]^{\intercal} \tag{4.19}$$

The vectors of the trace values of the predicted PCRLB for targets at the next time step are denoted by $\tilde{\mathbf{G}}_{Q,k+1}$, in which the states of the UAVs, targets and jammers in (4.16) to (4.19) are the predicted states based on their current states and their motion model.

If the radars are in active mode at time step k, i.e., $u_k = 1$, the summation of the trace of the PCRLB is $\mathbf{1}_{Q\times 1}^{\mathsf{T}}\mathbf{G}_{Q,k}$. Based on the current UAV states and the predicted target and jammer states, decisions on path planning and radar active-passive mode selection are made to minimize the summation of the trace of predicted PCRLB from the targets for the H next steps.

In the JPPAPS, the first radar mode decision variable u_{k+1} at time step k can be either 1 or 0. When $u_{k+1} = 1$, $u_{k+H} = 0$; when $u_{k+1} = 0$, $u_{k+H} = 1$. The radar mode decision variables for time steps $k+2$ to $k+H-1$ are all one. The choice of $u_{k+1} = 1$ or 0 is because in the RHC although the path and radar mode decisions for the next H steps are made, only the decision variables for the next step are used. The choice of $u_{k+H} = 0$ or 1 is made to guarantee that the number of items (the trace of the PCRLB) in the summation is the same for different active-passive mode scheduling at step k.

The radar systems estimate the detection probability based on the logic-based track management. If M_c measurements are associated with a track out of the last N_c measurement frames, the detection probability used in the IRF is M_c/N_c. Meanwhile, it is assumed that once the radar systems start operating in passive mode, the probability of detection in IRF in the next L steps will increase based on the detection probability value at time step k, where $L \leq H$. Based on the locations of the UAVs, the targets and the jammers, $\theta(\hat{\mathbf{x}}_{q,k+h}, \mathbf{s}_{s,k+h}, \hat{\mathbf{x}}_{j,k+h})$ is predicted, where

$1 \leq h \leq H$. When $\theta(\hat{\mathbf{x}}_{q,k+h}, \mathbf{s}_{s,k+h}, \hat{\mathbf{x}}_{j,k+h}) < \rho_2$, the detection probability in the IRF will not increase once the radar systems start operating in passive mode.

At the $(k+h)$th time step, the predicted PCRLB of the targets is denoted as

$$U_{k+h} = u_{k+h} \mathbf{1}_{Q\times 1}^{\intercal} \tilde{\mathbf{G}}_{Q,k+h}, \tag{4.20}$$

where $h = 1, 2, ..., H$. A receding horizon decay factor a_{k+h} is defined for each future step. The vector of the receding horizon decay factors is defined as $\mathbf{A}_k = [a_{k+1}, a_{k+2}, ..., a_{k+H}]^{\intercal}$. The summation of target predictive PCRLB in the receding horizon is

$$\sum_{h=1}^{H} a_{k+h} U_{k+h} = \mathbf{A}_k^{\intercal} \mathbf{U}_k, \tag{4.21}$$

where $\mathbf{U}_k = [U_{k+1}, U_{k+2}, ..., U_{k+H}]^{\intercal}$ is a vector of predicted target PCRLB at different time steps.

Therefore, the objective function is expressed as

$$\min \ \mathbf{A}_k^{\intercal} \mathbf{U}_k, \tag{4.22}$$

where the decision variables are ϑ_{k+h}, ψ_{k+h} and u_{k+1} for $h = 1, ..., H$.

The number of decision variables in this receding horizon control framework is $(2SH + 1)$, where 2 stands for the bank angle and flight path angle, and 1 represents the radar working mode at the future time step $k+1$. It can be seen that the greater the value H, the more of the computational resources needed to solve the problem.

When the value of H is large, the predicted states of targets and jammers will be inaccurate and the predictive PCRLB will not contribute much to the objective value considering the RHC decay factor. However, the advantages in planning for the

next H steps rather than the next one step alone are as follows: 1) Once the radars enter in passive mode, the increment in the detection probability corresponding to a corrupted target by the spatial anti-jamming process can be implemented in the following L active modes, which decreases the value of predicted PCRLB. 2) The constraints on UAV altitude and UAV no-fly zone will be hard to violate because the UAVs have a long planning horizon. Details on the handling of constraints are described in the next subsection.

4.4.2 Constraints on Path Planning

The path planning variables, which are the bank angles and the flight path angles, should satisfy constraints (4.5) and (4.6). Constraints (4.7), (4.8), (4.9), and (4.10) are transformed into soft constraints and are added to the objective function (4.22).

For the sth UAV, a penalty function is defined for the violation of the altitude constraints (4.7) and (4.8) as follows:

$$p_1(s,k) = w_1 \left[\max\left(z_{s,k} - \alpha_1 h_{\max}, 0\right) + \max\left(\alpha_2 h_{\min} - z_{s,k}, 0\right)\right], \tag{4.23}$$

where $0 < \alpha_1 < 1$, $\alpha_2 > 1$ and $\alpha_1 h_{\max} > \alpha_2 h_{\min}$. The use of α_1 and α_2 makes the altitude constraints more strict. The violation of $z_{s,k} \geq \alpha_2 h_{\min}$ and $z_{s,k} \leq \alpha_1 h_{\max}$ will firstly happen at large h values in the RHC for $1 \geq h \geq H$, which will influence the path selection for time step $k+1$. Incorporation of (4.23) will decrease the possibility of the violation of constraints (4.7) and (4.8) if α_1 and α_2 are appropriately selected based on the UAV kinematic model. The weight w_1 is used to balance the value of the penalty and the value of the predicted PCRLB to prevent one of these two components becoming dominant. The more severe the constraint violation, the higher the penalty

value. The value of (4.23) is positive and the function itself can be either linear or nonlinear.

For the sth UAV and the qth target, a penalty function is defined for the violation of the UAV-to-target range constraint (4.9) as follows:

$$p_2(s,q,k) = w_2 \max\left(\alpha_3 d_{\mathrm{S}}^{\mathrm{Q}} - D(\mathbf{s}_{s,k}, \mathbf{x}_{q,k}), 0\right), \tag{4.24}$$

Similarly, for the sth UAV and the jth jammer, a penalty function is defined for the violation of the UAV-to-jammer range constraints (4.10) as follows:

$$p_3(s,j,k) = w_3 \max\left(\alpha_4 d_{\mathrm{S}}^{\mathrm{J}} - D(\mathbf{s}_{s,k}, \mathbf{x}_{j,k})\right), \tag{4.25}$$

where the weights w_2 and w_3 as well as parameters α_3 and α_4 are selected as in (4.23) with $\alpha_3 > 1$, $\alpha_4 > 1$. The more severe the constraint violation, the higher the penalty value.

