Asif Khan, M.Sc.

Coordinated Unmanned Aerial Vehicles for Surveillance of Targets

DISSERTATION

to gain the Joint Doctoral Degree
Doctor of Philosophy (PhD)
Alpen-Adria-Universität Klagenfurt
Fakultät für Technische Wissenschaften

in accordance with
The Erasmus Mundus Joint Doctorate in Interactive and Cognitive Environments
Alpen-Adria-Universität Klagenfurt — Queen Mary University of London
Universitá degli Studi di Genova

First Supervisor
Prof. Bernhard Rinner
Institute of Networked Embedded Systems
Alpen-Adria-Universität Klagenfurt, Austria

Second Supervisor
Prof. Andrea Cavallaro
School of Electronic Engineering and Computer Science
Queen Mary University of London, UK

September, 2015

Acknowledgments

This PhD Thesis has been developed in the framework of, and according to, the rules of the Erasmus Mundus Joint Doctorate on Interactive and Cognitive Environments EMJD ICE [FPA n^o 2010-0012] with the cooperation of the following Universities:

Alpen-Adria-Universität Klagenfurt - AAU

Queen Mary University of London - QMUL

Technische Universiteit Eindhoven - TU/e

Universitá degli Studi di Genova - UNIGE

Universitat Politécnica Catalunya - UPC

According to ICE regulations, the Italian PhD title has also been awarded by the Universit degli Studi di Genova.
This work was supported by Lakeside Labs GmbH, Klagenfurt, Austria with funding from the European Regional Development Fund and the Carinthian Economic Promotion Fund (KWF) under grant 20214/21530/32602.

Lakeside Labs GmbH - Klagenfurt, Austria

First Reviewer
Prof. Bernhard Rinner
Institute of Networked Embedded Systems
Alpen-Adria-Universität Klagenfurt, Austria

Second Reviewer
Prof. Andrea Cavallaro
School of Electrical Engineering and Computer Science
Queen Mary University of London, UK

Declaration of Honor

I hereby confirm on my honor that I personally prepared the present academic work and carried out myself the activities directly involved with it. I also confirm that I have used no resources other than those declared. All formulations and concepts adopted literally or in their essential content from printed, unprinted, or Internet sources have been cited according to the rules for academic work and identified by means of footnotes or other precise indications of source.

The support provided during the work, including significant assistance from my supervisor has been indicated in full.

The academic work has not been submitted to any other examination authority. The work is submitted in printed and electronic form. I confirm that the content of the digital version is completely identical to that of the printed version.

I am aware that a false declaration will have legal consequences.

Asif Khan	Klagenfurt am Wörthersee, September 21, 2015
Signature	Place, Date

Dedicated to my mother and father.

Coordinated Unmanned Aerial Vehicles for Surveillance of Targets

Abstract

This thesis investigates the coordination approaches of multiple mobile and autonomous robots, especially resource-limited small-scale UAVs, for the surveillance of pre-defined ground targets in a given environment. A key research issue in surveillance task is the coordination among the robots to determine the target's time varying locations. The research focuses on two applications of surveillance: (i) cooperative search of stationary targets, and (ii) cooperative observation of moving targets. The objective in cooperative search is to minimize the time and errors in finding the locations of stationary targets. The objective of cooperative observation is to maximize the collective time and quality of observation of moving targets.

The thesis presents a survey of the approaches in a larger domain of multi-robot systems for the surveillance of pre-defined targets in a given environment. This survey identifies various factors and application scenarios that affect the performance of multi-robot surveillance systems. The thesis proposes a distributed strategy for merging delayed and incomplete information, which is a result of sensing and communication limitations, collected by different UAVs. An analytic derivation of the number of required observations is provided to declare the absence or existence of a target in a region. This number of required observations is integrated into an iterative use of Travelling Salesman Problem (TSP) and Multiple Travelling Salesmen Problem (MTSP) for autonomous path planning of UAVs. Additionally, it performs an exploration of the algorithmic design space and analyzes the effects of centralized and distributed coordination on the cooperative search of stationary targets in the presence of sensing and communication limitations.

The thesis also proposes the application of UAVs for observing multiple moving targets with different resolutions. A key contribution is to use the quad-tree data-structure for modelling the environment and movement of UAVs. This modelling has helped in the dynamic sensor placement of UAVs to maximize the observation of the number of moving targets as well as the resolution of observation.

Contents

List of Figures

List of Tables

Glossary

(x, y)	Coordinates in the environment.
A_k	Number of UAVs hovering at node k.
A	Number of UAVs.
B	Number of targets.
C	Cost of movement between the levels of quad-tree.
E	Total time steps.
F_i	Field of view of U_i.
K	Total number of nodes in Quad-tree.
M	Number of path traversals.
N	Number of simulation runs.
$O^t_{i,c}$	Observation in cell c by U_i at time step t.
O^t_{ij}	Observation of T_j by U_i at time step t.
P^m_c	Occupancy probability in c after m positive observations.
$P^t_{i,u}$	Occupancy probability in Ω_i at the location of U_u.
P_M	Probability that M^{th} observation is the first detection.
P^t_c	Probability of occupancy in cell c at time step t.
T_j	j^{th} target.
T	Average time to complete cooperative search.
U_i	i^{th} UAV.
X_c	Event in cell c.
Ω_i	Local search map of U_i.
Ω	Environment.
Φ	State transition matrix.
Ψ	Collective maximum resolution.
Θ^+	Threshold for assigning a cell as target cell.
Θ^-	Threshold for assigning a cell as empty cell.
Υ	Collective time of observation for B targets.
α	Weight assigned to time and resolution of observations.
γ	Process noise in taking measurement of a target.
κ	Number of correct detections in a cell.
$\mathcal{N}$	Normal distribution.
$\overline{B}$	Number of targets found.

$\overline{e}$	Number of faulty target detections.
$\overline{m}_{c,\mu}$	Average number of observations to declare that c is empty.
$\overline{m}_{c,\sigma}$	Minimum number of observations to declare that c is empty.
$\overline{m}$	Number of negative observations.
$\tau_{i,c}$	Time stamp.
$\mathbf{C}$	Set of cells in the environment.
$\mathbf{E}$	Set of edges.
$\mathbf{G}$	Graph.
$\mathbf{H}$	Observation matrix.
$\mathbf{K}_d$	Set of nodes at depth d.
$\mathbf{N}_i$	Set of neighboring UAVs of U_i.
$\mathbf{O}$	Observation noise covariance matrix.
$\mathbf{Q}$	Process noise covariance matrix.
$\mathbf{R}$	Set of paths $\mathbf{R} = \{\mathbf{r}_i : 1 \leq i \leq A\}$.
$\mathbf{S}$	Set of candidate cells $\mathbf{S} \subseteq \mathbf{C}$.
$\mathbf{T}$	Set of targets $\mathbf{T} = \{T_1, T_2, ..., T_B\}$.
$\mathbf{U}$	Set of UAVs $\mathbf{U} = \{U_1, U_2, ..., U_A\}$.
$\mathbf{V}$	Quad-tree.
$\mathbf{X}^t$	Set of states for all targets $\mathbf{X}^t = \{\mathbf{x}_1^t, \ldots, \mathbf{x}_B^t\}$.
$\mathbf{Y}^t$	Set of states for all UAVs $\mathbf{Y}^t = \{\mathbf{y}_1^t, \ldots, \mathbf{y}_A^t\}$.
$\mathbf{Z}^t$	Set of measurements for all targets $\mathbf{Z}^t = \{\mathbf{z}_1^t, \ldots, \mathbf{z}_B^t\}$.
$\mathbf{r}_i$	Path of U_i.
$\mathbf{w}$	Temporary variable to store the state $\mathbf{y}_i^t$.
$\mathbf{x}_j^t$	State of T_j at time step t.
$\mathbf{y}_i^t$	State of U_i at time step t.
$\mathbf{z}_{ij}^t$	Measurement of T_j by U_i at time step t.
ε	Maximum depth of the Quad-tree.
φ	Length of one side of FOV F_i.
ϑ	Observation noise.
b_k	Number of targets visible under node k.
c	A cell.
d	Depth of the Quad-tree.
e	Detection error (%).
$f(c,t)$	Objective function of cooperative search.
g	Collective observation time and resolution.
h_{ab}	Euclidean distance between cell a and cell b.
m_c	Number of observations in cell c.
$m_{c,\mu}$	Average number of observations to declare that c contains a target.
$m_{c,\sigma}$	Minimum number of observations to declare that c contains a target.
n	Number of probability values to be merged.

p	Probability of detection.
q	Probability of false positive.
r	Communication range.
s_{ij}^t	Resolution of observation O_{ij}^t.
t	Time step.
v_k	Value at node k.
w	Weight assigned to information merging strategies.
z_0	Threshold on minimum allowed altitude of UAV.
z_i^t	Altitude of U_i at time step t.
$\hat{s}_j^t$	Maximum resolution of T_j at t.

Chapter 1

Introduction

The application of autonomous mobile robots for the surveillance of an environment has gained a lot of interest over the last three decades [112]. Autonomous mobile robots have been utilized as surveillance agents in a variety of applications such as area coverage, intruder detection, search and rescue, building maps, localization of targets, and security [112, 126]. The advantages of such robots for the surveillance of an environment are to limit the risks to personnel, to reduce labor requirements, to reduce the cost, and to increase the efficiency. However, apart from these benefits, a single autonomous robot is resource-limited and cannot provide long-term and reliable surveillance in complex environments [107]. The recent advances in sensing, embedded processing, and wireless communication capabilities suggest the use of a team of multiple autonomous and mobile robots. Such multi-robot systems provide several advantages over the use of a single robot e.g., increased spatial coverage [129], robustness due to sharing of information and data fusion [17, 58], fault-tolerance due to information redundancy [121], lower cost of multiple simple robots rather than single powerful robot, reliability in case of single robot failure [117], and teaming of specialized robots [88, 90].

Coordination among the individual team members is a key towards the success of multi-robot systems for surveillance of an environment [42]. The coordination among the robots refers to sharing and merging of information and joint decision-making [54]. It helps the robots to manage their limited resources of sensing, communication, processing, and battery (life) and to increase the performance of surveillance. It also enables the operation of a large number of robots by a single user. Sharing of information increases the ability of robots to perceive the global view of the environment. Two important types of this information are information about the robots (e.g., location, path, future intention) and information about the environment (e.g., area coverage, uncertainty, location of obstacle). Joint decision-making, on the other hand, enables the robots to reason with each other on their motions [94]. An important aspect of joint decision-making is distribution of sensing and processing among the team members. It ensures that individual actions of the robots are beneficial to the team as a whole and contribute to the team-level objective. Because of successful coordination, the robots determine the type, time, and merging method of partial information. Additionally, they determine the time and location of a visit (when and where to move). Uncoordinated behavior is likely to result in redundant coverage of the environment and a waste of resources. Depending on the application and available resources, the coordination can be performed in a centralized or distributed fashion.

Unmanned Aerial Vehicles (UAVs) are special types of robots that can move in 3D space. Their high maneuverability and ability to move in space make them suitable for aerial surveillance of a large-scale environment and accessing hard-to-reach places. The significant technological advances in UAVs and their cost-effective solutions provide an opportunity to use a team of UAVs for aerial surveillance of an environment [11]. The use of teams of UAVs for automation of many civil applications such as search and rescue [119, 118], disaster management [100], surveillance [125], multispectral monitoring [124], forest fire detection [88], target search [129], goods delivery and construction [9], sports [83], crowd and social movement monitoring [85], and wildlife research [103] is therefore steadily increasing. However, the design principles of such a team of multiple autonomous UAVs for aerial surveillance of an environment need intensive investigations and remain an open research problem.

1.1 Motivation and Objective

The motivation for this thesis is to develop coordination approaches for a team of UAVs to perform aerial surveillance of an environment, which contains some targets of interest. This demands an investigation of the approaches of multiple mobile and autonomous robots for the surveillance of such an environment. We first survey the approaches in a larger domain of multi-robot system for the surveillance of a given environment with moving targets. We then limit this larger domain and consider the use of a team of resource-limited small-scale UAVs such as quad-rotors and multicopters [125, 81].

In surveillance missions, the information is typically gathered in the form of visual observation of the environment by sensors (e.g., cameras) on board the robots. However, the two major limitations of these sensors, i.e., limited field of view (FOV) and sensing errors (miss detection and false positive), make the problem of surveillance very challenging, even for very simple environments. The target in surveillance tasks can be a missing/injured person [119, 118], intruder [117], player [83], animal [103], fire [88], a pre-defined phenomenon [77], or chemical/biological/radiological event [133]. These missions are typically time critical and span a large geographical area. A key research issue in these missions is that of sensor placement—determining time varying locations of robots to gather information. Such missions require the coordination of individual robots to operate as a team and to achieve mission specific goal.

Teams of autonomous UAVs are used with increasing interest in civil and commercial applications and by the scientific research community. Small-scale UAVs are of particular interest due to their ease of deployment, high maneuverability, and low costs. Airborne cameras and sensors are valuable sources of information. They help in building an overview of the environment and assessing the current situation. Examples of aerial surveillance systems that use a team of UAVs are cDrones [101, 100, 1, 2] and SINUS [7, 124, 105, 3] projects of Alpen-Adria-Universität Klagenfurt, Austria.

The cDrones project has developed a system of wireless networked UAVs for generating an aerial overview image of a given environment. Several UAVs coordinate and fly in formation over an environment of interest and deliver sensor data such as images. These images are fused on the ground, analyzed in real-time to detect targets such as cars or persons, and delivered to the user. The goals of cDrnoes project are to establish and maintain a reliable communication network among the UAVs, to coordinate the movements of UAVs, and to mosaic the local images. This system can be used in disaster management applications.

The SINUS project focuses on integrating the four key components of aerial surveillance into a single system. These components are: (i) the multiple UAVs, (ii) the surveillance sensors on-board the UAVs, (iii) the aerial network, and (iv) the coordination, which organizes the individual tasks of the UAVs to achieve a common mission goal. Such a tight integration is necessary for deploying self-organizing UAVs in dynamic and partly unknown environments.

In this thesis, we study multi-robot surveillance systems in general and multi-robot aerial (multi-UAV) surveillance systems in particular. We focus on two applications of surveillance using a team of small-scale UAVs: (i) cooperative search of stationary targets [14, 128, 119, 118], and (ii) cooperative observation of moving targets [95, 6, 4].

In cooperative search, the UAVs survey the entire environment as quickly as possible to find the locations of stationary targets. The sensing and communication limitations make the cooperative search process dynamic, imprecise, and uncertain. It requires the UAVs to survey (part of) the environment multiple times before confirming the locations of targets. Generally, each UAV maintains local information about the environment [128, 14] that serves as the

UAV's knowledge base about the state of the environment and targets. The UAVs move around, observe parts of the environment, and update the local information. Each UAV is likely to perceive (parts of) the environment differently due to some deviations in the available information at the UAVs. The UAVs coordinate by sharing the updated local information with the team members and by making joint decisions about their next moves.

In cooperative observation, the UAVs dynamically plan their movements to increase the observation of moving targets, which are usually more than the UAVs. Observation of a target refers to its inclusion in the FOV of any UAV. In addition to sensing and communication limitations, the movement of targets also contributes to increase the complexity of this surveillance application. Coordination among the UAVs is important to minimize the observation overlap, which is observing a single target by multiple UAVs, and to guide the UAVs for observing the maximum number of targets.

The key research problem in both of these applications is that of coordinated and dynamic route planning of UAVs. The UAVs must know how to coordinate for determining their time varying locations in order to achieve the application specific goal. The objective of this thesis is the development of coordination techniques for dynamic placement of UAVs to accomplish the cooperative search and cooperative observation. The objective in cooperative search is to minimize the time and errors in finding the locations of stationary targets. Additionally, the objective is to identify the design space and to analyze the effects of the type of coordination (centralized or distributed) on the two components of coordination (information sharing and joint decision-making). The objective of cooperative observation is to maximize the collective time and quality of observation of moving targets. Information gathering is based on the sensor observation and quality of sensor. Therefore, our research objective pertains to show the effect of sensor quality on time and accuracy of information.

1.2 Research Questions

We address the following research questions:

1. How do the UAVs share and merge their local information?
   UAVs need to communicate with each other and the ground station to share information. The communication limitations do not allow the UAVs to share complete information all the time. This compels the UAVs to select a subset of information to share at a specific time. The local information of a UAV and the information received from other team members

are possibly incomplete, erroneous, and outdated. The UAVs should merge this information in such a way that helps in reducing the efforts of the whole team.

2. How does a UAV make joint decisions to plan its movement?
Even if UAVs share their local information, they should coordinate to decide the locations for future visits and the frequency of visits. This requires the UAVs to know the required number of sensor observations in each location they visit. Similarly

1.3 Contributions

This thesis explores the design space of coordinating a team of resource limited UAVs for aerial surveillance in a given environment. The surveillance goal is to find the location of targets in the environment and to increase the observation time and quality if the targets are moving. This requires coordination among a team of UAVs to determine their movements and sharing of information.

The key contributions can be summarized as follows.

1. We have performed a study of the existing approaches in multi-robot coordination for surveillance applications, focusing on cooperative search and cooperative observation of targets [70, 105]. We have highlighted the advantages and limitations of these approaches. This has led us to prepare a survey on existing approaches of coordinating multiple moving autonomous robots for observing moving targets [72].

2. We have proposed a distributed strategy for merging delayed and incomplete information about the environment collected by different UAVs. The delayed and incomplete information is a result of sensing and communication limitations of UAVs. We have also shown a trade-off in time of cooperative search and detection errors and the effect of various sensing and communication parameters [73, 98].

3. We have provided an analytic analysis of the number of required observations to collect accurate information and to declare the absence or existence of a target in a region. This number of required observations has been integrated into a formal system model for cooperative search with constraints in sensing, information exchange, and network connectivity. We have also analyzed the effects of centralized and distributed coordination on this model for cooperative search [74].

4. We have proposed the application of UAVs for observing multiple moving targets with different resolutions. The UAVs have the ability to hover and move in 3D environment, which enables them to observe the environment at different spatial scales (resolutions). This scale of observations has been associated to the quality of observation of a target. A key contribution is to use the quad-tree data-structure for modelling the environment and movement of UAVs. This modelling has helped in the dynamic sensor placement of UAVs to maximize the observation of the number of moving targets as well as the resolution of observation [71].

This work contributes towards a larger research effort aimed at developing cooperative control algorithms that will allow a team of UAVs to complete surveillance missions without direct human intervention, and in the presence of uncertainty.

1.4 Thesis Outline

The remainder of this thesis is organized as follows.