For UAVs s and s', the penalty function corresponding to constraint (4.11) is expressed as

$$p_4(s,s',k) = w_4 \max\left(\alpha_5 d_{\mathrm{SS}}^{\min} - D(\mathbf{s}_{s,k}, \mathbf{s}_{s',k}), 0\right), \tag{4.26}$$

For UAVs s and s', the penalty function corresponding to constraint (4.12) is expressed as

$$p_5(s,s',k) = w_5 \max\left(D(\mathbf{s}_{s,k}, \mathbf{s}_{s',k}) - \alpha_6 d_{\mathrm{SS}}^{\max}, 0\right), \tag{4.27}$$

where $\alpha_5 > 1$ and $0 < \alpha_6 < 1$, $\alpha_6 d_{\mathrm{SS}}^{\max} > \alpha_5 d_{\mathrm{SS}}^{\min}$.

The summation of penalty values at time step k is given by

$$\begin{aligned} P(k) = & \sum_{s=1}^{S} p_1(s,k) + \sum_{s=1}^{S}\sum_{q=1}^{Q} p_2(s,q,k) \\ & + \sum_{s=1}^{S}\sum_{j=1}^{J} p_3(s,j,k) + \sum_{s=1}^{S}\sum_{\substack{s'=1 \\ s'\neq s}}^{S} p_4(s,s',k) \\ & + \sum_{s=1}^{S}\sum_{\substack{s'=1 \\ s'\neq s}}^{S} p_5(s,s',k). \end{aligned} \tag{4.28}$$

A vector of penalty values during the predictive horizon is defined based on the predicted states of the UAVs, targets and jammers as

$$\mathbf{P}_k = [P(k+1), P(k+2), ..., P(k+H)]^{\mathsf{T}}. \tag{4.29}$$

4.4.3 Optimization Formulation

To sum up, at time step k, the paths and operating mode for the next step are planned by predicting and computing the summation of predicted PCRLB values for the next H steps considering the penalty in (4.29). The decision variables are ϑ_{k+h}, ψ_{k+h} and

u_{k+1}, where $h = 1, 2, ..., H$.

$$
\begin{aligned}
&\min\ \mathbf{A}_k^{\intercal}(\mathbf{U}_k + \mathbf{P}_k), \\
&\text{s.t. } |\vartheta_{k+h}| \leq \Theta \\
&\quad |\psi_{k+h}| \leq \Psi \\
&\quad u_{k+h} = \begin{cases} \{0, 1\}, & \text{if } h = 1 \text{ or } H \\ 1, & \text{otherwise} \end{cases} \\
&\quad u_{k+1} = 1 - u_{k+H} \\
&\quad \forall h = 1, 2, ...H
\end{aligned}
\tag{4.30}
$$

4.5 Solution Technique

From Section 4.4 we can see that the path control inputs are continuous and the active-passive mode scheduling variables are discrete. Therefore, (4.30) is a mixed-integer nonlinear programming (MINLP) [3]. MINLP includes both nonlinear programming (NLP) and mixed-integer linear programming (MILP) as subproblems, which is an NP-hard combinatorial problem and its solution typically requires searching enormous search trees [3]. Details on MINLP models and solution methods can be found in [3].

Since the proposed JPPAPS must be solved in real time for online path planning and active-passive mode scheduling, it is desirable to obtain a near optimal solution quickly rather than to wait for the global optimal solution. In this paper, a genetic algorithm based cooperative control is developed. Note that other heuristics for solving MINLPs, such as particle swarm optimization (PSO) [30], tabu search (TS) [38], and simulated annealing (SA) [3], can also be used.

The use of the genetic algorithm to solve this MINLP is inspired by [35]. The genetic algorithm is a class of learning algorithms based on a parallel search for an optimal solution [14]. The parallel searches, which are performed synchronously in time steps, are called generations. In each generation, a certain number of radar mode selection and path decision options called individuals are maintained. The whole set of individuals in a generation is referred to as the population. The main idea in the genetic algorithm is to preserve and create variations of individuals that seem most promising (with high value of fitness) and remove the others. The number of generations is denoted as N_{gen} and the population size as N_{pop}.

The individual (also known as the chromosome), which presents the decision variables in the proposed problem, is formed as follows: the first $2SH$ variables represent the UAVs' bank angles and flight path angles for the next H steps; the next variable represents the radar mode for time step $k+1$, which takes 1 if the radars are in active mode and 0 otherwise. Based on u_{k+1}, the remaining radar mode decision variables for future steps are know according to (4.30).

In order to handle the continuous values in the genetic algorithm, the bank angles and flight path angles are mapped to continuous values between 0 and 1 as follows:

$$\bar{\vartheta}_k = \frac{\vartheta_k + \Theta}{2\Theta}, \text{ and } \bar{\psi}_k = \frac{\psi_k + \Psi}{2\Psi}, \tag{4.31}$$

$$0 \leq \bar{\vartheta}_k, \bar{\psi}_k \leq 1. \tag{4.32}$$

Therefore, constraints (4.5) and (4.6) are enforced while generating individuals. These continuous variables are divided into evenly spaced numbers and randomly selected when generating the chromosome. They are converted back to real values of angles

when evaluating the fitness function and when the GA terminates.

The fitness function, which maximizes the negative value of the objective in (4.30), is evaluated in each time step.

The decisioning of parent to reproduce is based on the fitness of individuals. Individuals with high fitness values are most likely to be selected, whereas those with low fitness will be discarded. Many selection techniques, such as tournament selection and roulette wheel selection, are available [14]. A binary tournament selection is used in this paper.

In the crossover operation, a locus is randomly chosen and the subsequences before and after that locus between two chromosomes (parents) are exchanged to create two new individuals (offsprings). Crossover is normally performed with a probability P_c, i.e., a fraction P_c of the new individuals are formed by crossover and the remaining fraction $1 - P_c$ are copied.

In the mutation operation, each variable of the new individual is changed with a certain probability P_m; binary variables (i.e., radar mode decision) are flipped and continuous variables (i.e., path selection) are replaced by a random value between 0 and 1.

In each generation, the individuals are ranked by sorting their fitness values. A small proportion of the fittest individuals are copied into the next generation. These top scored individuals that are preserved unchanged through elitism remain eligible for selection as parents when breeding the remainder of the next generation. Elitism prevents losing the few best-found solutions and may improve the performance of the GA [35].

In this paper, a cooperative control is used for the UAVs. Each UAV performs the

following tasks at each time step. The UAV uses its radar to scan according to the radar mode decision made in the previous step, predicts and updates the state and state covariance of the targets and jammers, and transmits tracks for track-to-track fusion [24]. The fusion outputs of the targets and jammers are shared and used in decision making for the next step. Based on the state estimate of the targets and the jammers, as well as the models used in the IMM and mode transition probabilities, the locations of the targets and the jammers for the next H steps are predicted [20]. The path planning and active-passive mode selection for the next time step is made by solving (4.30) using the genetic algorithm.

4.6 Simulations

Note that the benchmark tracking and sensor management problem in [4] was proposed for a ground-based phased array radar. In our paper, the simulations consist of using two UAVs to track a single target as well as multiple maneuvering targets in the presence of jammers. The jammers are within the radar range and can be tracked in passive radar mode. Some of the targets' and jammers' trajectories in the benchmark [4] are adjusted for the airborne radar tracking scenario. An IMM estimator with a logic-based track management is used to track targets and jammers [20].

The sampling time is 1s. The maximum bank angle of the UAV is 30°/s, the maximum flight path angle is 15°/s, and the speed is 150m/s. In terms of the radar field-of-view (FOV), the target ranges vary from 0.2km to 120km, while the target elevation angle can vary from $-80°$ to $80°$ for both active and passive modes. Since only one face of the phased array radar is used, the bearing of the target is confined to $\pm 60°$ for both active and passive modes. For the proposed JPPAPS, the objective

function based on RHC plans for the next 3 steps and the decision variables for the next step are used for path and radar mode selection. The value of L is set to 3 when predicting the detection probability in the IRF of the receding horizon optimization. The minimum distance between UAVs is set to be 200m. The minimum distance between a UAV and a target is set to be 500m. The minimum distance between a UAV and a jammer is set to be 600m. The minimum altitude for UAVs is 200m and the maximum altitude is 6000m. The minimum distance between UAVs is 200m and the maximum distance between UAVs is 10km. In the penalty functions, $\alpha_1 = \alpha_6 = 0.8$ and $\alpha_2 = \alpha_3 = \alpha_4 = \alpha_5 = 1.25$. These parameters are the same for all UAVs.

4.6.1 Single Target Tracking with Two UAVs and Two Jammers

The initial locations of the UAVs are $[-6000, 4000, 500]^\intercal$m and $[-6000, -4000, 500]^\intercal$m respectively, and the initial headings of the UAVs are both $[1, 0, 0]^\intercal$. One target at $[0, 0, 2000]^\intercal$m moves to the east with a constant speed 150m/s. One jammer flies with an oval (race course) holding pattern in the clockwise direction with an initial position of $[26000, -3000, 3050]^\intercal$m and speed of 168m/s [4]. Similarly, the second jammer flies with an oval holding pattern in the counter-clockwise direction at an initial position of $[26000, 3000, 3050]^\intercal$m and speed of 168m/s.

It is assumed that the radar beamwidth is 12°, $\rho_1 = 6^\circ$ and $\rho_2 = 2^\circ$.

In the detection probability model, when $\{\theta(\mathbf{x}_{q,k}, \mathbf{s}_{s,k}, \mathbf{x}_{j,k}) < 6^\circ,\ \nexists j = 1, 2, ..., J\}$,

the qth target is not corrupted by jammers, we assume that

$$\begin{aligned} P_d(s,q,k) =& f_1(D(\mathbf{s}_{s,k},\mathbf{x}_{q,k})) \\ =& \begin{cases} 0.1 & \text{if } D(\mathbf{s}_{s,k},\mathbf{x}_{q,k}) > 2\times 10^4 \\ 0.9 & \text{if } D(\mathbf{s}_{s,k},\mathbf{x}_{q,k}) < 2.22\times 10^3 \\ 1-\varepsilon D(\mathbf{s}_{s,k},\mathbf{x}_{q,k}) & \text{otherwise} \end{cases} \end{aligned} \tag{4.33}$$

where $\varepsilon = 4.5\times 10^{-5}$.

When $\{2^\circ < \theta(\mathbf{x}_{q,k},\mathbf{s}_{s,k},\mathbf{x}_{j,k}) < 6^\circ,\ \exists j = 1,2,...,J\}$, the qth target return is corrupted or hidden by the jammer signal. Assume that the set of jammers that satisfies $\{2^\circ < \theta(\mathbf{x}_{q,k},\mathbf{s}_{s,k},\mathbf{x}_{j,k}) < 6^\circ\}$ is $J_1(k)$. If spatial anti-jamming technique is not implemented, we have

$$\begin{aligned} P_d(s,q,k) &= f_2\left(D(\mathbf{s}_{s,k},\mathbf{x}_{q,k}), \{\theta(\mathbf{x}_{q,k},\mathbf{s}_{s,k},\mathbf{x}_{j,k}),\ j\in J_1(k)\}\right) \\ &= f_1(D(\mathbf{s}_{s,k},\mathbf{x}_{q,k})) - (f_1(D(\mathbf{s}_{s,k},\mathbf{x}_{q,k})) - 0.02) \\ &\quad \times \frac{6-\min\{\theta(\mathbf{x}_{q,k},\mathbf{s}_{s,k},\mathbf{x}_{j,k}),\ j\in J_1(k)\}}{4}. \end{aligned} \tag{4.34}$$

When $\{2^\circ < \theta(\mathbf{x}_{q,k},\mathbf{s}_{s,k},\mathbf{x}_{j,k}) < 6^\circ,\ \exists j = 1,2,...,J\}$ and the spatial anti-jamming

technique is implemented, we have

$$
\begin{aligned}
P_d(s,q,k) &= f_3(D(\mathbf{s}_{s,k},\mathbf{x}_{q,k}),\{\theta(\mathbf{x}_{q,k},\mathbf{s}_{s,k},\mathbf{x}_{j,k}), j\in J_1(k)\},\\
&\quad \{\theta(\mathbf{x}_{j,k},\mathbf{s}_{s,k},\hat{\mathbf{x}}_{j,k}),\ j\in J_1(k)\})\\
&= \begin{cases}
\min\{f_2\left[D(\mathbf{s}_{s,k},\mathbf{x}_{q,k}),\{\theta(\mathbf{x}_{q,k},\mathbf{s}_{s,k},\mathbf{x}_{j,k}),\ j\in J_1(k)\}\right]\\
\qquad +(0.5-0.2\max\{\theta(\mathbf{x}_{j,k},\mathbf{s}_{s,k},\hat{\mathbf{x}}_{j,k}),\ j\in J_1(k)\}),\\
\qquad f_1(D(\mathbf{s}_{s,k},\mathbf{x}_{q,k}))\}\\
\qquad \text{if } \max\{\theta(\mathbf{x}_{j,k},\mathbf{s}_{s,k},\hat{\mathbf{x}}_{j,k}),\ j\in J_1(k)\}<2^\circ;\\
f_2\left[D(\mathbf{s}_{s,k},\mathbf{x}_{q,k}),\{\theta(\mathbf{x}_{q,k},\mathbf{s}_{s,k},\mathbf{x}_{j,k}),\ j\in J_1(k)\}\right]\\
\qquad \text{otherwise.}
\end{cases}
\end{aligned}
\tag{4.35}
$$

When $\{\theta(\mathbf{x}_{q,k},\mathbf{s}_{s,k},\mathbf{x}_{j,k})<2^\circ,\ \exists j=1,2,...,J\}$, the target return is totally corrupted or hidden by the jammer signal and $P_d=0.02$.