Chapter 2 covers related work on coordination in autonomous mobile robots for surveillance applications. The first part of the chapter surveys the coordination approaches for a team of multiple mobile robots that observe moving targets in a given environment. The second part of the chapter limits the scope of the study and describes related work on the use of only UAVs for searching stationary target.

In Chapter 3, a formal definition of the surveillance problem using a team of UAVs is given. The chapter introduces a formal description of environment, UAV, sensor, target, and sensing and communication limitations. Additionally, the objectives of cooperative search and cooperative observation are presented.

Chapter 4 describes the first component of coordination i.e., information sharing and merging in cooperative search of stationary targets. It explains the observation, updating of local information and different strategies for merging of shared information. The chapter also shows the effects of sensor and communication parameters on minimizing the time and errors of cooperative search.

Chapter 5 explains the second component of coordination i.e., joint decision-making for planning the paths of the UAVs. The chapter provides an analytic analysis of the number of required observations to decide the existence or absence of a stationary target. The chapter also describes the use of this required number of observation in TSP and/or MTSP formulation to decide the paths of UAVs. Additionally, the chapter explores the algorithmic design space by presenting four algorithms for different coordination scenarios.

In Chapter 6, the application of a team of UAVs for observation of moving targets is presented. It presents the proposed quad-tree based modeling of the environment and UAV motion. The chapter also presents a centralized algorithm for planning of UAVs to increase the observation time and quality of moving targets.

Finally, Chapter 7 concludes the thesis and gives an outlook to related future research directions.

Chapter 2

Related Work

This chapter discusses the current state-of-the-art in (a larger domain of) multi-robot surveillance systems for a given environment, which is populated with some targets. The discussion is based on coordination approaches among the robots and applications of these surveillance systems.

The chapter studies two applications of multi-robot surveillance systems: (i) cooperative observation of moving targets, and (ii) cooperative search of stationary targets. In cooperative observation, the robots dynamically plan their movements to maximize the observation of moving targets. In cooperative search, the UAVs survey the entire environment as quickly as possible to find the locations of stationary targets.

Section 2.1 describes a multi-robot surveillance system in general and the factors that affect it. Section 2.2 explores the work on first application of multi-robot surveillance systems (as mentioned in Section 1.1), namely cooperative observation of moving targets. Section 2.3 then limits the scope of the study and presents the existing work on second application of multi-robot surveillance systems (as mentioned in Section 1.1), namely (multi-UAV) cooperative search of stationary targets using a team of only aerial robots (UAVs). Section 2.4 discusses the differences of our contribution to the state-of-the-art.

2.1 Multi-robot Surveillance System

A multi-robot surveillance system consists of a set of robots that are equipped with surveillance sensors and wireless communication devices. These robots are deployed for the surveillance of an environment. The surveillance task usually depends on collecting information about a number of targets in the environment. The sensing and communication capabilities of these robots are limited, which make it difficult for the robots to gain a full view of the environment and targets. Fig. 2.1 shows an example scenario of a multi-robot surveillance system in an environment with multiple targets.

The coordination is sharing and merging of information and joint decision-making among the robots. Two important types of information that are shared are information about the robots (e.g., location, path, future intention) and information about the environment (e.g., area coverage, uncertainty, location of obstacle). To abstract meaningful information, the robots merge the shared information. Joint decision-making, on the other hand, enables the robots to reason with each other, which results in a decision about their paths.

Each robot iteratively executes four actions: (i) taking observations and processing the data locally, (ii) sharing and merging information with other robots (and the ground station), (iii) making joint decisions for planning their motion (path), and (iv) generating control actions to execute the physical motion of a robot according to its planned path. Once the robots move to a new location, they use this new location information and new observation from the sensor to repeat the process. Fig. 2.2 shows an overview of the information flow, which is valid for both the applications (as mentioned in Section 1.1) of a multi-robot surveillance system. This process is different from the traditional motion planning where the robots decide shortest paths from their known initial locations to known goal locations [53]. In the multi-robot surveillance system, the selection of locations to visit are dynamic. Once a robot takes an observation, it can then perform a variety of actions to achieve an application specific objective e.g., confirming the location of a target.

Approaches for both the applications of multi-robot surveillance systems (as

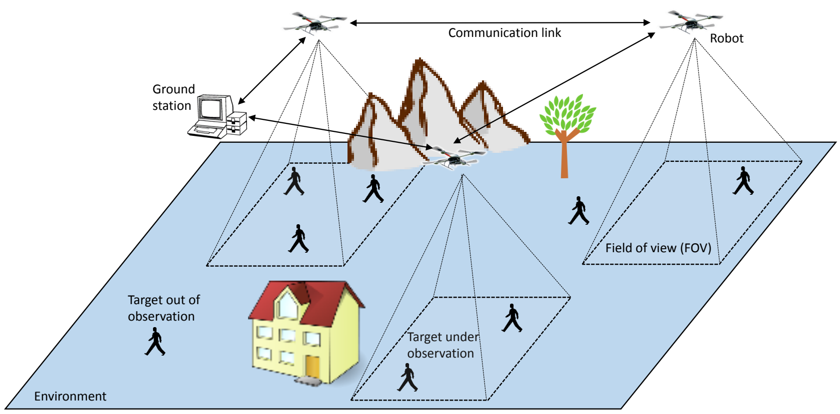


Figure 2.1: Illustrative example of cooperative mobile robots for observing multiple mobile targets. Three robots are observing six out of eight targets in a given environment. The coordination may be centralized on the ground station or distributed on the autonomous robots.

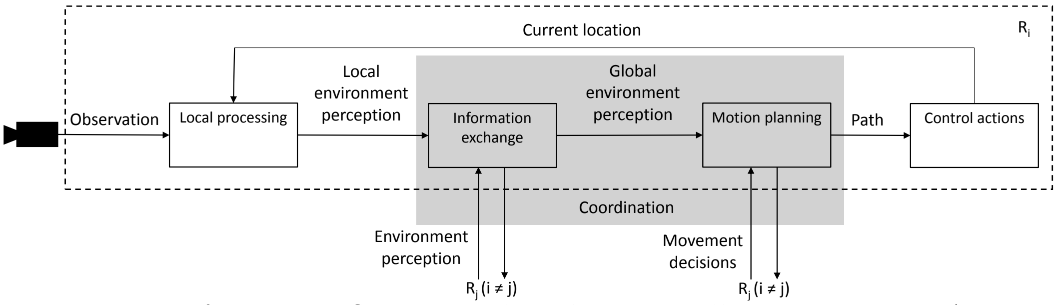


Figure 2.2: Information flow in multi-robot surveillance systems. A set of robots observe the targets in the environment and derive local information about the environment and targets. This local information is exchanged with other robots to derive global information, which serves as key input for the decision-making about robot movements. The coordination (information exchange and/or motion planning) can be either centralized or distributed among the different robots.

mentioned in Section 1.1) can be characterized by five factors: environment, targets, robots, sensors and coordination. The multi-robot surveillance systems also depend on why robots need to observe and for how long they need to observe the targets. In the following sections, we describe these five factors.

2.1.1 Environment

Robots can operate on the ground, in the air, or underwater. Even to just maintain a best view of an environment with no targets is a difficult task [107]. In this section, we identify key characteristics of the environment (its representation, structure and evolution) and their effects on our problem. Table 2.1 classifies related work based on these key characteristics.

The environment confines the movement of both robots and targets, and affects observations by potentially restricting a sensor's FOV. The environment, which is usually bounded, is *represented* as a continuous [95, 93, 50] or discrete [77, 71] 2D plane or 3D space. A ground robot may move on a 2D surface embedded in a 3D space [93]. A 3D Euclidean space environment facilitates operations of an aerial robot [57, 47, 71, 75, 48, 49, 50]. A region with a boundary of regular shape such as a circle [93, 80] or rectangle [122] can be used to represent 2D movement of robots and targets. Most approaches that use a regular shape of the environment are based on simulation. Limited research has been conducted using irregular shapes to represent the boundaries of the environment as they introduce extra challenges [29]. Moreover, the terrain of the environment (e.g., planar, irregular or cluttered surface) may limit the mobility of robots to observe the targets.

Table 2.1: Classification of related work based on key characteristics of the environment (S: Structured, U: Unstructured).

Structure	**Evolution**	**Representation**	
		2D	3D
U	Static	[26, 48]	[116, 95, 93, 79]
		[5, 49]	[111, 96, 35, 78, 120, 39, 34]
		[77, 66]	[39, 34, 80, 86, 123, 108, 71]
	Dynamic	[122]	[29, 28, 99, 63, 30]
S	Static	[82, 84]	[64, 65, 114, 115, 44]
		[36]	[51, 62, 19]
	Dynamic	-	[75]

The environment can be structured or unstructured. The *structure* of an environment may restrict the sensor's observation as well as the movement of targets and robots. The structure of the environment is usually represented by a suitable data-structure e.g., occupancy grid map [38, 50] or Voronoi diagram [62]. A structured environment may consist of a well-defined floor plan known to the robots [75]. Examples of navigation in structured environments include road-following approaches where the aim is to detect movable paths and to navigate them [15]. In structured environments [64, 65, 114, 115, 44, 51, 62, 19], the location and size of potential obstacles are usually assumed to be known, and allowed and forbidden regions for the movement of targets and robots are explicitly specified. *Unstructured* environments provide no defined paths, boundaries or locations of obstacles [63, 30], and usually consist of outdoor environments without a defined map or information about obstacles.

The environment can be static or can *evolve* dynamically. A static environment consists of unchanging surroundings throughout the mission [95, 78, 26, 48]. In dynamic environments, the movement of obstacles [29, 28], variations in geometry of the environment [75] and variation in terrain [99] continuously change the surroundings. Operations in dynamic environments require the robots to dynamically adapt to changing situations, thus introducing an additional challenge.

2.1.2 Target

The observation of a target depends on three characteristics: type, mobility and representation.

We can identify three *types* of targets, namely cooperative, non-cooperative, and evasive (Table 2.2). Cooperative targets [92] continuously or occasionally transmit some information, such as GPS coordinates or another signal, to

Table 2.2: Classification of related work based on three different types and the number of mobile targets.

Target type	**Number of targets**	
	Constant	Variable
Cooperative	[82, 84, 26, 92]	-
Non-cooperative	[62, 93, 71, 108, 111]	[29, 28]
	[96, 120, 86, 35, 79, 78, 39, 34, 80, 99, 123]	[47, 75]
	[95, 50, 63, 64, 65, 114, 115, 44, 59]	[57, 30]
Evasive	[122, 116, 5, 51, 62, 19]	[36]

robots thus making localization and observation easier. In most applications, targets are non-cooperative [95, 79, 78, 120, 47, 57]. Non-cooperative targets neither hide themselves from observation nor send information to robots about their location. Evasive targets can sense the robots and avoid being observed, which makes their observation more difficult [122, 116, 5, 51, 62, 19, 36].

The *mobility* of all types of targets is constrained by the environment. Most works on cooperative mobile robots for observing moving targets focus on ground targets and prior information about their mobility models. The movements of targets are usually unforeseen and independent of each other as observing one target generally does not provide useful information about the location or behaviors of other targets. Random walks [26] and linear motion models [57] are the dominant mobility models used for representing target motion. The mobility model of evasive targets depends also on the mobility of robots.

A target can be *represented* as a point, whose coordinates are determined by the sensor. This point can be represented as 2D planar coordinates on a predefined grid of known size [59, 77] to show the position of a target. In addition to positon coordinates, the representation of a target can also include constant [93, 78, 47, 48] or variable [29, 28] velocity components. Some approaches combine position and target density information to represent a target [65, 75]. In more complex scenarios, a target is represented using a combination of position, velocity, and uncertainty/noise components [30, 57, 49].

Most approaches are applicable only for observing known and constant *number of targets* moving in the environment [95, 35, 78, 120, 39, 50, 114, 66, 115, 44, 111, 48, 49, 122, 116, 5, 51, 62, 19, 82, 84]. Only few approaches consider movement of a variable number of targets [29, 36]. The number of targets may change when a target appears and disappears for certain duration of time [91]. Throughout a mission, targets may leave or enter the environment [29] in (usually) known entry or exit points [95, 93]. The lack of prior knowledge (number and location) about the entry/exit points in the environment introduces challenges when observing multiple moving targets [28].

2.1.3 Robot

A cooperative mobile robot needs four main *capabilities*: sensing, processing, communication, and mobility.

Robots *sense* the environment to generate observations and then process these local observations to produce a local perception of the environment and the targets. The processing depends on the available resources on the robot and on application-specific objectives. Based on their local perception of the environment, the robots exchange information, such as location information, the state of the environment and sensor data with team members, to reach a global perception of the environment. Moreover, to improve their future movement decisions robots may communicate to share information, such as their intentions, goals and actions. The movement decisions generate a path for the mobility of the robot, which produces control actions for moving to a new location according to this path. The perception of the environment and movement plan depend on the specific application objective.

There are generally three *types* of the autonomous moving robots (Table 2.3): Unmanned Ground Vehicles (UGVs), Unmanned Aerial Vehicles (UAVs), and Unmanned Underwater Vehicles (UUVs). Each type of robot has unique characteristics. The progress in UGVs and wireless mobile sensor nodes motivate the use of cooperative mobile robots for observation of moving targets [95, 79]. In this study, we consider wireless mobile sensor nodes that move on ground as UGVs. UGVs can enter narrow and indoor passages, can withstand rough environmental conditions and can carry heavy sensors. The major limitations of UGVs are smaller FOV and 2D movement. The recent development in UAVs [89, 81] has made it possible to develop approaches for 3D movement of robots. Aerial observation of ground moving targets [47, 57] with the help of sensors attached to the UAVs bring another interesting dimension to the research on cooperative mobile robots. The strength of these UAVs are low-altitude flight, hovering at a specific point in the environment, a larger FOV, and high-speed movement. However, the smaller size of UAVs limits their payload capabilities. They cannot carry heavy sensors available for UGVs. The conventional UGVs will still be used and researched due to specific constraints of the environment, the target, the robot, and the application. The use of UUVs is highly challenging due to their unique limitations e.g., problems with underwater signal (GPS, radio, acoustic) transmission, 3D mobility affected by water currents, and high cost [55]. Despite these challenges, research work on deployment of UUVs for data gathering is actively in progress. However, we could find very little work on underwater cooperative UUVs for observing multiple mobile underwater targets [20, 134].

Traditional applications of multi-robot systems are based on homogeneous

Table 2.3: Grouping of the approaches based on types of the robot.

Robot	Ref.
UGV	[95, 96, 93, 35, 79, 78, 120, 39, 34, 80, 86, 99, 114, 82]
	[50, 47, 57, 30, 26, 66, 48, 49, 122, 5, 19]
UAV	[29, 28, 75, 123, 64, 65, 51]
	[63, 108, 111, 59, 71]
UUV	[20, 134]
Heterogeneous	[115, 44, 116]

robots having similar capabilities. Recent developments in robotic applications benefit from a team of heterogeneous robots with different capabilities that increase the performance of multi-robot systems. For example, a region occluded by tall buildings or trees may not be observed by UAVs but coordinating UGVs could observe that region [116, 57]. The operating environment may also demand for heterogeneous robots with different capabilities to explore different parts of the environments [116, 115, 44]. The heterogeneity in robotic platforms introduces the challenge of how to effectively deal with heterogeneous-information exchanges and decision-making. Moreover, performance guarantees on whether such heterogeneity improve or degrade system performance are needed.

2.1.4 Sensor

Physical constraints of the robot and the environment may influence the directivity of the sensor [66, 122] (Table 2.4). Typical sensor *types* for target detection include vision sensors [51, 75] and range sensors (e.g., radar, sonar, laser scanner) [79, 78, 120, 80]. The aggregated area of all the sensors' FOVs is generally much smaller than the area to be monitored.

In addition to limited FOV, challenges in sensing include limited sensor performance (e.g., detection errors, noise) and limited target observability (e.g., due to occlusions). A *deterministic model* of a sensor represents a perfect sensor with no errors in reporting the location of a target. In this case, the observation follows the model in Eq. 3.12. Sensing *errors* can be modeled with target location uncertainties represented with probability distributions. A *probabilistic model* includes two types of sensing errors: measurement noise and detection errors. Measurement noise is represented as a probability distribution in the sensor output and models inaccurate estimations of the coordinates of the target location (and its size). Detection errors model false positive detections and miss-detections.

Combining information from *multiple sensors* mounted either on the same robot or on different robots is desirable for improving the perception of the environment [73, 106]. Information fusion from heterogeneous sensors, e.g., vision sensor and laser scanner, vision sensor and GPS sensor, helps increasing robustness against sensing errors, the accuracy of distance and orientation estimation of targets, and environment features as shown already for a static sensor network [23]. Information fusion can improve camera calibration and sensor movement in single-robot applications [104]. Moreover, information fusion can be employed to improve the robot (self-) state estimation with respect to environment, targets and other robots. New techniques are desirable to enhance sensor fusion in both distributed and non-distributed systems.

2.1.5 Coordination

Robots achieve coordination by sharing information to improve the perception of the environment and by jointly performing decision-making for motion planning [98]. The coordination among robots can be centralized, decentralized, or distributed (Table 2.5).

The functionality of cooperative robots depends greatly on their networking capabilities and timely information exchange, especially in time critical missions. Coordination needs reliable networks with guaranteed QoS (e.g., bandwidth, delay) to cope with connectivity and time-varying network latency of highly mobile and cooperative robots. Wireless communication among the robots is limited and may hinder the smooth and uninterrupted information exchange among the robots because of limited bandwidth or temporary loss of

Table 2.4: Classification of related work based on sensor type and sensor model (Om: Omni-directional, Di: Directional, Pr: Probabililistic, Dt: Deterministic, DE: Detection Errors).

			Sensor model			
			Pr		Dt	
			Noise	DE	Noise	No errors
Sensor type	Vision	Om	-	[116]	[99]	[75, 82, 84]
				[59]	[47]	[51, 62, 19]
				-	-	[123, 5]
		Di	[66]	[122]	-	-
	Range	Om	[114] [108]	-	[50]	[64, 36, 95]
					[57]	[96, 93, 35]
					[30]	[79, 78, 120]
					[63]	[80, 86, 29, 28, 111]
		Di	-	-	[48, 49]	-

Table 2.5: Classification of related work based on three types of coordination.

Centralized	[69, 80, 50, 57, 62, 75]
	[30, 66, 114, 108]
Decentralized	[86, 82, 84]
Distributed	[47, 116, 95, 93, 79, 48, 49]
	[29, 64, 65, 115, 44, 51, 122, 63, 19]
	[120, 35, 39, 34, 96, 123, 59, 5, 111, 78, 28, 99, 26]

connectivity. Moreover, the communication requirements of robots moving in 3D environments [7, 131] are different from those of ground robots.