To make comparisons to our proposed strategy, independent active-passive mode scheduling and path planning strategies are used: the active-passive mode scheduling follows a pre-defined radar mode order and the path planning minimizes the trace of the predictive PCRLB of the target.

In the active-passive mode scheduling following a pre-defined order, the following three strategies are used: 1) Alternating between active and passive radar modes, 2) Using active modes for four steps followed by passive radar mode for one time step, and 3) Using active modes for nine steps followed by passive radar mode for one time step. Radar mode scheduling and path planning operating are decoupled and the decisions are made independently. The above mentioned three methods are denoted by "PPA1P1", "PPA4P1" and "PPA9P1" in the following discriptions, respectively.

Table 4.1: Results using different UAV path planning and radar active-passive mode scheduling methods for two UAVs tracking one target in the presence of two jammers

	PPA1P1	PPA4P1	PPA9P1	JPPAPS
Position RMSE (m) when target corrupted	**208**	320	443	260
Position RMSE (m) when target not corrupted	238	183	135	**101**
Track Loss (%)	13	11.9	10.1	**9.5**
Passive Mode (%)	50	20	10	15

The position root mean square errors (RMSEs) of the target when the target return is not corrupted by jammer signals, the position RMSEs of the target when the target return is corrupted by jammer signals, and track loss percentages are recorded and listed in Table 4.1 based on 200 Monte Carlo runs.

From Table 4.1 we can see that with these independent path planning and radar mode scheduling methods, as the passive mode use rate percentage increases, the position RMSE decreases when the target return is corrupted, but the position RMSE increases when the target return is not corrupted along with track loss percentage. This is intuitive and can be interpreted as follows: when the target return is not corrupted, higher radar passive mode percentage means that the radars observe the target less frequently. Therefore, the RMSEs are large for PPA1P1 and PPA4P1 compared with that of PPA9P1. However, when the target is corrupted by the jammers, higher radar passive mode percentage results in means more accurate estimates of jammer locations, hence the spatial anti-jamming technique can suppress the jammer signal better. The number of detections is higher when the radar systems operate at higher passive mode percentage with anti-jamming process than when they operate less in passive mode when the target is corrupted. However, if the radar systems

operate more in passive mode, the track loss percentage will also increase, especially when the target is corrupted by the jammers.

The proposed JPPAPS achieves the lowest position RMSE when the target is not corrupted and the lowest track loss percentage. The position RMSE value of the proposed JPPAPS is the second lowest when the target is corrupted.

To show the outputs of path planning and mode scheduling, the results of one Monte Carlo run from different path planning and mode scheduling strategies are illustrated here. The trajectories of the target and the two jammers can be found in Fig. 4.2. The track of the target and the trajectories of the UAVs using the proposed JPPAPS are shown in Fig. 4.2. The UAVs approach the target and measure it from different angles.

In this Monte Carlo run, for UAV 1, jammer 2 is within 6° of the target LOS from 46s to 114s and within 2° of that from 61s to 85s using the proposed JPPAPS. For UAV 2, jammer 1 is within 6° of the target LOS from 46s to 130s and within 2° of that from 59s to 82s using the proposed JPPAPS. It can be seen from the third subfigure of Fig. 4.3, the radars on the UAVs work in the passive mode when the target is corrupted by the jammer signal. This is achieved by the predictive PCRLB with the detection probability model used by the radar system with the estimate of $\theta(\hat{\mathbf{x}}_{q,k}, \mathbf{s}_{s,k}, \hat{\mathbf{x}}_{j,k})$ and the logic-based track management. The result of "PPA1P1" is not shown here mainly because the stems that represent the radars operating in passive mode are too dense for iteratively one active mode after one passive mode scheduling.

4.6.2 Multiple Target Tracking

In this part, a scenario where two UAVs tracking two maneuvering targets in an environment with two jammers is simulated. The initial locations of the UAVs are $[4 \times 10^4, 5 \times 10^4, 500]^\intercal$m and $[5 \times 10^4, 5 \times 10^4, 500]^\intercal$m respectively, and the initial headings of the UAVs are both $[0, -1, 0]^\intercal$.

Two target trajectories in the benchmark [4] are used as shown in Fig. 4.4. Two jammers fly with an oval (race course) holding pattern in the clockwise direction with the initial positions of $[55000, -20000, 3000]^\intercal$m and $[80000, -20000, 3000]^\intercal$m, respectively, at a speed of 168m/s. In the detection probability model, $\varepsilon = 3.3 \times 10^{-3}$ in (4.33).

The UAV paths from one Monte Carlo run are shown in Fig. 4.4. The intervals of each target being corrupted by jammers and the intervals of radar operating in passive mode in one Monte Carlo run are shown in Fig. 4.5. The position RMSE of the two targets being corrupted by any jammer and that of the two targets not being corrupted by jammers from 200 Monte Carlo runs are shown in Table 4.2 along track loss percentage.

From Fig. 4.5, we can see that the jammers are not simultaneously within 2° of the LOS of UAVs to target 1. Meanwhile, the intervals during which a jammer is within 2° of the LOS of UAVs to target 1 do not last long. Therefore, there is no track loss for target 1 during the mission as shown in Table 4.2. In this case, higher passive mode percentage will lead to a large value of position RMSE regardless of the return of target 1 being corrupted by jammers or not. The proposed JPPAPS strategy achieves the lowest position RMSE when the return from target 1 is not corrupted by jammers and the second lowest position RMSE value when the return

Table 4.2: Results using different UAV path planning and radar active-passive mode scheduling strategies for two UAVs tracking two targets in the presence of two jammers

		PPA1P1	PPA4P1	PPA9P1	JPPAPS
Position RMSE (m) when targets are corrupted	Target 1	642	592	**429**	460
	Target 2	1951	1996	2017	**1327**
Position RMSE (m) when targets are not corrupted	Target 1	574	515	345	**324**
	Target 2	513	424	334	**317**
Track loss (%)	Target 1	0	0	0	0
	Target 2	10.7	7.2	**6.8**	7.7
Passive mode (%)		50	20	10	9

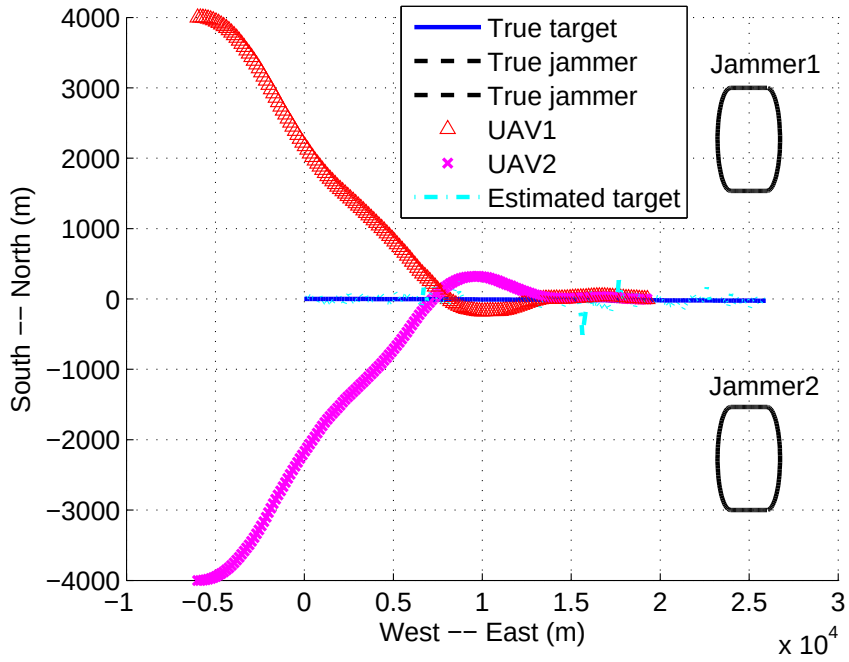
of target 1 is corrupted.

The advantage of the proposed strategy is more obvious in the tracking of target 2. From Fig. 4.5, we can see that the jammers are simultaneously within 2° of the LOS of UAVs to target 2 from 38s to 44s and simultaneously within 6° of the LOS of UAVs to target 2 from 35s to 54s and from 82s to 85s. The interference and radars in passive mode lead to the loss of target 2. The sequantial active-passive-active mode selection strategy results in highest track loss percentage. The proposed JPPAPS, with a low track loss percentage, achieves the best position RMSE values whether target 2 is corrupted or not.

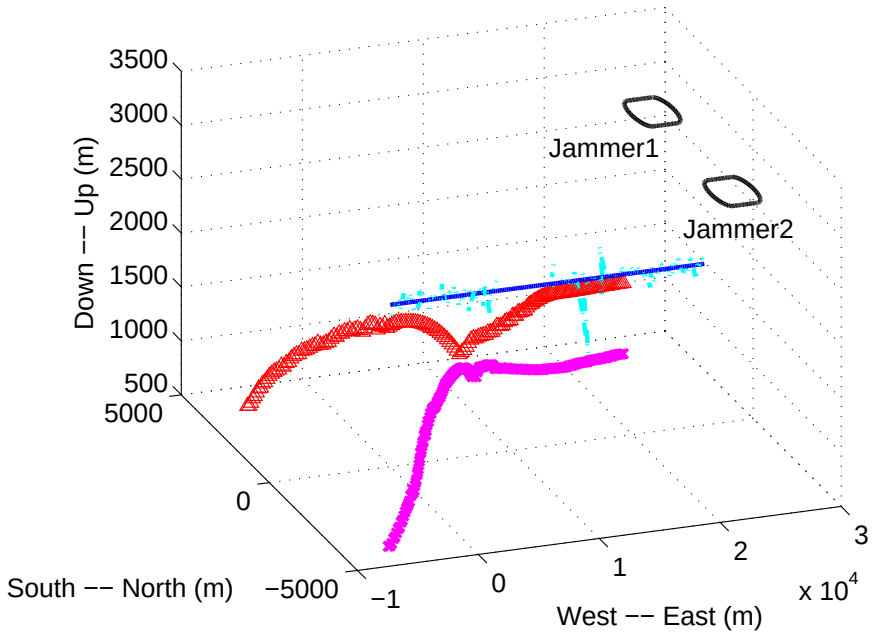
4.7 Conclusions

In this paper, a joint path planning and active-passive mode scheduling strategy was developed for multitarget tracking in a 3D contested environment using airborne radar systems on UAVs. The radars adopt a multibeam strategy with spatial anti-jamming and active-passive mode scheduling against maneuvering targets in the presence of jammers broadcasting wide-band noise as ECM. A receding horizon optimization

based on the predictive PCRLB was formulated to minimize the target state estimation uncertainty. At each time step, the radar determines whether to synthesize beams to point at specific targets or to work in passive mode to provide electronic support measures (ESM). The data association results are used to estimate the detection probability in the optimization formulation and the angle between the estimated target LOS and the estimated jammer LOS. Meanwhile, the UAVs plan the path to balance the tracking of targets or jammers, avoid possible target fire and collision and to reduce the effect of hostile interference. A genetic algorithm solution technique was presented to jointly determine the radar working mode and path selection, subject to constraints on the UAV kinematic model, UAV-to-target, UAV-to-jammer and UAV-to-UAV distances. The proposed JPPAPS performs better than the independent active-passive mode switching following a pre-defined order and path planning to minimize the estimation uncertainty, especially when target returns are corrupted by the jammer signal. As for future works, path planning can be tackled jointly with beam selection [41, 42] and/or power allocation [39] with new formulations using the predictive PCRLB based receding horizon optimization used in this paper.



(a) X-Y view



(b) 3D view

Figure 4.2: Trajectories of the target, two jammers, two UAVs and the estimated target track.