In *centralized* coordination, robots exchange information with a central node or ground station that produces globally optimal plans using task assignment [50], optimization techniques [57, 30], clustering [75], triangulation [66] or scheduling [62]. However, the central node is a single point of failure and might not receive complete and updated information due to sensing and communication limitations. In fact, communication problems severely affect centrally coordinated robots by isolating (permanently or temporarily) one or more robots from the available global information and decisions.

In *decentralized* coordination, there are multiple leader robots that act as central nodes for smaller groups of robots [82, 84, 86]. Each leader then coordinates with other leaders.

Finally, robots with a sufficient amount of memory and processing power can coordinate in a *distributed* manner. In distributed coordination each robot decides independently, even with limited available information, using for example artificial force vectors [96, 93, 35, 79, 78, 120], auctions [39], region partitioning [64, 65], data fusion [48, 49] or game theory [51, 116]. Distributed algorithms enable individual robots to operate with partial available information and are therefore only marginally affected by communication problems.

2.2 Multi-robot Cooperative Observation

Multi-robot cooperative observation refers to cooperative motion planning of moving robots to keep multiple moving targets in the limited FOV by at least a single robot. It is part of a broader topic multi-robot cooperation, which is a well-researched topic supported by a number of surveys. The limitations in the sensor's FOV and the highly dynamic nature of the environment due to movement of targets make the target observations a very challenging task for the robots. A key research problem for accomplishing such tasks is to plan the motion of the robots such that the number of targets under observation by at least a single robot is maximized. Fig. 2.1 shows an example scenario of cooperative mobile robots for observing multiple moving targets in a given environment. The applications of cooperative mobile robots for observing moving targets include surveillance [92], search operations [50], sports [83], crowd and social movement monitoring [69, 85], and wildlife research [103, 67].

A target is considered under observation if it is in the FOV of a robot at specific time. This observation of a target mainly depends on the type of the surveillance sensor attached to the robot and its FOV. A single robot can observe more than one target at a time. Similarly, more than one robot can observe the same target(s) at a time. The robots can coordinate with each other by communicating information about themselves and targets under their observation.

The problem is to plan the motion of the robots to increase the number of targets under observation by at least a single robot. The interesting aspect is the prior information about the count of the targets. Knowing how many targets are moving in the environment at a given time can greatly affect the cooperative observation process. By comparing the number of targets under observation and total number of targets moving in the environment, robots can decide to search for the unobserved targets or continue observing the already found targets. The lack of prior information about the number and location of targets makes the motion planning difficult for the robots as they always assume and search for unobserved targets. In the problem of observing moving targets, the goal locations are the locations of targets, which are not known and highly dynamic. Once a robot observes a target and determines its state, the robot can then perform a variety of actions to achieve application specific objective e.g., tracking the target.

While a number of surveys on related topics have been published [42, 68, 15, 37, 53, 8, 21, 33, 126], they do not present an overview on multi-robot systems where robots *as well as* targets move in a given environment. Survey papers on mobile robots [15, 37, 53] have been focused on the classification and explanation of motion planning approaches for a single robot to explore a region. A system of cooperative mobile robots requires approaches that combine coordination and motion planning [8, 21, 33, 126]. Existing surveys on cooperative mobile robots encompass tasks *without targets* such as collision avoidance, area coverage, map making and marching [126], or with *static targets*, such as foraging and landmine detection [21, 33, 126]. Farinelli et al. [42] presented a taxonomy of multi-robot system approaches that classified them based on coordination dimensions and system dimensions. Only coordination aspects of multi-robot systems were covered in [42, 68], whereas the use and benefits of the cloud infrastructure to support the operations of coordinating robots (cloud robotics) were discussed in [68]. No survey has so far covered approaches for cooperative mobile robots equipped with sensing, processing and communication capabilities for observing *multiple moving targets*.

We group existing approaches based on four major application scenarios, namely Cooperative Tracking (CT), where the objective is to persistently track moving targets; Cooperative Multi-robot Observation of Multiple Moving Targets (CMOMMT) [95], which aims to increase the collective time of observation for all targets; Cooperative Search, Acquisition, and Track (CSAT) [47], which alternates search and track of moving targets; and Multi-robot Pursuit Evasion (MPE) [116], whose objective is to capture evasive targets. In the following subsections, we discuss these application scenarios.

2.2.1 Cooperative Tracking (CT)

CT aims to minimize the uncertainty over moving target locations and/or to increase the visibility of multiple moving targets. In CT, *cooperative paths* can be designed for fixed-wing, minimum turn-angle UAVs, to increase the frequency of individual observations of moving target in an uncluttered, outdoor environment [111]. Regions with high target density lead to wasted resources as robots visit each target individually. Optimal circular paths can also be designed for fixed-wing, high altitude UAVs to increase visibility of targets in densely populated urban areas [75]. All the targets may be divided into groups (whose number is equal to the number of UAVs) and the center of each UAV's circular path is updated for the best view of the group.

When target paths are predefined, a simple strategy of revisiting paths can be designed using moving Gaussian peaks [123]. To best visualize the targets with known locations, multiple views of the targets can be desirable [26]. Triangulation-based location estimation of moving targets can be achieved with a moving target being constantly observed by multiple moving robots [66]. Cooperative paths for the movement of robots can be designed not only to accurately localize the target but also to minimize the energy consumption of robot movement. The movement of robots was formulated on a graph where each node represents a location in the environment.

Several works track moving targets in outdoor, unstructured, uncluttered and bounded environments. Clustering of robots and a distributed mechanism for coordination are used in [48, 49] to track cooperative targets that transmit signals to the robots. The robots use directional antennas, line-of-sight, time-of-arrival and direction-of-arrival to generate (noisy) measurements of the target location that are predicted using a Kalman filter. Multiple moving small ground robots can be used to observe animals using range sensors and binary decisions to indicate the presence/absence of a target in a sensors neighborhood [67]. The location of a target is corrected locally by vote decision fusion.

Tracking is based on a penalized maximum likelihood framework to addressed the problem of variable number of targets as animals move in (appear) and out (disappear) of the environment. Multiple moving targets can be observed in a multi-region structured environment [63] with the assumption of prior knowledge of the densities of both targets' and robots' locations, but without coordination between robots of the same region. This approach was extended to outdoor environments [64] for tracking multiple targets in regions with a high target-density to robot-density ratios.

Using a flocking control algorithm, swarms of ground robots with range sensors with omnidirectional FOV have been used to track targets that avoid obstacles in a structured and cluttered environment [82, 84]. Flocks of robots split and merge into multiple smaller flocks that track a single target. Each robot is assumed to know the location of other robots and targets. A combination of UAV and UGV [115] has also been proposed for tracking mobile targets, where ground robots move in a structured and cluttered environment. They used range sensors with omnidirectional circular FOV in a rectangular environment. Each new observation triggers exchange of information between robots and causes a change of the behavior in the robot's movement. The work was further extended to include detection errors in sensing and by minimizing energy consumption for sensor path-planning [44, 114].

In CT, the estimated locations of targets are assumed to be known to the robots, which do not search for unknown targets. The search for unknown targets is covered in the next subsections.

2.2.2 Cooperative Multi-robot Observation of Multiple Moving Targets (CMOMMT)

The goal of CMOMMT is to dynamically place robots to maximize the *collective time* during which targets are observed, when the number of moving targets is larger than that of robot. The number of non-cooperative targets with unknown locations is constant and the environment is uncluttered and with regular shape. CMOMMT maximizes not only the number of targets under observation but also the duration of observation for each target. A target is assumed to be under observation when it is within the FOV of a robot. CMOMMT is an NP-hard problem and was first proposed by Parker and Emmons [95].

To maximize the collective time of observation, robots operate in search or track mode. Mode switching is decided based on the presence of targets in the FOV of each robot. In *search* mode, when a robot finds one or more targets, it starts to *track* them. During the tracking mode the robot moves toward the center of mass of the set of moving targets under observation. Then the robot switches back to search mode when there are no more targets in its FOV. Local force vectors are used to attract and repel robots to achieve coordination [95, 93]: a robot is attracted to the targets to keep close enough to observe them and repulsed by neighbor robots to avoid observation overlap.

Approximate CMOMMT (A-CMOMMT) includes weighted local force-vectors [96, 93] to reduce observation overlap of a single target by multiple robots to help increase the collective time of observation. Personality CMOMMT (P-CMOMMT) [35] addresses the problem that some targets may be observed for most of the time, while others could be completely unobserved. To make sure that all targets are observed, P-CMOMMT uses information entropy for the evaluation of the target diversity under observation. To minimize the problem of losing a target from observation, Weighted CMOMMT (W-CMOMMT) assigns different priority weights to targets based on Broadcast of Local Eligibility (BLA) [120].

Behavioral CMOMMT (B-CMOMMT) [79, 78] includes a third mode of operation, the *help* mode, when a robot broadcasts a help request to other robots when it is losing a target from its FOV. The robots in search mode respond to these help requests by approaching the robot that is in need of help. B-CMOMMT also introduces targets tags in the coordination process to reduce overlap in target observation. B-CMOMMT can be improved with an extended Kohonen map for each robot to reach the target and an auction-based algorithm for cooperation [39].

Instead of using local force vectors and help calls, Formation-CMOMMT (F-CMOMMT) uses flexible formation of robots [34]. Model-predictive control strategies [80] can also be used for CMOMMT but at the cost of a high computational complexity. The effects of degree of decentralization, speed of targets, and sensing range on collective observation of targets are analyzed in [86]. The work compares K-means clustering and hill-climbing algorithms, which are scalable in degree of decentralization, for achieving the objective of CMOMMT. The expected motion patterns of the targets can be exploited to observe each target for an equal amount of time [10].

The CMOMMT framework has been used for iceberg observation [29, 28]. The number of targets in iceberg observation problem vary with time and the entry/exit points of these targets are not known. The objective in such a problem is to minimize the time of initial contact with the newly generated targets.

The strengths of CMOMMT are its capability of switching between two modes of operation (search and track) and of working under limited-range communication. However, most CMOMMT approaches are based on uniform FOVs, observations with constant resolution and assume a perfect sensor without errors. Moreover, as soon as a robot finds one or more targets in its FOV it starts its tracking mode, thus restricting the search of the remaining regions that may contain more targets. In addition to this, there is no situational awareness, as robots do not share their perceptions of the environment. For example, in search mode robots work independently without cooperating with each other. For these reasons, CMOMMT is not suitable for real-world applications where the assumption of perfect sensors is unrealistic and situational awareness is needed.

2.2.3 Cooperative Search, Acquisition, and Track (CSAT)

CSAT integrates search and track modes into a single strategy [50]. Task assignment is used for the team of robots to search and track (observe) as many targets as possible. Each robot in CSAT purposely switches its mode repeatedly between search and track. Mode switching is decided using the uncertainty level in location of a target. CSAT includes a situational awareness in the form of search map that is updated by all the robots enabling tracking robots to aid the search process. The robot tries to keep the location uncertainty of a found target under a given bound throughout the mission. Targets are tracked only for a specified amount of time or until their states are adequately determined. Once a target is located accurately, its location is recorded and the target is temporarily left unobserved. The robot then starts searching for more targets thus causing a gradual increase in the uncertainty about the location of already located target. To keep this uncertainty under a given bound the robot quits the search mode, approaches the previously located target, and switches back to track mode. The motivation for a robot to switch its mode from tracking to search is the assumption that there is always at least one unknown target. Most of the work in CSAT uses aerial robotics with a downward pointing camera for observation [57, 47].

Unlike CMOMMT, in CSAT noisy sensors and measurement noise can be handled, a robot tracks only one target at a time and can switch to search mode even if it is successfully tracking a target. In CSAT, mode changes are frequent and a robot does not lock its operation in a given mode thus facilitating keeping a balance between search and track operations [57].

The first CSAT approach used a recursive Bayesian framework and a 2D grid containing probability density function (PDF) of a target that guides the robot movement for search and maintains the information about the environment and target [50]. The robots use the PDF to share their perceptions of the environment and to decide the mode switching. A tracking metric based on covariance matrix of the target state [47] incorporates the growth of the target-location uncertainty. Besides using the covariance matrix, a multi-agent task assignment algorithm is used for mode switching [57, 30]. To integrate the conflicting objectives of target-search and target-track, an objective function is used that is based on the average value of information gained by the mission and that represents the number of targets detected and how well each detected target is tracked [99]. Different terms in the objective function control the importance given to detection and tracking. This approach depends on prior information about target distributions and lack on-line path generation.

Existing CSAT approaches face the same problem as CMOMMT: a small number of targets may capture the attention of a robot (i.e., a robot frequently switches mode only for a small subset of targets in a small part of the environment).

2.2.4 Multi-robot Pursuit Evasion (MPE)

Unlike CT, CMOMMT and CSAT, in MPE targets are evasive and can be faster than the robots. MPE approaches do not aim to increase the observation time of a target, but to observe targets only once. The observation in pursuit-evasion is also called capture: one or more robots try to capture one or more targets that try to avoid being captured. The goal of MPE is to minimize the time required to capture evasive targets.

A distinguishing feature of pursuit-evasion (also known as adversarial search) is the *intelligence* of the target that has full knowledge of the environment and is aware of robot location and intent [27]. The robot and target motions are therefore somehow inter-dependent as robots and targets compete with each other [126]. For this reason, motion-planning problems that arise in adversarial settings are related to a probabilistic game theoretical framework.

While single-target pursuit-evasion has been an active topic of research for the last two decades (see survey in [27]), only a few works consider observation of multiple moving targets by multiple moving robots, i.e., the Multi-robot Pursuit Evasion (MPE) problem [116, 16]. Two main MPE variations exist [27]. In the first variation, a robot associates itself with a captured target and maintains this association until the mission ends [5] (target tracking for only M targets). In the second variation, a target is removed from the mission as soon as it is detected/captured [122] and the robot that captured this target continues looking for other targets.

Several works assume unstructured environments. A distributed approach based on hierarchical decomposition algorithm for differential game theory was used for UAVs [51]. The deterministic model of the sensor (with no sensing errors) makes the approach less attractive and applicable. Targets can be captured using a centralized MPE algorithm with task-scheduling heuristics that assign robots to different parts of the environment, which was partitioned using Voronoi diagrams [62]. The negotiation and auction mechanisms of economics theory were also used to assign multiple robots to the targets [19] with the objective of maximizing the probability of capture while minimizing the time to capture. In order to achieve this objective, the robots used dynamic coalition formation for planning their paths.

Other MPE works consider irregular-shaped, structured and planar environment with obstacles for area coverage using range sensors with cone-shaped directional FOV [36]. A team of mobile ground robots with limited sensing and communication capabilities coordinates to guarantee the capture of targets, considering a deterministic model of a perfect sensor (i.e. with no errors). A greedy decentralized approach that employs a heterogeneous team of UAVs and UGVs was used to minimize the capture time of targets in [116]. A distinctive characteristic of this approach is the modelling of detection errors in the sensing process.

2.3 Multi-UAV Cooperative Search

In multi-UAV cooperative search, generally, UAVs coordinate to minimize the time of finding the location(s) of one or more stationary targets. Each UAV maintains a data structure, known as search map or cognitive map [128, 14] that serves as the UAVs knowledge base about the state of the environment and other UAVs. The UAVs update the search map by collecting information from parts of the environment using their on-board surveillance sensors (e.g., cameras). The sharing and merging of these local search maps help UAVs in better understanding of the environment and planning their paths. The ultimate goal is to determine the necessary movement actions in order to gain as much information as possible about the target locations. To achieve the goal the UAVs must decide what search action to take (i.e., when and where to move in the environment) and what information to send or receive. Multi-UAV cooperative search is thus defined by three components: (i) sensing the environment and updating the search map by individual UAVs, (ii) sharing local information, and (iii) making joint decisions about search actions based on the available information. These three components correspond to local processing, information exchange and motion planning blocks of Fig. 2.2. Multi-UAV cooperative search always require joint decision-making.

Each UAV has limited sensing (e.g., limited FOV, detection errors) and communication (e.g., limited range and bandwidth) capabilities. These limitations result in an incomplete and potentially outdated search map of the environment for each UAV. The detection errors in sensing require the UAVs

Table 2.6: Summary of approaches for cooperative observation of multiple moving targets. C: Centralized, Co: Cooperative, Comb: Combination of UAV and UGV, D: Distributed, Dc: Decentralized, DE: Detection Error (False detection and miss detection), Di: Directional, Dt: Deterministic, E: Evasive, M: Number of robots, N: Number of targets, Nc: Non-cooperative, Om: Omnidirectional, Pr: Probabilistic, S: Structured, U: Unstructured.

	Ref.	Environment		Target				Robot	Sensor				Coordination	
		Structure	Space	M/N	N	Identification	Type		Type	Error	Model	FOV	Algorithm	Type
CMOMMT	[95]	U	3D	< 1	Known	No	Nc	UGV	Range	No	Dt	Om	Force vectors	D
	[96, 93, 35]												Weighted force vectors	
	[79, 78, 120]					Yes							Force vectors and help calls	
	[39]												Neural network and auctions	
	[34]												Robot formation	
	[80]					No							Model-predictive control	C
	[86]												Clustering	Dc
	[29, 28]				Unknown	Yes		UAV					Gaussian Mixture Model	D
CSAT	[50]	U	3D	< 1	Known	Yes	Nc	UAV	Camera	Noise	Dt	Om	Task assignment	C
	[99]												Task assignment	D
	[47]			> 1	Unknown	No			Range				Recursive Bayesian	
	[57, 30]			<< 1									Optimization	C
CT	[75]	S	3D	< 1	Unknown	No	Nc	UAV	Camera	No	Det	Om	Clustering	C
	[123]	U			Known								Moving Gaussian peaks	D
	[59]			> 1						DE	Pr		Data fusion	
	[63]								Range	No	Dt		Region partitioning	
	[64, 65]	S	2D	< 1									Region partitioning	
	[26]	U					Co	UGV		Noise	Pr		Kalman filter	
	[108]		3D					UAV					Algorithmic	C
	[114]	S		>= 1		Yes	Nc	UGV					Adaptive sampling	
	[66]	U	2D	< 1					Camera			Di	Triangulation	
	[115, 44]	S	3D	>> 1		No		Comb		No	Det	Om	Geometric optimization	D
	[82, 84]		2D	< 1			Co	UGV	Range				Flocking algorithm	De
	[111]	U	3D	> 1			Nc	UAV	Camera				Gradient approximation	D
	[48, 49]		2D					UGV	Range	Noise		Di	Data fusion	
MPE	[36]	S	2D	> 1	Unknown	Yes	Ev	UGV	Range	No	Dt	Om	Region clearing	D
	[62]	U			Known	No							Negotiation	
	[122]			= 1					Camera	DE	Pr	Di	Greedy pursuit	
	[116]	S	3D	> 1				Comb				Om	Game theory	
	[5]					Yes		UGV		No	Dt		Force vectors	
	[51]							UAV					Differential game	
	[62]			< 1				UGV					Task-scheduling	C
	[19]												Negotiation and auctions	D

to visit parts of the environment multiple times before confirming the existence and location of a target. When neighboring UAVs communicate and share their incomplete and outdated search maps, the merging of information within the search maps helps the UAVs in accurately estimating and learning the global state of the environment [18]. Joint decision making and planning, on other hand, enable the UAVs to predict the behavior (position, orientation, area covered) of other UAVs in the neighborhood [110]. A UAV should be able to make its decisions based on subjective information obtained through observation, communication with other UAVs and merging of information. The coordination approach among the UAVs e.g., task assignment depends on the goal and performance metric of the multi-UAV cooperative search (e.g., reducing search time, collision avoidance and fast coverage).