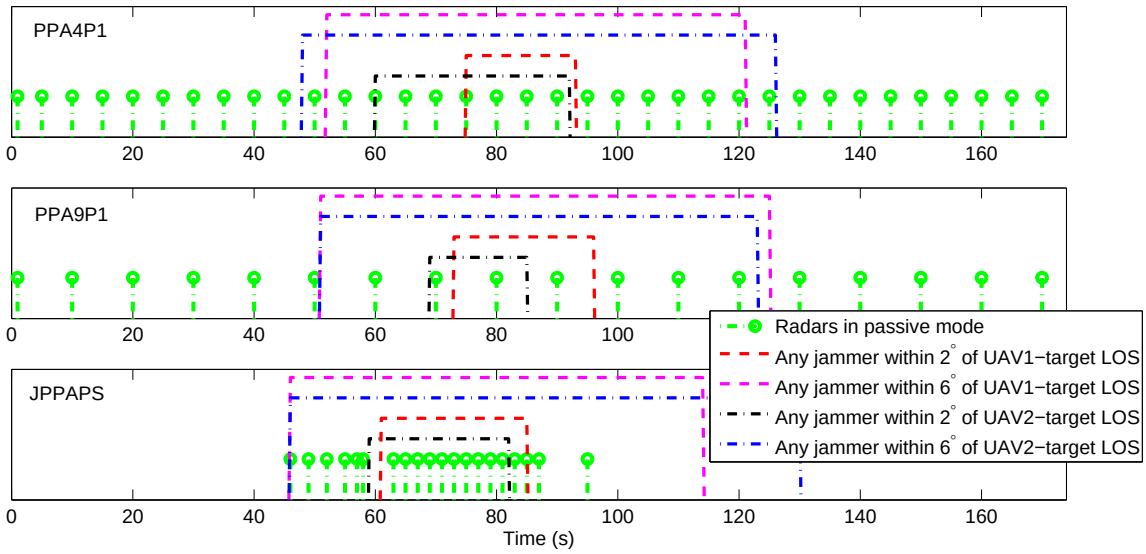
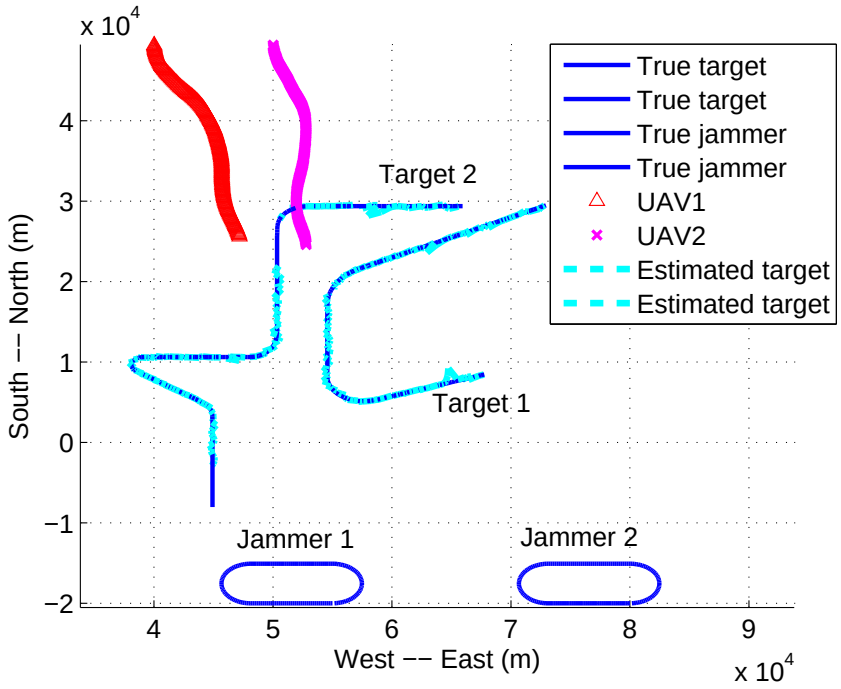
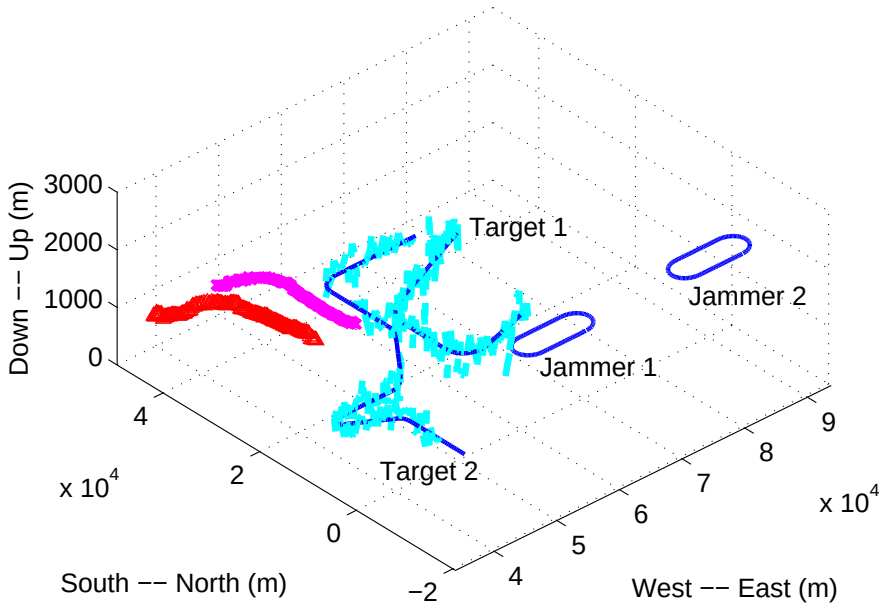


Figure 4.3: Intervals of target returns being corrupted by jammer signal and intervals of radars operating in passive modes in one Monte Carlo experiment for "PPA4P1", "PPA9P1" and the proposed JPPAPS strategies.



(a) X-Y view



(b) 3D view

Figure 4.4: Trajectories for two targets, two jammers, two UAVs and estimated target tracks.