Multi-UAV cooperative search can be traced back to the work of Passino et al. [97]. They model a pre-defined rectangular environment as a discretized grid of cells (search map), which is shared by a team of fixed wing, high speed UAVs. They propose to coordinate the paths of fixed wing UAVs to increase the number of visits in all the cells uniformly. Most of the existing multi-UAV cooperative search methods are designed for fixed wing, high speed UAVs with unique constraints on their speed, altitude, and turn angle. Depending on the application scope, different approaches have been proposed in the past to perform multi-UAV cooperative search. The performance metrics of these approaches include: minimizing the time of finding the location of a target [119], minimizing the energy consumption of covering the environment [109], minimizing the uncertainty in the environment [128, 110] and maximizing the accuracy of information about the target location [56], maximizing the number of found targets or the combination of two or more of these metrics. The constraints of physical maneuverability (speed, altitude, turn angle), fuel time, sensor range and communication make multi-UAV cooperative search a very challenging problem.

Although coordination among UAVs is desirable [129, 128], there is no universally optimal behavior [86] for cooperative search. One must evaluate some interrelated criteria (e.g., independence vs interdependence of UAVs, local vs. global communication among UAVs and homogeneity vs. heterogeneity of information) for designing various coordination strategies. The intuitive response is to allow each UAV to communicate its local information with other UAVs. Unfortunately, this depends on the available communication and sensing resources.

We classify the existing multi-UAV cooperative search approaches based on two dimensions of coordination i.e., information merging and decision-making. The coordination in these approaches is based on either joint decision-making or the combination of joint decision-making and information merging. Additionally, we classify these approaches based on the type of coordination i.e., centralized or distributed.

In centralized coordination, the information sharing and decision-making about movement actions are performed either on a single UAV or at a ground station—both equipped with sufficient computing equipment and connected with all other UAVs during the mission. Centralized strategies result in an optimal global solution but require reliable, explicit and global communication and may lose efficiency or accuracy, if communication is limited. Limited communication hinders the collection of information from individual UAVs at the centralized ground station. Similarly, out-of-range UAVs cannot receive information from the centralized ground station. Centralized strategies are not optimal in such situations. The team's performance is highly sensitive to the failure of the centralized node and communication limitations. In distributed coordination, information sharing and decision-making are distributed among the UAVs. Distributed coordination increases the robustness of the team, but introduces control overhead and may lead to performance degradations due to imperfect decisions based on limited information. A detailed survey on distributed coordination of multiple vehicles for a variety of applications is available in [20].

We find that there are some multi-UAV cooperative approaches where UAVs coordinate by joint decision-making without merging of search maps (information merging). However, we couldn't find any work where UAVs coordinate by only merging of search maps and not by making joint decisions. We, thus, classify the multi-UAV cooperative search approaches on the basis of the following six types of coordination: (i) Centralized Decision-making (CD) without information merging, (ii) Distributed Decision-making (DD) without information merging, (iii) Centralized Information Merging and Centralized Decision-making (CIMCD), (iv) Centralized Information Merging and Distributed Decision-making (CIMDD), (v) Distributed Information Merging and Centralized Decision-making (DIMCD), and (vi) Distributed Information Merging and Distributed Decision-making (DIMDD). Table. 2.7 groups the existing multi-UAV cooperative search approaches according to this classification. We explain each of these classes in the following subsections.

2.3.1 Centralized Decision-Making (CD)

The approaches in this class use only centralized decision-making for planning paths of the UAVs and do not use any information merging about the environment. The authors in [102] propose a centralized decision-making approach to jointly optimize routes and sensor orientations for a team of two

Table 2.7: Summary of approaches for cooperative search of stationary targets. CD: Centralized Decision-making, DD: Distributed Decision-making, CIMCD: Centralized Information Merging Centralized Decision-making, CIMDD: Centralized Information Merging Distributed Decision-making, DIMCD: Distributed Information Merging Centralized Decision-making, DIMDD: Distributed Information Merging Distributed Decision-making, Co: Cooperative, Nc: Non-Cooperative, Ev: Evasive, Det: Deterministic, Pro: Probabilistic.

	Ref.	Coordination Algorithm	Target			UAV Altitude	Sensor		
			N	Identification	Type		Type	Error	Model
CD	25	Agent-based mobility	>1	No	Nc	Constant	Range	No	Det
	92	Optimization using dynamic graph	1	Yes			Camera		Pro
DD	33	Task assignment	1	Yes	Nc	Constant	Camera	No	Det
	105	Line formation			Ev		Range		
	116	Artificial potential force	>1	No	Nc				Pro
CIMCD	77	Voronoi partitioning	>1	No	Nc	Constant	Range	No	Pro
	79	Dynamic programming						Yes	
CIMDD	8	Co-evolutionary	>1	No	Nc	Constant	Range	No	Pro
	37	Dynamic programming					Camera		
	47	Optimization	1					Yes	
	67	Sensor management	>1						
	114	Rivaling force							
	115	Cooperative learning						No	
	118	Greedy look-ahead planning							Det
	38	Mixed integer linera programming							
DIMCD	44	Task assignment	>1	No	Co	Constant	Range	No	Pro
DIMDD	12	Data fusion	1	No	Nc	Constant	Camera	No	Pro
	18	Data fusion						Yes	
	35	Max-sum	>1					No	
	49	Average consensus						Yes	
	107	Data fusion	1						
	98	Negotiation	>1					No	Det
	106	Data fusion				Variable		Yes	Pro

UAVs searching for a mobile target. They model the environment by dynamically updating the graph whose vertices represent waypoints for the UAVs and whose edges indicate potential connections between the waypoints. They use receding-horizon optimization to share and to decide their paths for minimizing the time of search. Daniel et al. [31] examine centralized communication aware mobility algorithms for a team of UAVs to cover a given environment and to avoid redundant observations of identical regions. Their work does not support sensing limitations and information gathering tasks. They focus on channel aware mobility and on self-organizing mesh topologies of networked UAVs with respect to communication constraints. The decisions in their work are taken in a centralized manner and there is no information sharing and merging about the environment.

2.3.2 Distributed Decision-Making (DD)

The approaches in this class use only distributed decision-making for planning paths of the UAVs. They do not use any information merging about the environment or the targets. The authors in [40, 117, 129] propose a distributed coordination of decision-making among UAVs without having a search map. They generate promising results for spatial and temporal coverage of a search region. The authors in [40, 117] propose different configurations for a number of networked UAVs to exhaustively search a given area for unknown locations of targets. The approach focuses on dynamically accommodating on increasing/decreasing number of UAVs for searching the environment. Moreover, they do not focus on modeling the sensor errors and sharing the environment information among the UAVs.

The authors in [129] present distributed decision-making using artificial force vectors to coordinate the movement of UAVs. The UAVs share their location and orientation information with each other and generate a force vector, which is used to guide the UAVs. The approach enables UAVs to avoid collisions and to maximize the redundant coverage of the environment for target detection. Moreover, due to lack of global information about the environment, the approach suffer from unnecessary redundancy of information collection.

2.3.3 Centralized Information Merging and Centralized Decision-Making (CIMCD)

Sujit and Ghose [109] propose an approach where the centralized ground station is responsible for information merging and path planning of a team of homogeneous UAVs. Each UAV collects information during its flight along a predefined path. The UAVs are unable to communicate with the ground station or other UAVs during the flight. Once the UAVs return to the ground station, they deliver their collected information to the ground station. The ground station merges the delayed and partial information in a centralized search map. The ground station then assigns new paths based on the k-shortest path algorithm [61] to the UAVs and the process continues until the uncertainty in the search region drops a threshold.

Lum et al. [87] use heterogeneous UAVs with different capabilities to cover a given environment and find the location of stationary targets. Their method assumes perfect communication of the UAVs with the ground station during their flight, which avoids the problem of delayed information. The centralized ground station maintains and merges information in an occupancy based search map. The ground station generates online and instantaneous paths for all the UAVs that restrict each UAV to a distinct Voronoi partition. The method guarantees an exhaustive search of the environment. A similar approach is proposed by Mirzaei et al. [90], where a centralized assignment of heterogeneous UAVs to different Voronoi partition is performed. They use a limited look-ahead dynamic programming algorithm to find the paths of UAVs for maximizing the amount of information gathered by the whole team. One type of UAVs spreads out over the environment to optimally cover the environment, while an other type of UAVs iteratively modifies their configuration, based on the information provided by first types of UAVs, to improve the accuracy of cooperative search.

2.3.4 Centralized Information Merging and Distributed Decision-Making (CIMDD)

Much of the early work in multi-UAV cooperative search relies on centralized search maps for information merging and distributed decision-making for movement of UAVs [97, 45]. In these approaches, the UAVs can access the centralized information, which consists of information about the environment and the UAVs. Based on accessibility to this centralized information the UAVs plan their own paths in a distributed way. These approaches use artificial potential fields [127], machine learning techniques [128], group dispersion patterns [132], mixed integer linear programming [46], and evolutionary algorithms [13] as coordination mechanisms to reduce the potential overlap in the UAV paths. These approaches conclude that reducing overlap in UAV paths can improve the efficiency of cooperative search. In fact, the approaches in this category do not need sophisticated information merging strategies. They only need a method to update the centralized information base based on their sensor observations and prior information. These approaches model either a perfect sensor without any detection errors [132] or a sensor with only miss-detections [128].

Moreover, these approaches assume perfect communication between the ground station and the UAVs.

Some approaches focus on more common types of sensing limitations e.g., measurement noise [56] and detection errors of false alarms and miss detections [77]. The authors in [77] use an information-theoretic sensor management to decide the paths for UAVs in a distributed way. This sensor management directs the movement of a UAV to a region, where the expected information gain is maximized by future sensor observations. Although these methods reduce the uncertainty about the search region, they do not consider communication limitations.

2.3.5 Distributed Information Merging and Centralized Decision-Making (DIMCD)

Alvaro et al. [52] propose a centralized decision-making approach for cooperative search where each UAV distributedly maintains and merges information about the environment. UAVs share their sensor observations with each other. These observations are used to update the locally available map of the environment. The local information of each UAV consists of the number of observations performed in each sub region of the environment. The UAVs can communicate with each other and they share the environment information, however, the centralized ground station or satellite station dynamically assigns these UAVs to different regions.

2.3.6 Distributed Information Merging and Distributed Decision-Making (DIMDD)

UAVs become more autonomous when they have both the information merging and decision-making capabilities. The fully distributed multi-UAV cooperative search approaches [17, 113, 43] require UAVs to possess sensing, communication, memory, and processing capabilities. Each UAV senses the environment, updates its own search map, exchanges and merges information, and modifies its path whenever required to efficiently search the environment. Some of these methods do not consider sensing and communication limitations [17, 113, 43].

A subset of approaches in this class considers either sensing or communication limitations, but cannot work if both of these limitations are present. The authors in [110] propose a similar approach that considers the communication range limitation. One unique characteristic of this approach is that UAVs can agree on actions using mutual decision-making based on an exchange of multiple messages. Although the method assumes limitations in communication, it does not apply a realistic sensor model with sensing limitations. Another approach with fully distributed cooperation is proposed in [24] where UAVs coordinate in terms of sharing binary sensor observations. This method considers a realistic sensor model with both types of errors but does not include limitations in communications.

The work of Hu et al. [58] in this category of multi-UAV cooperative search approaches uses both the communication and the sensing (both types of errors) limitations. However, the approach focuses on consensus among UAVs to maintain similar maps on each UAV with a finite number of observations, and not on increasing the efficiency of the search operation.

2.4 Differences to State of The Art

2.4.1 Multi-robot Cooperative Observation

The difference of our work to the literature is the proposed approach for *cooperative multi-scale observation of moving targets* in outdoor environments, which has not been covered in the literature. So for, most research in cooperative observation of moving targets is based on a model of perfect sensors with fixed FOV. In our approach, the environment is modeled as quad-tree to represent 3D space and the sensor is modeled to include noise in the sensor measurement. A centralized 3D path-planning algorithm is proposed for a team of small-scale UAVs.

2.4.2 Multi-robot Cooperative Search

The first difference is the development of a method for *merging probabilistic search maps* (local information) from different UAVs in a distributed manner. The presented work shows the effects of merging probabilistic information, sensing limitations, and communication limitations on cooperative search efficiency and accuracy. It also highlights a trade-off in search time and detection errors.

The second difference is the *analytic analysis of the number of observations* required for collecting information from a given region to decide on target existance or non-existance. This allows immediate decision making about revisiting a part of the environment to save UAV battery life and search time. In addition, the parts of the environment that require more observations are iteratively predicted. To the best of our knowledge, this analytic analysis using binomial distribution and sensor parameters (miss-detection and false alarm rate) along with the iterative prediction for parts of the environment to be visited in decision-making has never been presented in the literature.

Finally the *analysis of design options* considering centralized or distributed decision making for information merging and path-planning by iterative use of Travelling Salesman Problem (TSP) and/or Multiple Travelling Salesmen Problem (MTSP) algorithms. The existing approaches present a specific solution for a given scenario and do not explore various options of coordination with respect to sensing, time, and communication limitations.

2.5 Summary

In this chapter, we organized, critically discussed and compared works from the last 20 years in the area of cooperative mobile robots for surveillance of stationary and moving targets. We identified the basic actions performed by each robot and five factors that affect the design and performance of cooperative mobile robots. It is important to notice that most related works on the topic are based on simulation and lab studies, which may be sometimes far from real-life applications. We group existing approaches of cooperative observation of moving targets based on four major application scenarios, namely Cooperative Tracking (CT); Cooperative Multi-robot Observation of Multiple Moving Targets (CMOMMT); Cooperative Search, Acquisition, and Track (CSAT); and Multi-robot Pursuit Evasion (MPE). We classify the existing multi-UAV cooperative search approaches based on two dimensions of coordination i.e., information merging and decision-making. Additionally, we classify these approaches based on the type of coordination i.e., centralized or distributed.

Chapter 3

Problem Formulation of Multi-UAV Search and Observation

3.1 Overview

This chapter limits the general problem of multi-robot surveillance (as mentioned in Section 2.1) to multi-robot aerial surveillance. It defines a task of multi-robot aerial surveillance i.e., multi-UAV search and observation. In multi-UAV search and observation, a team of resource-limited small-scale UAVs e.g., quad-rotors searches for and observes a set of targets in a given environment. The key problem is to coordinate these UAVs for completing the task of search and observation. This coordination can be centralized or distributed and consists of other problems. The first problem is the representation and selection of local information about the environment and targets that a UAV shares with other team members and/or the centralzied ground station. The second problem is the merging of (parts or complete) shared local information by a UAV to determine the global information about the environment and targets. The final problem is the selection of paths for the UAVs to efficiently explore the environment.

We focus on two instances of multi-UAV search and observations: (i) cooperative search (Section 2.2) and (ii) cooperative observation (Section 2.3). In cooperative search, the objective is to minimize the time of finding the locations of stationary targets. The objective of cooperative observation is to maximize the collective time and quality of observations of moving targets that are larger in number than the UAVs.

The rest of the chapter is organized as follows. Section 3.2 describes all the

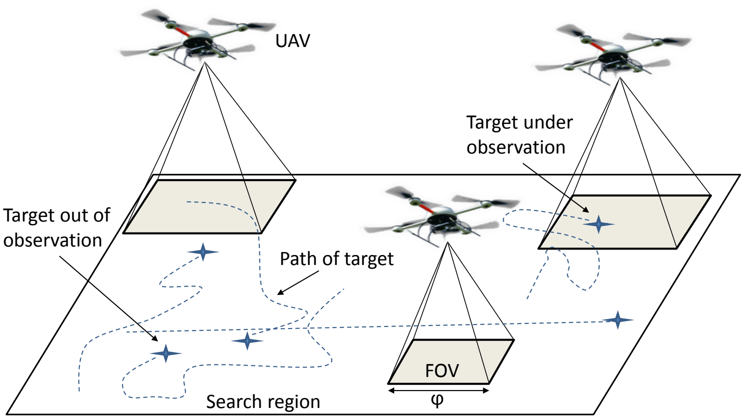


Figure 3.1: Observing multiple moving targets using cooperative UAVs.

assumptions and formulates the problem. Section 3.3 explains the objective functions of cooperative search and cooperative observation.

3.2 Problem Formulation

3.2.1 Environment

Inspired from [38, 58, 76] the rectangular environment $\Omega \in \mathbb{R}^2$ is represented by $L \cdot W$ equally-sized, disjoint cells, where L and W represent the number of rows and the number of columns, respectively. Each cell is identified by $c = (x, y)$, where $x \in \{1, 2, ..., L\}$ and $y \in \{1, 2, ..., W\}$ are the coordinates of its center. Thus, $\mathbf{C} = \{1, 2, 3, ..., LW\}$ represents the set of cells in the discretized environment. A search map, which is a grid of LW cells can be maintained either in a centralized or distributed way.

3.2.2 Target

Let $\mathbf{T} = \{T_1, T_2, ..., T_B\}$ be the set of B targets present in Ω. The number of targets is assumed to be known and constant during the mission. These non-cooperative, non-evasive and uniquely identifiable targets can be stationary or moving.