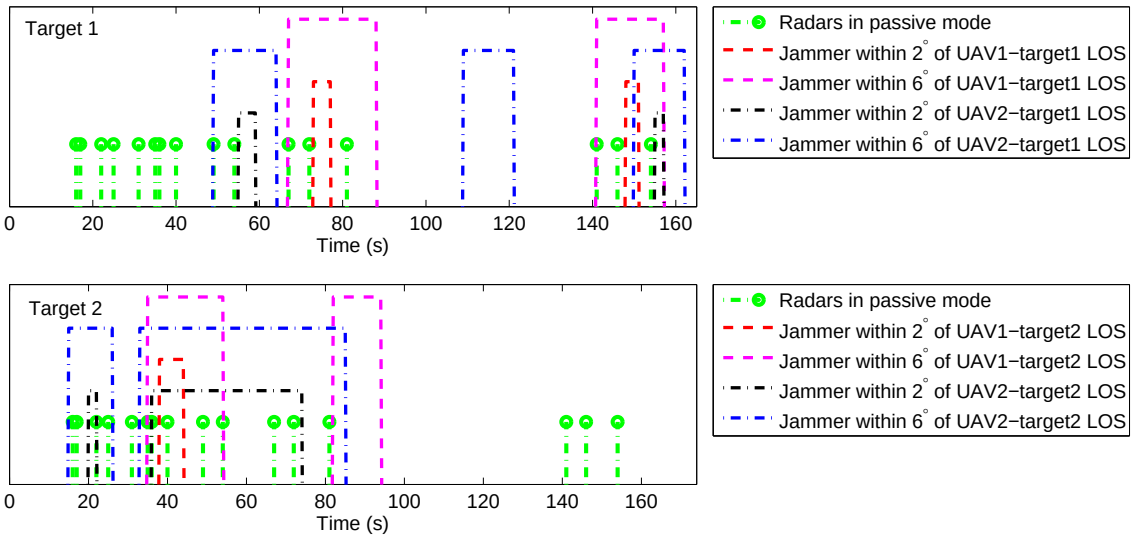


Figure 4.5: Intervals with target returns being corrupted by jammer signal while radars are in passive mode in one Monte Carlo experiment with JPPAPS deployment.

Bibliography

[1] M. Arik and O. B. Akan, "Enabling cognition on electronic countermeasure systems against next-generation radars," in *2015 IEEE Military Communications Conference*, Oct. 2015, pp. 1103–1108.

[2] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software.* John Wiley & Sons, 2004.

[3] P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, and A. Mahajan, "Mixed-integer nonlinear optimization," *Acta Numerica*, vol. 22, pp. 1–131, May 2013.

[4] W. D. Blair, G. A. Watson, T. Kirubarajan, and Y. Bar-Shalom, "Benchmark for radar allocation and tracking in ECM," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 4, pp. 1097–1114, Oct. 1998.

[5] R. Chai, A. Savvaris, A. Tsourdos, S. Chai, and Y. Xia, "Optimal tracking guidance for aeroassisted spacecraft reconnaissance mission based on receding horizon control," *IEEE Transactions on Aerospace and Electronic Systems*, vol. PP, no. 99, pp. 1–1, Jan. 2018.

[6] E. Daeipour, Y. Bar-Shalom, and X. Li, "Adaptive beam pointing control of a phased array radar using an IMM estimator," in *American Control Conference, 1994*, vol. 2, Jun. 1994, pp. 2093–2097.

[7] K. Doğançay, "UAV path planning for passive emitter localization," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 2, pp. 1150–1166, Apr. 2012.

[8] C. Fernndez-Prades, J. Arribas, and P. Closas, "Robust GNSS receivers by array signal processing: theory and implementation," *Proceedings of the IEEE*, vol. 104, no. 6, pp. 1207–1220, June 2016.

[9] F. Gavilan, R. Vazquez, and E. F. Camacho, "An iterative model predictive control algorithm for UAV guidance," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 3, pp. 2406–2419, July 2015.

[10] E. Grossi, M. Lops, and L. Venturino, "Track-before-detect for multiframe detection with censored observations," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 3, pp. 2032–2046, July 2014.

[11] Y. Gu and A. Leshem, "Robust adaptive beamforming based on interference covariance matrix reconstruction and steering vector estimation," *IEEE Transactions on Signal Processing*, vol. 60, no. 7, pp. 3881–3885, July 2012.

[12] M. L. Hernandez, A. Farina, and B. Ristic, "PCRLB for tracking in cluttered environments: measurement sequence conditioning approach," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 42, no. 2, pp. 680–704, Apr. 2006.

[13] M. L. Hernandez, T. Kirubarajan, and Y. Bar-Shalom, "Multisensor resource deployment using posterior Cramer-Rao bounds," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, no. 2, pp. 399–416, Apr. 2004.

[14] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. MIT press, 1992.

[15] J. Hu, L. Xie, J. Xu, and Z. Xu, "Multi-agent cooperative target search," *Sensors*, vol. 14, no. 6, pp. 9408–9428, 2014.

[16] C. Jauffret and Y. Bar-Shalom, "Track formation with bearing and frequency measurements in clutter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 26, no. 6, pp. 999–1010, Nov. 1990.

[17] H. Jiang, W. Yi, T. Kirubarajan, L. Kong, and X. Yang, "Multiframe radar detection of fluctuating targets using phase information," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 2, pp. 736–749, Apr. 2017.

[18] Y. Jin, Y. Liao, A. A. Minai, and M. M. Polycarpou, "Balancing search and target response in cooperative unmanned aerial vehicle (UAV) teams," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 36, no. 3, pp. 571–587, June 2005.

[19] T. Kirubarajan and Y. Bar-Shalom, "Low observable target motion analysis using amplitude information," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, no. 4, pp. 1367–1384, Oct. 1996.

[20] T. Kirubarajan, Y. Bar-Shalom, W. D. Blair, and G. A. Watson, "IMMPDAF for radar management and tracking benchmark with ECM," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 4, pp. 1115–1134, Oct. 1998.

[21] T. Kirubarajan, Y. Bar-Shalom, and E. Daeipour, "Adaptive beam pointing control of a phased array radar in the presence of ECM and false alarms using IMMPDAF," in *American Control Conference, Proceedings of the 1995*, vol. 4, June 1995, pp. 2616–2620.

[22] J. Li, P. Stoica, and Z. Wang, "On robust Capon beamforming and diagonal loading," *IEEE Transactions on Signal Processing*, vol. 51, no. 7, pp. 1702–1715, Jul. 2003.

[23] M. Mallick, V. Krishnamurthy, and B.-N. Vo, *Integrated Tracking, Classification, and Sensor Management: Theory and Applications*. John Wiley & Sons, 2012.

[24] D. Mušicki, T. L. Song, H. H. Lee, X. Chen, and T. Kirubarajan, "Track-to-track fusion with target existence," *IET Radar, Sonar Navigation*, vol. 9, no. 3, pp. 241–248, Mar. 2015.

[25] N. H. Nguyen and K. Doğançay, "Optimal geometry analysis for multistatic TOA localization," *IEEE Transactions on Signal Processing*, vol. 64, no. 16, pp. 4180–4193, Aug. 2016.

[26] R. Pitre, X. Li, and R. Delbalzo, "UAV route planning for joint search and track missions—An information-value approach," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 3, pp. 2551–2565, July 2012.

[27] S. Ragi and E. K. P. Chong, "UAV path planning in a dynamic environment via partially observable Markov decision process," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 4, pp. 2397–2412, Oct. 2013.