Stationary Target

A target is assumed to occupy at most a single cell in Ω and a cell is allowed to have at most one target. A cell is termed as target cell if it contains a target and as empty cell if it does not. The occupancy probability [38] in cell c at discrete time steps t is denoted as P_c^t. This occupancy probability is modeled as a Bernoulli distribution, i.e., the event $X_c = 1$ (a target is present in cell c) has a probability P_c^t and the event $X_c = 0$ (no target is present in cell c) has a probability $1 - P_c^t$. The target location is discretized in space i.e., cell and definite knowledge about target existence or absence in a cell c is represented as $P_c^t = 1$ or $P_c^t = 0$, respectively. No knowledge about target existence in cell c is thus represented as $P_c^t = 0.5$ [38]. It is assumed that no prior knowledge is available about Ω or the locations of targets, which may require the UAVs to search every cell of Ω at least once. Cell c is considered as containing a target if $P_c^t \geq \Theta^+$ and as an empty cell if $P_c^t < \Theta^-$, where Θ^+ and Θ^- are predefined thresholds.

Moving Target

The state of the j^{th} target (T_j) at time t is denoted by

$$\mathbf{x}_j^t = (x_j^t, \dot{x}_j^t, y_j^t, \dot{y}_j^t), \tag{3.1}$$

where (x_j^t, y_j^t) and $(\dot{x}_j^t, \dot{y}_j^t)$ are the position and velocity of the target. The motion of T_j is

$$\mathbf{x}_j^{t+1} = \Phi \mathbf{x}_j^t + \gamma, \tag{3.2}$$

where Φ is the state transition matrix with process noise $\gamma \backsim \mathcal{N}(0, \mathbf{Q})$ and process noise covariance matrix $\mathbf{Q}$. The movement of the targets is independent of each other.

3.2.3 UAVs

Let $\mathbf{U} = \{U_1, U_2, ..., U_A\}$ be the set of A homogeneous and synchronized UAVs moving above the Ω in discrete time step t. The state of the i^{th} UAV (U_i) at time step t is

$$\mathbf{y}_i^t = (x_i^t, y_i^t, z_i^t), \tag{3.3}$$

where z_i^t represents the altitude of U_i at time step t. The x_i^t and y_i^t components of $\mathbf{y}_i^t$ coincide with x and y components of a cell $c \in \mathbf{C}$. We assume that more than one UAVs can go to the same location and have the same state. Each UAV is equipped with (i) a position sensor which facilitates the UAV to know its position within the resolution of a cell at any time, (ii) a surveillance sensor for observing Ω, (iii) a wireless communication unit for exchanging information with the ground station and with other UAVs in the team, and (iv) a computing unit for performing local search map updates. Each UAV U_i has its own local search map Ω_i of the Ω. At time step t, each UAV executes the following three actions: take observation, receive new location for movement, and move to the new location.

3.2.4 Observations

The observation depends on the surveillance sensor's FOV and the target. The surveillance sensor of U_i consists of a downward-looking camera with a fixed zoom level. For simplicity we consider the FOV $F_i \in \Omega$ to be a square with side of length φ (Fig. 3.1). We define different observations for stationary and moving targets.

Observations for Stationary Targets

The sensor observation by the surveillance sensor of U_i in cell c at time step t is denoted as $O^t_{i,c}$. Assuming that multiple observations are independent of each other, two observation results are defined for each cell, i.e., $O^t_{i,c} = 0$ (negative observation) or $O^t_{i,c} = 1$ (positive observation). Depending on the target's true presence or absence and the made observation, the following probabilities are defined [58, 25, 77]:

$$\begin{aligned} P(O^t_{i,c} = 1|X_c = 1) = p, P(O^t_{i,c} = 0|X_c = 1) = 1 - p \\ P(O^t_{i,c} = 1|X_c = 0) = q, P(O^t_{i,c} = 0|X_c = 0) = 1 - q \end{aligned} \tag{3.4}$$

where sensor parameters $p, q, 1-p$ and $1-q$ denote probabilities of detection, false alarm, false miss and true miss, respectively. Considering an informative sensor with $0.5 < p < 1$ and $0 < q < 0.5$, we assume that only one observation per cell can be taken at a single time step and the FOV of surveillance sensor coincides with a single cell. The number of observations in a given cell c is denoted by m_c.

Observations for Moving Targets

For moving targets, we relax our assumptions on sensing limitations. The sensing limitations in this case include variable FOV (depends on altitude of the sensor) and measurement noise in reporting the location of a target. We do not assume miss detections and false alarms for moving targets. The FOV of U_i, denoted as F_i, depends on its altitude[1]. A target is considered under observation when it is in the FOV of at least one UAV. The observation of T_j by U_i at time t is defined as

$$O^t_{ij} = \begin{cases} 1 & \text{if } (x^t_j, y^t_j) \in F_i \\ 0 & \textit{otherwise.} \end{cases} \tag{3.5}$$

[1]The method is equally applicable to constant-altitude UAVs with variable zoom levels.

A single UAV can observe multiple targets and a single target can be observed by multiple UAVs. However, the observation of a single target by multiple UAVs at time t is of no advantage as we are not interested in depth perception, in multi-view analysis or in improving the estimate of the target position. We use the OR operator [95] to show that observation by a single UAV is sufficient:

$$\bigvee_{i=1}^{A} O_{ij}^{t} = \begin{cases} 1 & \text{if } \exists i : O_{ij}^{t} = 1 \\ 0 & otherwise. \end{cases} \tag{3.6}$$

The UAV U_i collects a measurement $\mathbf{z}_{ij}^t$ for target T_j under its observation. The measurement $\mathbf{z}_{ij}^t$ consists of the $\mathbf{x}_j^t$. At time t, there can be $n_i^t \leq B$ targets in F_i and thus n_i^t measurements can be generated. A measurement $\mathbf{z}_{ij}^t$ is generated by the following model

$$\mathbf{z}_{ij}^t = \mathbf{H}\mathbf{x}_j^t + \vartheta, \tag{3.7}$$

where $\mathbf{H} = (1\ 0\ 0\ 0; 0\ 0\ 1\ 0)$ is the observation matrix with observation noise $\vartheta \backsim \mathcal{N}(0, \mathbf{O})$ and observation noise covariance matrix $\mathbf{O}$. The sensing errors (Eq. 3.7) do not allow a UAV to know the exact location of a target that is under its observation. The states of all the targets, all the UAVs, and the measurements for all the targets are denoted as $\mathbf{X}^t = \{\mathbf{x}_1^t, \ldots, \mathbf{x}_B^t\}$, $\mathbf{Y}^t = \{\mathbf{y}_1^t, \ldots, \mathbf{y}_A^t\}$ and $\mathbf{Z}^t = \{\mathbf{z}_1^t, \ldots, \mathbf{z}_B^t\}$, respectively

3.2.5 Movement and Path Planning

The movement of U_i is discretized in space (*cells*) and time (*time step*). It is assumed that U_i has sufficient battery (life) or flight time to complete the mission. The UAV U_i makes movement decisions only at discrete time intervals referred to as time step. The duration of a single time step t is sufficient for U_i to move from its location to an adjacent location, to take a sensor observation, to update its local information, and to exchange information with other UAVs.

In cooperative search, the discretized movement of U_i is represented as $\mathbf{y}_i^t = c_i^{t+1} \in \{(x+1, y)_i^t, (x-1, y)_i^t, (x, y+1)_i^t, (x, y-1)_i^t\}$, where c_i^{t+1} (location of U_i at time step $t+1$) must stay within the boundary of Ω. This representation does not include the altitude of UAVs as it is constant. UAVs cannot move to diagonal cells to synchronize the time and distance of movements. Each UAV U_i traverses a path $\mathbf{r}_i$ which is an ordered sequence of cells, and $\mathbf{R} = \{\mathbf{r}_i : 1 \leq i \leq A\}$ represents the set of search paths for all UAVs in $\mathbf{U}$. The cells constituting a search path for a UAV are considered as waypoints and determine the sub-regions where observations should be made to gather information. The paths in $\mathbf{R}$ can be pre-defined static or dynamically calculated during the search mission. For the pre-defined static paths, we assume standard sweep model or lawnmower-type search [22]. In the sweep mobility model, a UAV moves from one boundary to the other boundary of Ω in a single row. It then jumps to the next row and starts movement in reverse order to completely traverse that row. The UAV continues this movement to visit each cell of the Ω.

In cooperative observation for moving targets, UAVs can move in $3D$ space and the speed of each UAV is higher than that of the fastest target. The variation in altitude causes variation in the size of FOV and quality of observation. Increasing the altitude of UAV increases its FOV but reduces the quality of observation. However, there is a threshold z_0 on the minimum allowed altitude of a UAV.

3.2.6 Communication

Following a discussion [7] on the communication requirements of UAV networks, we simplify the assumptions that suit to the problem of cooperative search. Let the communication range among UAVs, measured by the Euclidean distance, be limited to r cells. Thus, information can only be exchanged when the UAVs are within distance r. Communication is considered free of any delays or failures, once the UAVs are within range r. Let $\mathbf{N}_i = \{U_l \in \mathbf{U} : l = 1, ..., A \wedge \|\mathbf{y}_i - \mathbf{y}_l\| \leq r\}$ be the set of UAVs that are within the communication range r of U_i ($\mathbf{N}_i \subseteq \mathbf{U}$) and $1 \leq |\mathbf{N}_i| \leq A$. Note that $\mathbf{N}_i = \{U_i\}$ and $|\mathbf{N}_i| = 1$, if $r = 0$ or $\|\mathbf{y}_i - \mathbf{y}_l\| > r$ for $i \neq l$. We relax the assumptions on communication range limitations for cooperative observation and assume that UAVs can always communicate.

3.2.7 Coordination

The UAVs coordinate in terms of sharing information and decisions about the search action. Primarily, each UAV updates its own local information without coordination with other UAVs (cp. Section 4.2). Due to different UAV locations, errors in the surveillance sensor, number of visits to a given cell and especially limited communication range, the UAVs may have different local information. The UAVs coordinate by exchanging and merging individual local information to best represent the situation in the environment. The UAVs can communicate with the ground station whenever needed. In addition to the exchange of information, the UAVs also coordinate the decision-making about their paths.

3.3 Objectives

3.3.1 Objective of Cooperative Search

The objective of each UAV is to find the locations of B targets as fast as possible. Let the number of observations m_c increase as the time spent by a UAV in cell c increases. The objective function of each UAV can be defined as

$$\text{minimize } t : B = \sum_{c \in \mathbf{C}} f(c,t) \tag{3.8}$$

where

$$f(c,t) = \begin{cases} 1 & \text{if } P_c^t \geq \Theta^+ \\ 0 & \text{otherwise} \end{cases} \tag{3.9}$$

and

$$P_c^t \rightarrow \begin{cases} 1 & \text{if } m_c \rightarrow \infty \text{ and } c \text{ contains a target} \\ 0 & \text{if } m_c \rightarrow \infty \text{ and } c \text{ is empty} \end{cases} \tag{3.10}$$

The constraint in Eq. 3.10 shows that the value of P_c^t depends on target existence and the observations made by a UAV in cell c. Increasing the number of observations in cell c also increases the time spent in that cell. Once this objective is achieved by any of the UAVs or the ground station, the search is finished. To achieve this objective, the UAVs coordinate by performing two tasks: (i) information merging, and (ii) decision-making.

In case of limited communication, the UAVs have inconsistent local information, as complete and updated information is not available to each UAV. This means that each UAV may represent different occupancy probabilities for a given cell. Individual probabilities by various UAVs for a given cell should be merged to calculate a probability that best represents information about the target existence in that cell. Utilizing the information from other teammates a UAV can improve the search in two ways: (i) by increasing its observability of the environment by considering other UAVs' observations and (ii) by improving its knowledge in a given cell by merging probabilities in that cell by other UAVs. The key question is how to combine information from one UAV with information from other UAVs so that the team can work together to locate the target in minimum time with minimum location errors. Several information merging techniques to maintain maps need to be investigated where the amount and type of information exchanged between the UAVs vary. Chapter 4 describes the information merging for cooperative search in detail.

Once a UAV updates its map, it needs to select a search action, which is a combination of two sub-actions: (i) how many observations to take in a given cell and (ii) which cells to visit and in which order (path). The first sub-action requires determining the number of observations, based on the initial observation and prior information, in order to declare the cell as empty or occupied with the target. In other words, a UAV must know the time or efforts in fully exploring that cell. The second sub-action requires a UAV to predict the cells and the order of visiting those cells, which increase the chances of containing targets. A dynamic path planning approach, which includes both of these sub-actions, can increase the performance of multi-UAV cooperative search. Decision-making should answer the questions of how to predict the cell with more chances of target existence, how to plan the path to visit the predicted cells, and how much time a UAV should spend in gathering information from a given cell. Chapter 5 presents details of decision-making in cooperative search.

3.3.2 Objective of Cooperative Observation

The objective of cooperative observation is inspired from the standard CMOMMT [96, 93, 35, 78, 120] problem. CMOMMT maximizes the collective time of observation represented by the following objective function

$$\Upsilon = \sum_{t=0}^{E} \sum_{j=1}^{B} \bigvee_{i=1}^{A} O_{ij}^t, \tag{3.11}$$

where E is the total time of the mission. We extend the CMOMMT problem formulation to include variable resolution of observations and measurement noise. We refer to this problem as *multi-scale observation of multiple moving targets*.

A higher value of the altitude z increases the FOV but reduces the resolution of observation (i.e., the spatial scale the target is being observed at). The variable resolution observations can also improve the movement decisions of the UAVs in order to maximize the number of targets under observation. Reducing the value of z from the surface of the environment improves the resolution of observation. However, we set a minimum allowed altitude z_0, as reducing altitude of the UAV below z_0 may cause the UAV to hit the target.

The resolution of observation of T_j by U_i at time t is defined as

$$s_{ij}^t = \begin{cases} \frac{1}{z_i^t} & \text{if } (x_j^t, y_j^t) \in F_i^t \\ 0 & otherwise. \end{cases} \tag{3.12}$$

In case, multiple UAVs are observing T_j with different resolutions, the resolution used is given as

$$\hat{s}_j^t = max\{s_{1j}^t, ..., s_{Aj}^t\}. \tag{3.13}$$

Table 3.1: Duration of observation and resolution of observation for B targets, mission duration of E time steps, and highest resolution $1/z_0$ (z_0 is lowest altitude).

	$E/2$	E
Lowest resolution ($\alpha = 1$)	$g = 0.5$	$g = 1$
Highest resolution ($\alpha = 0$)	$g = 1/z_0$	$g = 1/2z_0$

In addition to maximizing the collective time of observation, we want to maximize the collective resolution of observations.

Maximizing the collective resolution of observation of the targets under observation corresponds to maximizing the following objective function:

$$\Psi = \sum_{t=0}^{E} \sum_{j=1}^{B} \hat{s}_j^t. \tag{3.14}$$

With a limited number of UAVs (i.e., $A < B$) not all targets might be under observation and it is not possible to observe all the targets with high resolution all the time. The goal thus becomes to maximize

$$g = \frac{1}{BE}\left(\alpha\Upsilon + (1-\alpha)\Psi\right), \tag{3.15}$$

where $g \in [0, 1]$, $g = 0$ implies that no target is under observation throughout the mission and $g = 1$ implies that all the targets are under observation with the desired resolution throughout the mission. The parameter α assigns a priority weight or importance to the resolution of observation. Setting $\alpha = 1$ makes the problem as a standard CMOMMT problem with constant FOV and no interest in high resolution observations.

In Table I, we provide some numeric values of g for B targets and mission duration of E time steps. These values are calculated by putting Υ (Eq. 3.11) and Ψ (Eq. 3.14) in Eq. 3.15 for two different values of E and α. Setting $\alpha = 0$ means that we want to maximize the collective resolution of observations of the targets that are currently under observation. Note that it is difficult to get $g = 1$ for $\alpha = 0$, as targets will easily escape the smallest FOV. The multi-scale multi-UAV coverage problem at hand is dynamic and, at each time step, the coordinated movement approach should determine which UAVs observe, the part of the environment to observe, and the resolution of observation. We focus on developing a centralized cooperation and movement strategy for a team of UAVs to maximize g.

3.4 Summary

This chapter defines a specific problem of multi-UAV search and observation. The chapter describes all the assumptions made for a multi-UAV system and presents mathematical notations for the environment, the target, the UAV and the sensor. It provides a mathematical formulation for the two objectives corresponding to two applications of multi-UAV search and observation.

Chapter 4 and Chapter 5 explain cooperative search and Chapter 6 is dedicated to cooperative observation. In Chapter 4, we use the concept of occupancy probability and Bayesian analysis to represent information about the environment and the target existance. We then analyze different strategies for merging of shared information. We also show the effects of sensor and communication parameters on minimizing the time and errors of cooperative search. In Chapter 5, we provide analytic analysis of the number of observations required to decide the existence or absence of a stationary target. We then use this required number of observation in TSP and/or MTSP formulation to decide the paths of the UAVs. Additionally, the chapter explores the algorithmic design space by presenting four algorithms for different coordination scenarios. Chapter 6 presents the quad-tree based modeling of the environment and UAV motion. The chapter also presents a centralized coordination algorithm for movement of UAVs to increase the observation time and quality of moving targets.

Chapter 4

Information Merging in Multi-UAV Cooperative Search

4.1 Overview

This chapter discusses the local information of a UAV and merging of information (as mentioned in Subsection 3.3.1) contributed by other UAVs to perform a cooperative search. The chapter is adopted from [73]. As shown in Fig. 4.1 multi-UAV cooperative search is performed in three steps: (i) sensing the environment, (ii) sharing and merging of information, and (iii) making joint decisions for movement. In first step, each UAV observes a part of the environment, using its surveillance sensor, and updates its local information about the environment. The local information is updated using observation of the environment and prior information available about the environment. We use the observation model (defined in Chapter 3) and a Bayesian update rule to perform this step. In the second step, each UAV broadcasts its local information to share with the other team members. Each member of the team merges the received information with its local information to improve its perception of the environment. The sensing and communication limitations, however, cause each UAV to receive potentially incomplete, erroneous, or outdated information. Finally, the UAVs decide their next moves in the environment. In this chapter, we explain only the first two steps of multi-UAV cooperative search.

In cooperative search, the information of a UAV consists of a search map (as described in Section 2.3 and Section 3.2), which is updated in two ways: (i) uncoordinated search map update based on sensor observation, and (ii) information merging or coordinated search map update, which is the first task of coordination among the UAVs (cf. Subsection 3.3.1).

At the beginning of the search mission, U_i initializes its local search map Ω_i

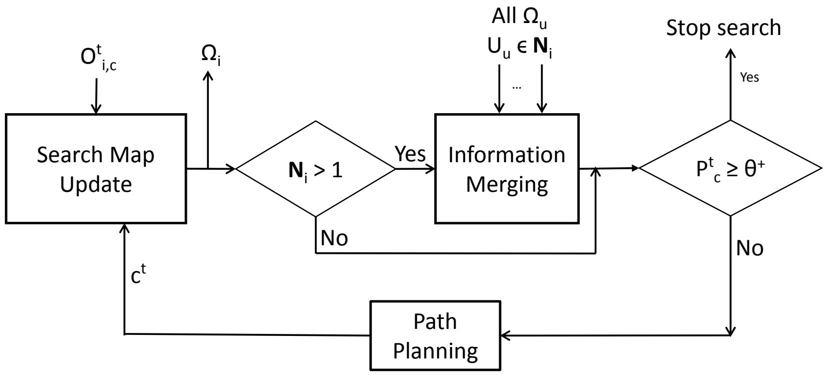


Figure 4.1: Processing diagram for cooperative search of an individual UAV U_i.

with $P^0_{i,c} = 0.5$ for $i = 1, ..., A$ and $c \in \mathbf{C}$ to represent complete uncertainty or lack of prior knowledge about the search region Ω. It then starts taking sensor observation at its current location $\mathbf{y}^t_i$. Based on the sensor observation $O^t_{i,c}$ and prior probability $P^{t-1}_{i,c}$ in the current location c, U_i updates the occupancy probability to $P^t_{i,c}$ in its own search maps Ω_i. This *uncoordinated* map update by U_i depends on the detection errors (detection and false alarm probabilities) of the surveillance sensor on board the UAV.

The UAV U_i then broadcasts the updated information to neighboring UAVs $\mathbf{N}_i$ in the team. The number of UAVs in set $\mathbf{N}_i$, $1 \leq |\mathbf{N}_i| \leq A$, depends on the communication range of U_i. Depending on the communication range, the UAVs in the team now have different search maps at time t. These search maps represent partial information contributed by neighboring UAVs. The difference in search maps is caused by the variation in the occupancy probability values of some cells. Different occupancy probability values corresponding to a given cell are merged to determine a new occupancy probability value that best represents the existence of a target. Information merging iteratively performes this step for each cell $c \in \mathbf{C}$ to merge the search maps. The UAV U_i then moves to another cell according to its mobility model and continues the process. The process is depicted in Fig. 4.1, which is executed by each UAV at every time step. The search is finished when any of the A UAVs identifies a cell c with $P^t_c \geq \Theta^+$, where Θ^+ is a predefined detection threshold to stop the search. In this chapter, we discuss the uncoordinated search map update and information merging (map merging).

The rest of the chapter is organized as follows. Section 4.2 describes the method of uncoordinated search map update for updating the local information of a UAV. Section 4.3 explains our proposed work on information merging. In Section 4.4, we present the simulation results to show the effects of information merging on multi-UAV cooperative search.

4.2 Search Map Update

As shown in Fig. 4.1, primarily, each UAV updates its own search map using its own sensor observations and prior information. At each time step t, a UAV U_i observes some part of the search region, which corresponds to a single cell c in the search map. The UAV then updates cell c in its search map Ω_i by using the observation and prior information in cell c. The following Bayesian rule [135, 58] is a common tool to calculate the updated probability in cell c using the sensor characteristics (p and q), sensor observation $O^t_{i,c}$ and prior probability $P^{t-1}_{i,c}$ in cell c.

$$P^t_{i,c} = \begin{cases} \frac{pP^{t-1}_{i,c}}{pP^{t-1}_{i,c} + q(1-P^{t-1}_{i,c})} & \text{if } O^t_{i,c} = 1 \\ \frac{(1-p)P^{t-1}_{i,c}}{(1-p)P^{t-1}_{i,c} + (1-q)(1-P^{t-1}_{i,c})} & \text{if } O^t_{i,c} = 0 \end{cases} \tag{4.1}$$

Thus, if c contains a target $P^t_c \to 1$ as $m_c \to \infty$, and if c is empty $P^t_c \to 0$ as $m_c \to \infty$ [58]. It can be shown from Eq. (4.1) that $P^t_{i,c} = 1$ if $P^0_{i,c} = 1$ and $P^t_{i,c} = 0$ if $P^0_{i,c} = 0$ for all $t > 0$. If $p = 0$, $P^t_{i,c}$ becomes 0 once U_i gets a sensor observation equal to 1, and will remain unchanged regardless of future observations. We consider an informative sensor with $0.5 < p \leq 1$, and $0 \leq q < 0.5$. Fig. 4.2 shows the relationship of q, Θ^+ and the required number of observations in a given target cell c to declare that the cell contains a target. In Chapter 5, we will derive analytic expressions for calculating the required number of observations in a cell.

4.3 Information Merging

Once U_i updates its search map (local information) Ω_i using sensor observation and prior information, it exchanges the search map with neighboring UAVs $\mathbf{N}_i$. If no other UAV is in the communication range of U_i, U_i has no neighbors and $|\mathbf{N}_i|$ becomes 1. In a given cell c, U_i may now receive different values for P^t_c which are merged to determine a new value for $P^t_{i,c}$ that best represents the collected information about target existence in c. Depending on r, U_i can now have $1 \leq n \leq A$ occupancy probability values for a given cell c at time t. If $r = 0$, then $|\mathbf{N}_i| = 1$ and cell c has only one ($n = 1$) value contributed by the local search map Ω_i of U_i. If the communication range is limited, the neighbors $\mathbf{N}_i$ may have different values for a given cell c and U_i can now have at most ($n = A$) values for cell c. In the case of unlimited communication, there are $n = 2$ values for each cell, one contributed by Ω_i and the other by Ω_u, where $U_u \in \mathbf{N}_i$. In case of unlimited communication, the UAVs have consistent maps,

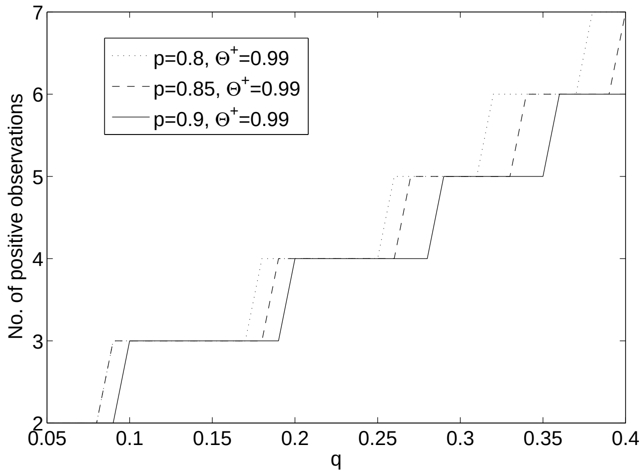


Figure 4.2: Minimum number of observations required in a single cell to satisfy condition $P_c^t \geq \Theta^+$. It corresponds to the best case, i.e., when a target is present in the visited cell and the observation is always positive ($O_{i,c}^t = 1$).

as complete and updated information is available to each UAV. The UAVs require a strategy to merge these different probabilities in a single probability for increasing the performance of the search.

In order to maintain the timeliness of the occupancy probabilities, a simple time stamping mechanism is introduced. Whenever P_c^t is updated by U_i, the time stamp $\tau_{i,c}$ of this update is captured. If the update is caused by an observation of the cell, the current time step is captured as a time stamp. If the update is caused by merging cell values from different UAVs, the most recent time stamp among the contributing cell values is taken as the new time stamp. The time stamps are stored in the search maps and are exchanged together with the probability values of the cells. Map merging is only performed in those cells that have different time stamps with respect to the neighboring UAVs.

Obviously, when $|\mathbf{N}_i| = 1$, U_i does not receive information from other UAVs and there is no information merging stage. UAV U_i simply uses the uncoordinated occupancy probability in the search map[1]. If $|\mathbf{N}_i| > 1$, U_i receives n occupancy probability values for cell c represented as $\mathbf{P} = \{P_{i,c}^t, P_{u,c}^t : u \neq i \wedge U_u \in \mathbf{N}_i\}$. Similarly, U_i receives n time stamps for cell c represented as $\mathbf{t} = \{\tau_{i,c}, \tau_{u,c} : U_u \in \mathbf{N}_i\}$. The information merging method can be expressed as

$$P_{i,c}^t = f(\mathbf{P}, \mathbf{t}). \tag{4.2}$$

Fig. 4.3 represents a small search region with a single target and the local

[1]For the sake of simplicity, we do not distinguish between uncoordinated and merged occupancy probabilities throughout the remainder of this chapter.

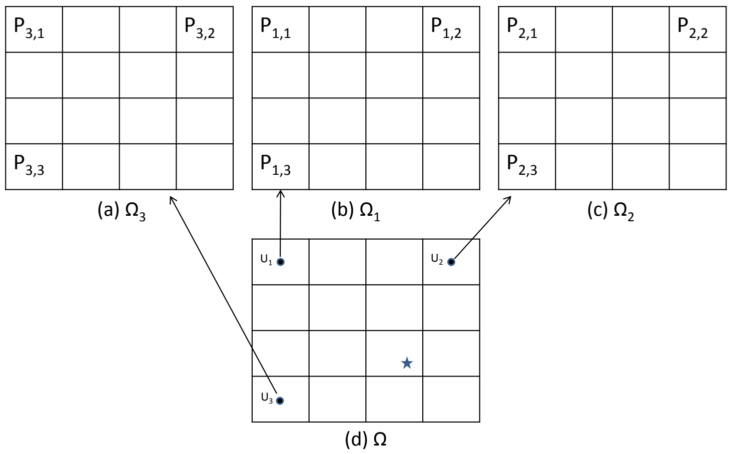


Figure 4.3: Local 4×4 search maps of three UAVs (a, b and c) and the environment (d). UAVs are marked with the dots and the target location is marked with a star. The local occupancy probability of U_i at the location of U_u is denoted as $P_{i,u}$.

4×4 search maps of three UAVs having unlimited communication. Fig. 4.4 shows the information contents of U_1 after exchange of information with all other UAVs. Sharing and merging of information results in at most A cell updates in each individual search map. To avoid confusion, we use the notation $P_{i,u}^t$ to represent the occupancy probability of U_i in Ω_i at the location of U_u where $u = 1, 2, \ldots, A$. As indicated in Fig. 4.1 there are two different updates performed by each UAV at each time step: uncoordinated *search map update* and *information merging* i.e., merging of search maps.

There are different methods for merging probability values in $\mathbf{P}$ to calculate a new value for P_c^t. We propose four strategies to merge information from multiple UAVs. These merging strategies are: (i) belief update, (ii) average, (iii) modified occupancy grid map merging, and (iv) sensed data sharing. We consider different types of information and communication limitations. We start with *unlimited* communication and then elaborate on the modifications required for efficient implementation of each strategy under limited range condition.

4.3.1 Belief Update (BU)

The belief update merging strategy simply replaces the outdated information with the most recent information. The occupancy probability associated with the latest time step is given more importance to prioritize the most recently collected information. UAV U_i declares the occupancy probability $P_{i,c}^t$ in a given cell c of Ω_i outdated if $\tau_{i,c} < max(\mathbf{t})$. It means that another UAV has recently updated the occupancy probability in cell c. UAV U_i then updates

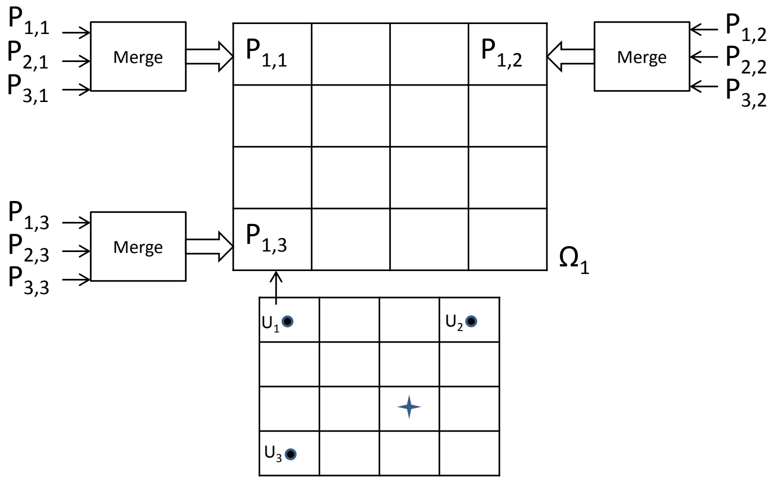


Figure 4.4: The search map merging at U_1 having unlimited communication.

$P^t_{i,c}$ by replacing it with a value from $\mathbf{P}$ with the most recent corresponding time stamp. The UAV U_i also updates the time stamp $\tau_{i,c}$ by $\tau_{i,c} = max(\mathbf{t})$. In belief update the merging strategy, (Eq. 4.2) becomes

$$P^t_{i,c} = \{P^t_{u,c} : \tau_{u,c} = max(\mathbf{t})\}. \tag{4.3}$$

4.3.2 Average (AVG)

The average merging strategy takes the arithmetic mean of all the probability values in $\mathbf{P}$ to calculate a new value for $P^t_{i,c}$. The UAV U_i then overwrites $P^t_{i,c}$ by this mean value. The UAV U_i updates its map Ω_i by

$$P^t_{i,c} = \begin{cases} P^t_{u,c}, & n = 2 \text{ and } P^t_{i,c} = 0.5 \\ \frac{1}{n}(P^t_{i,c} + \sum_{u=1}^{n-1} P^t_{u,c}) & \text{otherwise} \end{cases} \tag{4.4}$$

where n depends on the communication range. If the communication range is unlimited, all UAVs become neighbors of U_i, which makes $|\mathbf{N}_i| = A$. In case of unlimited communication and distinct locations of all UAVs, a cell c can get at most two different values ($n = 2$): one from U_i and the other from a UAV that is visiting cell c. If two different values of occupancy probability are available for a given cell c and U_i has no prior information for cell c ($P^t_{i,c} = 0.5$), then U_i replaces $P^t_{i,c}$ by the occupancy probability provided by neighboring UAV U_u. If the communication range is limited, cell c may have A different occupancy probability values. This variation is caused by updates of c by different UAVs at different time steps. In this case $n \leq A$ is equal to the number of UAVs with different values, which are stored in $\mathbf{P}$, for c. If more than two values of occupancy probability are available for cell c, then U_i replaces $P^t_{i,c}$ by the average of occupancy probability values provided by $\mathbf{P}$.

4.3.3 Weighted Integration of Occupancy Grids (WIOG)

Integrating Occupancy Grids [18] is a well-known technique used in simultaneous localization and mapping (SLAM). This technique is not applicable in its original form to the cooperative search. We propose a modified version of this technique, which is applicable to the cooperative search. The original integration rule is given as

$$P_c^{iog} = \frac{odds_c}{1 + odds_c} \tag{4.5}$$

$$odds_c = \prod_{u=1}^{n} odds_{u,c} \tag{4.6}$$

$$odds_{u,c} = \frac{P_{u,c}}{1 - P_{u,c}} \tag{4.7}$$

where P_c^{iog} is the probability of occupancy at c calculated by integrating occupancy grids (IOG)[2]. It models an environment with a grid where the occupancy probability at each cell of the grid represents information about an obstacle. In conventional IOG, robots determine obstacle cells that need to be avoided during the movement of robots.

We use the occupancy probability to model the existence of a target in a cell. We use IOG to determine the cells that may contain target and need to be visited frequently. The IOG method aims to reinforce cell values and thus reaches low or high probability values very fast. This property supports quick decision making but results in a considerable amount of detection errors, if the repetitive observations include false alarms and false negatives. Table 4.1 shows an example of this problem for specific values of $p = 0.9$ and $q = 0.1$, where at time t_1 one UAV receives a false alarm from its sensor and the other UAV has no observation at cell c. The merging results in updating both the maps at $P_{1,c}$ and $P_{2,c}$ with value 0.9. Later at t_2, one of the UAVs observes a true negative at cell c but merging of values results in no change in both the maps. The probability of occupancy at c is now fixed to 0.9 and cannot be reduced by even infinite numbers of correct observations in that cell. The detection of another false alarm at c will further increase the value of P_c leading to exceed the threshold value and will terminate the search with an erroneous result. Thus, IOG in its original form is not suitable for cooperative search scenario.

[2]This rule is adopted from SLAM where robots develop partial maps using occupancy grids and integrate the partial occupancy grids at the end of the SLAM process by using Eq. (4.5).

Table 4.1: Merging occupancy probabilities in a given cell multiple times using integrating occupancy grids method.

Time	$O_{1,c}$	$O_{2,c}$	$P_{1,c}$	$P_{2,c}$
t_0	–	–	0.5	0.5
t_1	1 (false alarm)	–	0.9	0.9
t_2	–	0 (true negative)	0.9	0.9

The effect of this problem can be reduced if we restrict the output of the IOG technique to change slowly. In order to do so, we combine the average value of occupancy probabilities at c and occupancy value using IOG at c by a weighted average. We call this rule as Weighted Integration of Occupancy Grids (WIOG), which is given as

$$P_{c_i} = w(P_c^{avg}) + (1-w)(P_c^{iog}) \tag{4.8}$$

where

$$P_c^{avg} = \frac{1}{n}\sum_{u=1}^{n} P_{u,c} \tag{4.9}$$

The weight w can be chosen based on the sensor parameters and search constraints.

4.3.4 Sensed Data Merging (SDM)

Instead of sharing probability values, the UAVs can share their current locations and sensor observations with each other. In this strategy, each UAV keeps a record of sensor observations for each cell in the search region and updates the P_c^t iteratively based on the total number and type of observations in cell c. The strategy enables UAVs to share full information but requires more memory, computation power and bandwidth if surveillance sensors are heterogeneous with different characteristics (p and q). The updated probability in cell c can be calculated by iteratively using Eq. (4.1) for all consecutive observations from all UAVs. Observe that adding time stamps (i.e., history of observations) to the search map increases the information to be exchanged and processed by the UAVs.

4.4 Simulation Results

To evaluate the effectiveness of our proposed merging strategies, we simulate a search region of $L \times W = 10 \times 10$ cells with a single stationary target located at

$(6, 7)$. We initialize the location of up to $A = 5$ UAVs at randomly selected cells and consider a standard sweep model for the mobility of UAVs. We consider $\Theta^+ = 0.99$ which means the search is finished if one of the UAVs finds a cell c in its own map with $P_c^t \geq 0.99$ and that cell is designated as location of the target. If the result of the search is a cell other than $(6, 7)$, we record a detection error. We perform simulations to compare the results of our proposed strategies in case of no communication, limited communication and full communication among UAVs. We use the communication range r in terms of cells (the unit of r is cells) and consider two UAVs in range when the Euclidean distance between them is less than or equal to the specified communication range. All results are based on $N = 1000$ runs of simulations and $w = 0.7$ in WIOG strategy. We also present results for uncoordinated search (UC), where UAVs only use their own observations to update their maps as reference.