[28] M. Richards, *Fundamentals of Radar Signal Processing, Second Edition*. McGraw-Hill Education, 2014.

[29] J. R. Riehl, G. E. Collins, and J. P. Hespanha, "Cooperative search by UAV teams: A model predictive approach using dynamic graphs," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 4, pp. 2637–2656, Oct. 2011.

[30] V. Roberge, M. Tarbouchi, and G. Labonte, "Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 132–141, Feb. 2013.

[31] A. Sinha, T. Kirubarajan, and Y. Bar-Shalom, "Autonomous ground target tracking by multiple cooperative UAVs," in *Aerospace Conference, 2005 IEEE*, Mar. 2005, pp. 1–9.

[32] ——, "Autonomous surveillance by multiple cooperative UAVs," *Proc. SPIE*, vol. 5913, pp. 59 131V–59 131V–12, 2005.

[33] ——, "Autonomous search, tracking and classification by multiple cooperative UAVs," *Proc. SPIE*, vol. 6235, pp. 623 508–623 508–12, 2006.

[34] R. Tharmarasa, T. Kirubarajan, M. L. Hernandez, and A. Sinha, "PCRLB-based multisensor array management for multitarget tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 2, pp. 539–555, Apr. 2007.

[35] R. Tharmarasa, T. Kirubarajan, and T. Lang, "Joint path planning and sensor subset selection for multistatic sensor networks," in *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, Jul. 2009, pp. 1–8.

[36] S. A. Vorobyov, A. B. Gershman, and Z.-Q. Luo, "Robust adaptive beamforming using worst-case performance optimization: a solution to the signal mismatch problem," *IEEE Transactions on Signal Processing*, vol. 51, no. 2, pp. 313–324, Feb. 2003.

[37] Y. Wang, L. Wang, Z. Zhang, R. Yao, and K. Wei, "Robust beamforming for motional interference mitigation in DSSS," in *WAMICON 2013*, Apr. 2013, pp. 1–4.

[38] G. Wu, W. Pedrycz, H. Li, M. Ma, and J. Liu, "Coordinated planning of heterogeneous earth observation resources," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 1, pp. 109–125, Jan. 2016.

[39] M. Xie, W. Yi, T. Kirubarajan, and L. Kong, "Joint node selection and power allocation strategy for multitarget tracking in decentralized radar networks," *IEEE Transactions on Signal Processing*, vol. 66, no. 3, pp. 729–743, Feb. 2018.

[40] S. Xu and K. Doğançay, "Optimal sensor placement for 3-D angle-of-arrival target localization," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 3, pp. 1196–1211, June 2017.

[41] J. Yan, H. Liu, B. Jiu, B. Chen, Z. Liu, and Z. Bao, "Simultaneous multibeam

resource allocation scheme for multiple target tracking," *IEEE Transactions on Signal Processing*, vol. 63, no. 12, pp. 3110–3122, June 2015.

[42] J. Yan, H. Liu, W. Pu, S. Zhou, Z. Liu, and Z. Bao, "Joint beam selection and power allocation for multiple target tracking in netted colocated MIMO radar system," *IEEE Transactions on Signal Processing*, vol. 64, no. 24, pp. 6417–6427, Dec. 2016.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this thesis, the optimal path planning for airborne surveillance by multiple aerial vehicles is combined with temporal scheduling, scan area decision making and radar active-passive mode scheduling, respectively.

The spatial and temporal coordination of multiple sensor platforms (e.g., patrolling aircraft) for target searching in disjoint AOIs over an extended time horizon, i.e., a multi-period coverage path planning and temporal scheduling, is formulated as a combinatorial optimization and solved by multiobjective evolutionary algorithm. Optimal surveillance mission plans are developed considering the cooperative visiting and revisiting of areas by aircraft taking off from different base stations. New objective functions based on the number of covered areas are proposed to evaluate the gathered information to promptly update information collected from AOIs and to ensure persistent surveillance.

Considering the flyable region of the sensing platforms (e.g., UAVs) , such as the

UAV altitude, UAV-to-target and UAV-to-UAV distance, the optimal sensor geometry and optimal path planning for target localization using active or passive sensors are formulated and analysed. A joint path planning and scanning strategy is developed for a group of UAVs to search possible targets and track detected targets in a mobile 3D surveillance space. The objective function is formulated to maximize the predictive information in the Fisher sense from the surveillance space, which includes the surveillance space centroid, detected targets as well as the yet-to-be-detected targets.

To take advantage of path planning and the spatial signature of the jammers to mitigate hostile interference, a joint path planning and active-passive scheduling strategy is developed for a group of UAVs equipped with radars, which work in either active or passive modes. A receding horizon optimization is formulated based on the predictive PCRLB under the UAV flyable region constraints. Simulations and real-world case studies are given to illustrate the performance and advantages of the proposed strategies and solution techniques.

5.2 Future Work

In the research of coverage path planning and temporal scheduling, we integrated the path planning inside and outside the AOIs, and aircraft take-off time scheduling for each flight in one optimization formulation. For future research, these tasks can be decoupled and tackled separately, which will lead to new problem definition and formulation. For example, the path of POIs in one AOI can be selected at first, and then the path of AOIs can be planned. The entire mission time can be divided into slots for each aircraft so that the path planning and scheduling can be done within

each time window. The advantage of this separate path planning and scheduling approach is that it requires less computational resources, it makes dynamic planning based on information gathered possible, and it could involve more sensor resource management issues, such as the selection of base stations. In addition to the multi-objective evolutionary algorithm, we can test other solution techniques, such as the column generation [1, 4] and Markov decision process [2, 3], to tackle this mission planning problem.

In the research of joint path planning and scanning, we assumed that UAVs in the same group work in the same mode. For future research, the optimal sensor geometry and path planning for a group of UAVs with sensors in multiple modes observing the same target can be studied. In the research of joint path planning and active-passive scheduling, benchmark problems for maneuvering target tracking in the presence of ECM using a group of moving sensors can be developed for future work, which should consider issues such as radar signal processing, target tracking, data association, and sensor management.

Bibliography

[1] E. Choi and D.-W. Tcha, "A column generation approach to the heterogeneous fleet vehicle routing problem," *Computers & Operations Research*, vol. 34, no. 7, pp. 2080–2095, 2007.

[2] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

[3] S. Ragi and E. K. P. Chong, "UAV path planning in a dynamic environment via partially observable Markov decision process," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 4, pp. 2397–2412, Oct. 2013.

[4] R. A. Skitt and R. R. Levary, "Vehicle routing via column generation," *European Journal of Operational Research*, vol. 21, no. 1, pp. 65–76, 1985.