First, we consider full communication ($r = \infty$), where all UAVs can exchange information at each time step and evaluate our strategies for various values of q and A. Fig. 4.5 and Fig. 4.6 show the average number of time steps (T) required and the percentage of erroneous results (e) versus the false alarm rate q for $A = 2$ and $A = 5$ UAVs, respectively. The figures show that degrading the quality of the sensor (increasing the value of q) increases the number of time steps to locate the target in all strategies. Comparing the results in these figures, in contrast to other strategies, the errors for the *average* strategy reduces as the value of q increases. This reduction in errors comes with a cost of increase in time steps. The repetitive behavior or jumps in the plots are due to the fact that there are a fixed number of observations required to exceed the threshold for certain ranges of q (as explained in Fig. 4.2). As the value of q increases within a given range, the number of false alarms increases but the number of steps required to reach a decision remains constant. Having consecutive false alarms in a cell will end up in an erroneous result. Similarly, decreasing the value of p for fixed value of q increases the number of time steps required to terminate the search.

Second, we show the effect of increasing the number of UAVs on our merging strategies with unlimited communication. Fig. 4.7 shows the effect of increasing the number of UAVs with fixed values of $p = 0.9$ and $q = 0.2$ on the search time. Note that increasing the number of UAVs with coordinated map updates is more efficient than increasing the number of UAVs in uncoordinated search (UC). It is evident from Fig. 4.7 that sensed data merging and belief update require less time to search the region but at the cost of higher location errors. We can tune the value of v in the WIOG strategy to obtain better results depending on the values of p, q and the number of UAVs.

Third, we evaluate our proposed strategies for limited communication with fixed values of p, q and the number of UAVs. Fig. 4.8 and Fig. 4.9 show the ef-

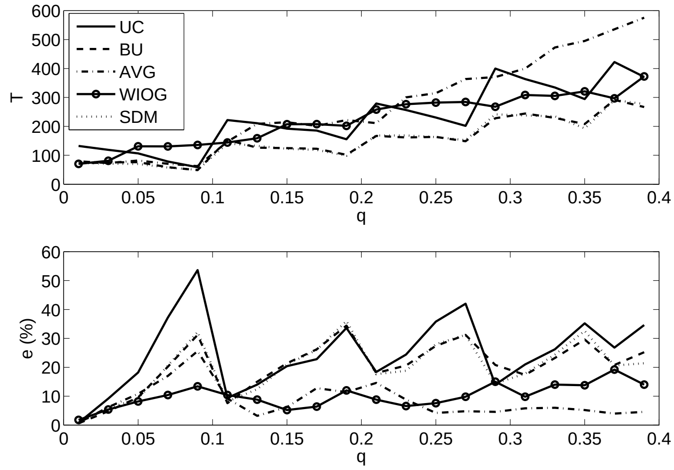


Figure 4.5: The effect of increasing the false alarm rate q on the average search time (T) and location errors (e) with $A = 2$ UAVs ($p = 0.9$, $r = \infty$).

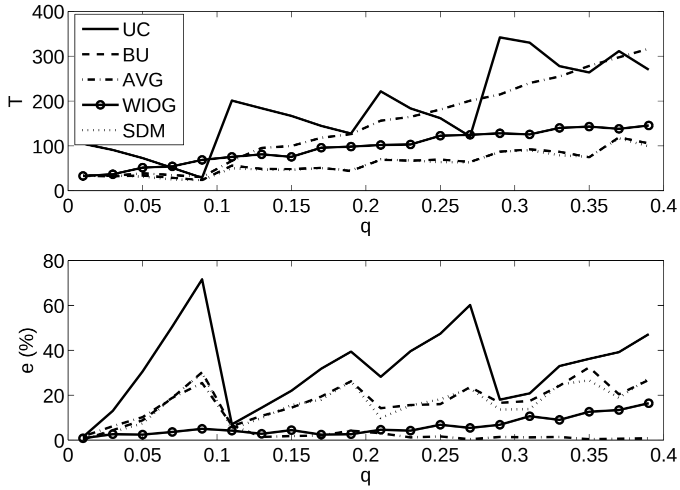


Figure 4.6: The effect of increasing the false alarm rate q on the average search time (T) and location errors (e) with $A = 5$ UAVs ($p = 0.9$, $r = \infty$).

fect of increasing the communication range on time steps required to terminate the search and percentage of erroneous results for 2 and 5 UAVs, respectively. We show the results for no communication to unlimited communication (in a 10×10 grid with $r < 14$). As the communication range increases, the performance of these strategies converge to a point that is consistent with the results of Fig. 4.7. The variations in average search time (T) and location errors (e) for uncoordinated search (UC) are caused by averaging smaller number of results.

Finally, we show the percent gain for the various merging strategies with

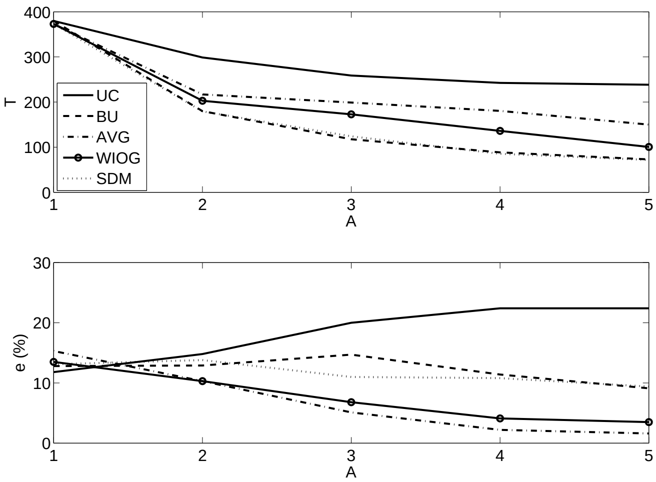


Figure 4.7: The effect of increasing the number of UAVs (A) on the average search time (T) and location errors (e) for $p = 0.9, q = 0.2$, and $r = \infty$.

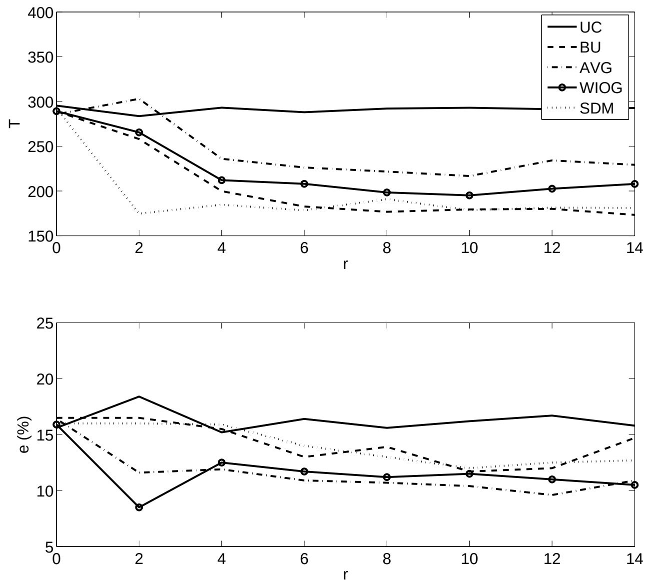


Figure 4.8: The effect of increasing the communication range (r) on the average search time (T) and the location errors (e) for $A = 2$ UAVs, $p = 0.9$, and $q = 0.2$.

respect to uncoordinated search (UC) in Table 4.2. We define the percent gain as $((\overline{T} - T)/\overline{T}) * 100$, where $\overline{T}$ represents time steps required to complete the search without coordination among UAVs. In general, the improvements rise with increasing the number of UAVs. For unlimited communication, the

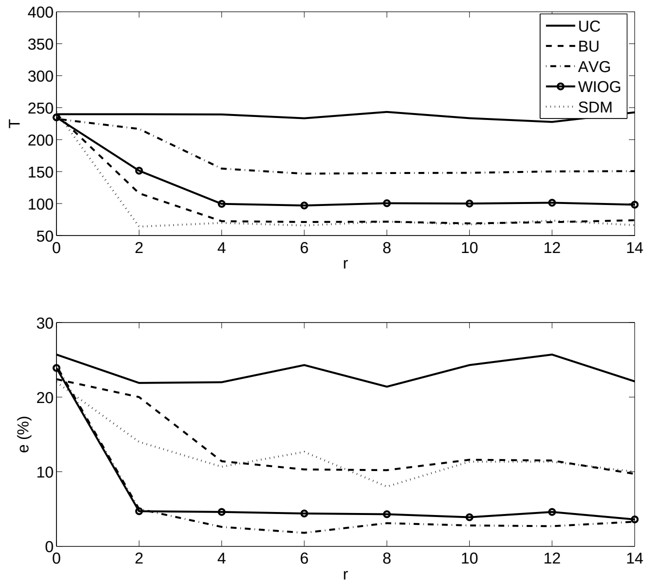


Figure 4.9: The effect of increasing the communication range (r) on the average search time (T) and the location errors (e) for $A = 5$ UAVs, $p = 0.9$, and $q = 0.2$.

minimum gain that we can achieve for a minimum number of UAVs (i.e., 2 UAVs) is 27% and the maximum gain that we can reach for maximum number of UAVs (i.e., 5 UAVs) is 70%. The improvement is also increasing with enlarging the communication range but saturates once the communication is stable. The negative gain in Table 4.2 shows that uncoordinated search performs better than coordinated search with merging of information. It happens only for short ranges of communication when UAVs take the average of thier search maps after long periods of time. Note that when there is unlimited communication, exchanging the probability maps (i.e., belief update) is sufficient to perform as good as sharing all observations. As the communication range reduces, the improvements with belief update also reduce (from 70% to 41%) since the UAVs meet each other at different times and keep different maps, while sharing observations still sustains high improvements. In this case, increasing the number of UAVs is also not sufficient. Therefore, under stringent communication, deployed merging strategy needs to be chosen carefully.

Table 4.2: Percentage gain in terms of time steps with respect to uncoordinated search ($p = 0.9, q = 0.2$)

	A	**BU**	**AVG**	**WIOG**	**SDS**
r=2	2	15.6	-7.2	14.7	36.7
	3	20.4	-2.1	16.2	48.0
	4	39.6	5.80	28.7	67.6
	5	41.6	8.00	29.7	70.9
r=6	2	30.9	22.9	32.6	37.0
	3	50.6	14.5	36.8	52.3
	4	64.0	23.9	51.3	68.6
	5	70.0	40.0	61.1	69.5
r=14	2	39.8	27.4	32.1	39.9
	3	54.5	23.1	33.1	52.0
	4	63.5	25.6	43.8	64.8
	5	69.4	37.0	57.8	69.8

4.5 Summary

In this chapter, we discussed a method to update the local information of a UAV about its environment. This method uses not only the sensor parameters and observations but also the prior information about the environment. The method relies on a discretized search map with a Bayesian update rule for the occupancy probabilities. We presented our proposed strategies for merging incomplete and outdated information about the environment. The reasons behind incomplete and outdated information are the sensing and communication limitations of the UAVs. We presented simulation results to validate the proposed merging strategies, to compare the performance of our proposed merging strategies with the uncoordinated search and to show the effects of sensing and communication parameters on the performance of the proposed merging strategies. However, there are some limitations of this work. Our proposed work considered only detection errors (false positive and missed detection) but did not consider measurement noise, which is a common sensing problem. Among a set of communication related limitations, we considered only communication range limitations. The work in this chapter is limited only to exchange and merging of simple data. However, the problems of complex data (e.g., images, videos) exchange among the UAVs need more investigation.

Chapter 5

Decision-making in Multi-UAV Cooperative Search

5.1 Overview

This chapter discusses the final task of cooperative search, i.e., joint decision-making of UAVs for planning their paths (defined in Section 3.3). This decision-making includes the selection of a cell to be visited and the order in which these cells are visited. Additionally, it is also part of the path planning to determine the time spent or the number of observations required in a given cell. The time spent in a given cell is always determined by initial observation made in that cell by a UAV. Both the information merging and decision-making components of coordination can be processed at a centralized entity or on each UAV in a distributed way enabling four processing options. The processing options are: (i) both the information merging and decision-making are centralized, (ii) information merging is centralized and decision-making is distributed, (iii) information merging is distributed and decision-making is centralized, and (iv) both the information merging and decision-making are distributed. This chapter proposes an algorithm for each processing option to analyze the effects of centralized and distributed coordination on information merging and decision-making. Each of these algorithms incorporates the steps shown in Fig. 5.1.

The rest of the chapter is organized as follows. Section 5.2 describes the computation of the path of a UAV. It also derives analytically the number of required observations to declare the absence or existence of a target in a given cell. Section 5.3 explains our proposed algorithms to execute the complete multi-UAV cooperative search and to analyze the effects of centralized and distributed coordination. In Section 5.4, we present the simulation results to validate the proposed work.

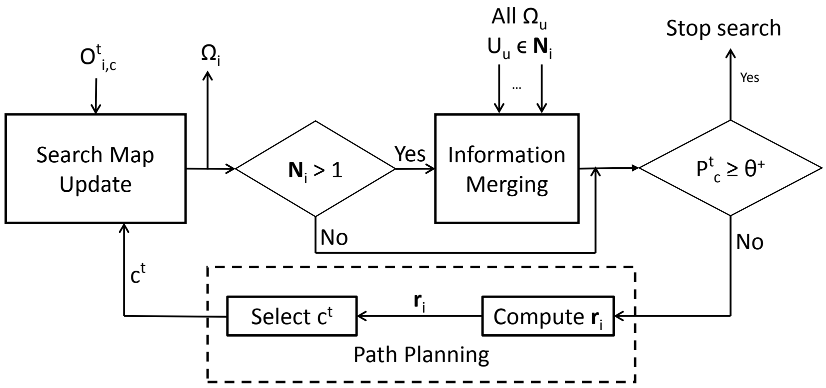


Figure 5.1: Processing diagram for cooperative search of an individual UAV U_i at time step t.

5.2 Path Planning

At each time-step t, a UAV visits a specific cell for taking an observation and exchanging local information with other team members (as shown in Fig. 5.1). The selection of cells to be visited and the order of visiting these cells (path planning) constitute the path of a UAV. The proposed approach of path planning consists of two actions: (i) to determine the path, and (ii) to determine the required number of observations in a given cell.

A closely related work can be found in [14], which uses a greedy approach in determining the path by selecting an adjacent cell each time step. It does not include the probability of miss detection and the probability of false alarm into a single expression while determining the required number of observations. Moreover, strong assumptions on the selection of sensor model, threshold, and requirement of additional prior information further limits this work [14]. Unlike [14], our proposed path-planning approach derives the expressions for calculating the required number of observations considering probability of miss detection, probability of false alarm, modeling observations as a binomial distribution, and relaxing assumptions on the selection of the sensor model.

5.2.1 UAV Path

Based on prior information, we predict the cells that are likely to contain a target. Only these predicted cells are included when determining the paths of UAVs. A path is planned in such a way that reduces the distance covered by a UAV. Instead of visiting all the cells of search region, the UAVs visit only a subset of cells using shortest paths, which reduce the time of cooperative search. We iteratively predict and determine the paths of the UAVs until all the targets are located. To predict paths, we first predict cells that need observations. We call these predicted cells as candidate cells and represent them by a set $\mathbf{S} \subseteq \mathbf{C}$. Depending on the initial observations and P_c^t, $\mathbf{S}$ is updated iteratively after each time a path is traversed (Section 5.3). The Euclidean distance between the candidate cells is considered as cost, and the set of paths for all the UAVs $\mathbf{R}$ is determined to visit cells in $\mathbf{S}$. The start and end of the paths depend on centralized (path computation at the ground station) or distributed (path computation at UAVs) coordination. Consider a graph $\mathbf{G} = (\mathbf{S}, \mathbf{E})$, where $\mathbf{E}$ is the set of edges connecting cells $a \in \mathbf{S}$ and $b \in \mathbf{S}$ ($a \neq b$) and h_{ab} is the Euclidean distance associated with edge $(a, b) \in \mathbf{E}$. Finding shortest paths to visit each cell in $\mathbf{S}$ exactly once resembles with solving the well-known Traveling Salesman Problem (TSP) and/or the Multiple Traveling Salesmen Problem (MTSP) [12] depending on the number of UAVs. The work in [12] also provides a survey on a subset of exact and heuristic solutions for the TSP and the MTSP. Any existing solution (exact or heuristic) for TSP and MTSP can be applied as an off-the-shelf component to compute the path for visiting candidate cells. Thus, computational complexity and limitations of TSP and MTSP solutions are inherited in the proposed approach.

5.2.2 Required Number of Observations

A distinct feature of the proposed path planning is the calculation of limits for the required observations for deciding on the target existence at a specified confidence level. These limits are derived considering an imperfect surveillance sensor (i.e., with miss detections and false alarms) and a Bayesian update model for the cell's occupancy probability.

The minimum and average number of sensor observations required to declare c as a target cell are represented by $m_{c,\sigma}$ and $m_{c,\mu}$, respectively. Similarly, the minimum and average number of sensor observations required to declare c as an empty cell are represented by $\overline{m}_{c,\sigma}$ and $\overline{m}_{c,\mu}$, respectively. We exploit the Bayesian update rule in Eq. 4.1 to calculate the minimum ($m_{c,\sigma}$) and average ($m_{c,\mu}$) number of observations required in a given cell c to satisfy the condition $P_c^t \geq \Theta^+$.

In order to calculate the required number of observations for a target cell, let us consider a single target cell where a sensor takes m independent observations. The sequence of consecutive binary observations has a binomial distribution with the probability of success $= p$ and frequency of successes $= \kappa$ and can be written as

$$p_{m,\kappa} = \binom{m}{\kappa} p^\kappa (1-p)^{m-\kappa}. \tag{5.1}$$

If all observations are positive, the probability of occupancy for m observations can be calculated by iteratively solving Eq. (4.1). In this case the updated occupancy probability in cell c for m positive observations is given by

$$P_c^m = \frac{p^m P_c^0}{p^m P_c^0 + q^m(1 - P_c^0)}, \tag{5.2}$$

where P_c^0 is the initial occupancy probability of cell c. Given the values of p, q and Θ^+ and the target is present, the minimum number of observations m_{min}^+ required in a cell c to satisfy the condition $P_c^m \geq \Theta^+$ can be computed, if the target is present. By transforming Eq. (5.2) with some simple algebra, the number of observations is computed by

$$m = \left\lceil \log\left(\frac{P_c^0(1 - P_c^m)}{P_c^m(1 - P_c^0)}\right) / \log\frac{q}{p} \right\rceil, \tag{5.3}$$

$$m_{c,\sigma} \geq \log\left(\frac{P_c^0(1 - \Theta^+)}{\Theta^+(1 - P_c^0)}\right) / \log\frac{q}{p}, \tag{5.4}$$

where $\lceil \cdot \rceil$ denotes the ceiling function to ensure positive integral time steps. It is clear from Eq. (5.4) that increasing the value of q or decreasing the value of p increases the minimum number of observations required to decide whether a target is in the cell.

The probability of having m consecutive negative observations in a target cell is given as $(1-p)^m$ (using Eq. 5.1). In case of positive and negative observations, the average number of observations required to satisfy the condition $P_c^m \geq \Theta^+$ can be determined as follows. Suppose $\overline{m}$ and $\underline{m}$ represent the number of negative and positive observations such that $m = \overline{m} + \underline{m}$. The binomial distribution has a mean of mp which shows that $\underline{m} = mp$ and $\overline{m} = m - mp$. The probability of occupancy after $\underline{m}$ number of positive observations $P_c^{\underline{m}}$ can be calculated using Eq. (5.2). The probability of occupancy after $\overline{m}$ number of negative observations $P_c^{\overline{m}}$ can be derived in a similar way and is given by

$$P_c^{\overline{m}} = \frac{(1-p)^{\overline{m}} P_c^0}{(1-p)^{\overline{m}} P_c^0 + (1-q)^{\overline{m}}(1 - P_c^0)}. \tag{5.5}$$

Considering $P_c^{\underline{m}}$ as a prior probability in cell c and using Eq. (5.5) to find the probability of occupancy after $\overline{m}$ consecutive negative observations, yields

$$P_c^m = \frac{(1-p)^{\overline{m}} p^{\underline{m}} P_c^0}{(1-p)^{\overline{m}} p^{\underline{m}} P_c^0 + (1-q)^{\overline{m}} q^{\underline{m}}(1 - P_c^0)}. \tag{5.6}$$

By replacing the values of $\overline{m}$ and $\underline{m}$, and simplifying algebra the average number of observations $m_{c,\mu}$ required to satisfy the condition $P_c^m \geq \Theta^+$ is calculated as

$$m = \left\lceil \frac{\log\left(\frac{P_c^0(1-P_c^m)}{P_c^m(1-P_c^0)}\right)}{(1-p)\log\frac{1-q}{1-p} + p\log\frac{q}{p}} \right\rceil, \tag{5.7}$$

$$m_{c,\mu} \geq \frac{\log\left(\frac{P_c^0(1-\Theta^+)}{\Theta^+(1-P_c^0)}\right)}{(1-p)\log\frac{1-q}{1-p} + p\log\frac{q}{p}}. \tag{5.8}$$

In analogy we can derive the probability of occupancy after m observations in an empty cell. Given the threshold Θ^- such that $P_c^m < \Theta^-$, the minimum number of observations $\overline{m}_{c,\sigma}$ and the average number of observations $\overline{m}_{c,\mu}$ required to declare a cell empty are

$$\overline{m}_{c,\sigma} = \left\lceil \log\left(\frac{P_c^0(1-P_c^m)}{P_c^m(1-P_c^0)}\right) / \log\frac{1-q}{1-p} \right\rceil, \tag{5.9}$$

and

$$\overline{m}_{c,\mu} \geq \frac{\log\left(\frac{P_c^0(1-\Theta^-)}{\Theta^-(1-P_c^0)}\right)}{(1-q)\log\frac{1-q}{1-p} + q\log\frac{q}{p}}. \tag{5.10}$$

With probability q^m, the sensor will provide false alarms in all the m observations and eventually the cell will be erroneously declared a target cell (false alarm).

5.3 Centralized and Distributed Coordination

Centralized and distributed strategies have different characteristics [86] and we want to explore the design space in the presence of resource limitations. The initial observation of an informative sensor (cp. Eq. (4.1)) greatly affects the occupancy probability which in turn determines whether a cell remains a candidate cell (if $P_c^t \geq \Theta^-$). For further observations, the search actions of the UAVs are updated to focus only on the candidate cells. Such search strategy reduces the resource usage and increases the efficiency of the cooperative search.

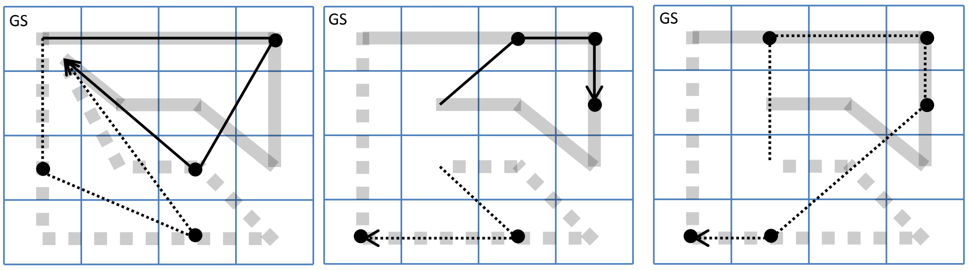
At the beginning of the search, the ground station generates a pre-computed movement plan for the *team* of UAVs. After this initial phase, representative algorithms are proposed for four possibilities with (i) centralized decision making and information merging with team movement plans (CCT), (ii) centralized decision making and distributed information merging with individual UAV movement plans (CDI), (iii) distributed decision making and information merging with individual UAV movement plans (DDI), and (iv) distributed decision making and centralized information merging with individual UAV movement plans (DCI).

5.3.1 Centralized Decision Making and Information Merging with Team Movement Plans (CCT)

CCT is a completely *centralized* algorithm (Alg. 1), where all UAVs have access to a single search map Ω on the ground station, and the ground station is responsible for the selection of the UAV paths $\mathbf{R}$ throughout the mission. Exemplified paths of two UAVs following the CCT algorithm are shown in Fig. 5.2a. The number of targets found $\overline{B}$ is initialized with 0, the set of candidate cells $\mathbf{S}$ is initialized with $\mathbf{C}$, and the number of negative observations ($\overline{m}_c$) is set to zero for each cell. The paths are traversed a fixed number of times M to have at least M number of observations per cell as shown in line 5 of Alg. 1. Based on the M initial observations, some cells (with M negative observations) are removed from the search (Alg. 2) while others become candidate cells and new paths are computed (lines 8 to 13) by the ground station to focus only on candidate cells. Thus, the number of cells to be visited for additional observations at each iteration is reduced. Each path is then traversed only once (line 17) and the process of updating $\mathbf{S}$ and $\mathbf{R}$ continues until B targets are found. An empty set $\mathbf{S}$ indicates that the search process completely missed the target, in which case the search is re-initialized (line 15).

Algorithm 1 CCT runs on the ground station.

1: **procedure** CCT$(\Omega, A, B, \overline{B}, M)$
2: $\quad \overline{B} = 0$
3: $\quad \mathbf{S} = \mathbf{C}$
4: $\quad \mathbf{M} = \{\overline{m}_c = 0 : c \in \mathbf{C}\}$
5: $\quad$ **traverse** $\mathbf{R} = MTSP(\mathbf{S}, A)$**,** M **times**
6: $\quad$ **while** $\overline{B} < B$ **do**
7: $\quad\quad \mathbf{S} = CANDIDATECELLS1(\mathbf{S}, \mathbf{M}, M)$
8: $\quad\quad$ **if** $0 <$ **size(S) then**
9: $\quad\quad\quad$ **if size(S)** $< A$ **then**
10: $\quad\quad\quad\quad \mathbf{r}_1 = TSP(\mathbf{S})$ $\qquad \triangleright\ R_u = \varnothing : u = 2, ..., A$
11: $\quad\quad\quad$ **else**
12: $\quad\quad\quad\quad \mathbf{R} = MTSP(\mathbf{S}, A)$
13: $\quad\quad\quad$ **end if**
14: $\quad\quad$ **else**
15: $\quad\quad\quad$ **initialize search**
16: $\quad\quad$ **end if**
17: $\quad\quad$ **traverse** $\mathbf{R}, 1$ **times**
18: $\quad\quad \overline{B} =$ **number of cells having** $P_c^t \geq \Theta^+$
19: $\quad$ **end while**
20: **end procedure**



(a) (b) (c)

Figure 5.2: Exemplified search paths for (a) CCT, (b) CDI and (c) DDI for 2 UAVs. Paths in gray represent initial iterations, paths in black represent the second iteration. Solid lines correspond to U_1 and dotted lines to U_2. In CCT, all UAVs start from and return to the ground station (GS) at each iteration. In CDI, the GS selects $\mathbf{R}^0$ during the first iteration and each U_i selects $\mathbf{S}_i$ and $\mathbf{r}_i$ in the second iteration (note that $\mathbf{r}_i \in \mathbf{R}_i^0$). In DDI, U_2 traverses its path $\mathbf{r}_2^0$ and selects new $\mathbf{S}_2$ and $\mathbf{r}_2$, while U_1 is still on $\mathbf{r}_1^0$ (note that $\mathbf{r}_i \notin \mathbf{r}_i^0$).

Algorithm 2 Selection of candidate cells in $\mathbf{S}$ for CCT algorithm.

procedure CANDIDATECELLS1($\mathbf{S}, \mathbf{M}, M$)
  if $\overline{m}_c \geq M$ **then** ▷ $\forall c : c \in \mathbf{S}$
    remove c **from** $\mathbf{S}$
  end if
end procedure

At each iteration, the ground station waits for all the UAVs to complete their paths and then updates both $\mathbf{S}$ and $\mathbf{R}$. While the paths are the best possible paths with the given information, the time to search is long due to the wait times. There are two reasons that stop the ground station from frequent re-planning. First, due to limited communication the information available at the ground station may be incomplete. Second, the decision made by the ground station may not be communicated to all the UAVs on time. This centralized algorithm is motivated by [109] where new paths for UAVs are computed once the previous paths are completely traversed and information along those paths is collected. Distributed algorithms where autonomy is provided to UAVs will take implicit advantage of frequent updates based on locally available information.

The selection of the value for M depends on p and the probability that a sensor can miss the target in initial $M-1$ observations but detects it in the M^{th} observation. The probability that the M^{th} observation is the first success in a target cell and $M-1$ initial observations miss the target (geometric distribution) is

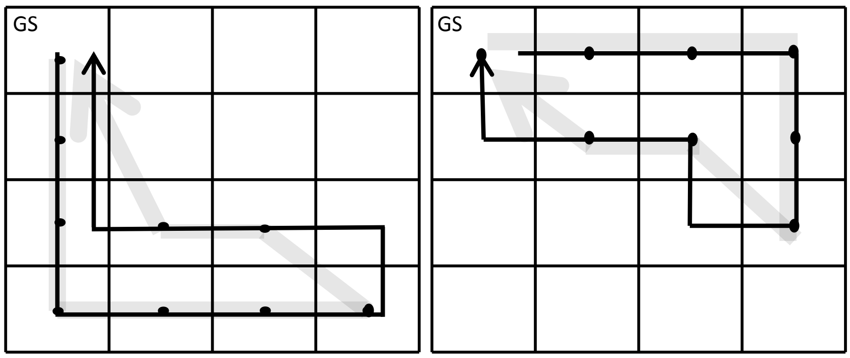
$$P_M = (1-p)^{M-1}p \tag{5.11}$$

and

$$M = \frac{log(p_M/p)}{log(1-p)} + 1 \tag{5.12}$$

For example, if $p = 0.9$ there is a less than 10% ($P_M = 0.09$) chance that a sensor will miss the target in the first observation and will detect it in the second ($M = 2$) observation. If P_M is given (by the operator), the number of initial observations necessary to predict a cell as a candidate can be selected.

If $|\mathbf{S}| > A$, the ground station updates $\mathbf{R}$ by using MTSP otherwise it assigns a single UAV (TSP path) to visit the candidate cells (lines 10 to 12 of Alg. 1). As U_i can only move to neighboring cells, it may cover multiple cells while moving from one cell of $\mathbf{r}_i$ to another. Fig. 5.3 shows an example of paths taken by the UAVs. The paths generated by the TSP and MTSP solutions are drawn with edges that cut corners of the cells and UAVs use straight-edge distances between cells in our computations. Spending one complete time step by U_i in each cell of $\mathbf{r}_i$ is necessary for time step synchronization among the UAVs.



(a) (b)

Figure 5.3: (a) Path $\mathbf{r}_1$ and (b) path $\mathbf{r}_2$ of two UAVs. The gray line shows the path computed by MTSP and the black line shows the actual path taken by an UAV. Cells with dots represent candidate cells, and GS represents the ground station.

5.3.2 Centralized Decision Making and Distributed Information Merging with Individual UAV Movement Plans (CDI)

In contrast to CCT, CDI (Alg. 3) is distributed where each UAV U_i has its own search map Ω_i, a set of candidate cells $\mathbf{S}_i$, and the ability to compute a TSP path for itself. The centralized part in this algorithm is the initial assignment of MTSP paths $\mathbf{R}^0$ (line 3) by the ground station, which restricts an UAV U_i to a distinct cluster of cells $\mathbf{r}_i^0$ for the rest of the algorithm execution. Exemplified paths of two UAVs following CDI algorithm are shown in Fig. 5.2b.

While traversing $\mathbf{r}_i$ UAV U_i determines the number of observations to take in cell c based on P_c^t and $O_{i,c}^t$ (lines 6 to 12). If $P_c^t == 0.5$ (no prior visits to c), UAV U_i takes observation $O_{i,c}^t$ and based on the value of $O_{i,c}^t$ the UAV U_i decides either to stay in c taking $m_{c,\sigma}$ observations ($O_{i,c}^t == 1$) or to move to the next cell ($O_{i,c}^t == 0$). If $P_c^t \neq 0.5$, (c has already been visited) $m_{c,\mu}$ number of observations are taken in c. As soon as an UAV moves to a cell with probability p^m, the cell will be declared as target cell in m number of observations.

Once $\mathbf{r}_i$ is completely traversed and not all B targets are found, UAV U_i updates $\mathbf{S}_i$ (line 15) and $\mathbf{r}_i$ (line 19) for itself. The selection of candidate cells in $\mathbf{S}_i$ is also distributed where a candidate cell can only be a cell from the initial path $\mathbf{r}_i^0$. The selection of a candidate cell based on M negative observations is not possible here (each cell has a different number of observations) and can be based on P_c^t. If $P_c^t < \Theta^-$, cell c is considered empty and is removed from $\mathbf{S}_i$ (Alg. 4). UAV U_i iteratively updates $\mathbf{S}_i$ and path $\mathbf{r}_i$ independently of the other UAVs and the ground station. The independent selection of $\mathbf{S}_i$ and $\mathbf{r}_i$ eliminates the waiting time of the individual UAVs at the ground station as in CCT. If there is no candidate cell UAV U_i starts again with the initial path (line 17). CDI cannot benefit fully from sharing and merging of information because the UAVs have only a small overlap in their paths and a given cell c is most likely visited only by a single UAV.

5.3.3 Distributed Decision Making and Information Merging with Individual UAV Movement Plans (DDI)

DDI differs from CDI only in the selection of candidate cells (Alg. 5). Once the initially assigned paths $\mathbf{R}^0$ are traversed by the UAVs and the targets are not found, the selection of candidate cells takes into account all cells $\mathbf{C}$ in the search region (Fig. 5.2c). The cells in the search region that have an occupancy probability $P_c^t \geq \Theta^-$ are selected for further visits.

Algorithm 3 CDI on each UAV U_i.

```
procedure CDI(r_i^0, B, Θ+, Θ−)
    B̄ = 0
    r_i = r_i^0
    S_i = {c : c ∈ r_i^0}
    while B̄ < B do                                   ▷ ∀c : c ∈ r_i
        if P_c^t == 0.5 then
            if O_c^t == 1 then
                Take m_{c,σ} observations
            end if
        else
            Take m_{c,μ} observations
        end if
        B̄ = number of cells having P_c^t ≥ Θ−
        if y_i^t == End(r_i) then
            S_i = CANDIDATECELLS2(S_i, Θ−)
            if S_i == ∅ then
                S_i = r_i^0
            end if
            r_i = TSP(S_i)
        end if
    end while
end procedure
```

Algorithm 4 Selection of candidate cells in $\mathbf{S}_i$ for CDI.

```
procedure CANDIDATECELLS2(S_i, Θ−)
    if P_c^t < Θ− then                               ▷ ∀c : c ∈ r_i
        remove c from S_i
    end if
end procedure
```

Algorithm 5 Selection of candidate cells in $\mathbf{S}_i$ for DDI.

```
procedure CANDIDATECELLS3(B−)
    if P_c^t < Θ− then                               ▷ ∀c : c ∈ C
        remove c from S_i
    end if
end procedure
```

In contrast to CCT and CDI, the distributed coordination in DDI consists of information merging as well as decision making about search actions. The selection of $\mathbf{r}_i$ considers the whole search region including the cells that are in $\mathbf{r}_u$, $i \neq u$. This in turn introduces an overlap in the paths and redundant observations in a given cell c by multiple UAVs and results in benefits in terms of search time reduction. The system's intelligence is completely embedded into each UAV, thus raising the level of autonomy of the UAVs. An UAV can now decide by itself, even if communication and availability of information is limited.

5.3.4 Distributed Decision Making and Centralized Information Merging with Individual UAV Movement Plans (DCI)

DCI and DDI are exact replicas of each other with the only difference being the location for information merging. In contrast to DDI, DCI depends on a global map Ω at the ground station for information merging. Each observation made by the UAV U_i updates Ω_i as well as Ω. There is no communication among the UAVs and the only communication possible is between UAVs and the ground station. If the communication is limited and the UAV U_i is out of communication range ($\|\mathbf{y}_i^t - GS\| > r$), then UAV U_i uses only its own map for information updates.

5.4 Simulation Results

5.4.1 Simulation Set-up

In our simulation study we measure the effect of various parameters on the search time and the search error of our proposed algorithms. We further compare our algorithms to two algorithms, which perform only information merging with predefined paths, and one algorithm, which performs distributed decision making as a reference.

In our simulations, the search region Ω is composed of $L \times W = 10 \times 10$ cells with an initial value of $P_c^t = 0.5$. The targets are randomly placed in the search region for each simulation run. The start location of all UAVs and the ground station are set to cell $c = (1, 1)$. Note that in our setup, the communication range $r \geq 14$ corresponds to unlimited communication.

All experiments are based on $N = 1000$ simulation runs. The average search time (T), the average search errors (e) and the false discovery rate (FDR) [130] are used as performance metrics. T is defined as the time of completion of the search algorithm, i.e., when at least B cells with $P_c^t \geq \Theta^+$ have been detected.

A completed search is erroneous, if at least one target has not been properly detected ($\exists c \mid P_c^t \geq \Theta^+ \wedge X_c = 0$). Thus, the error rate is given as $e = \frac{e_s}{N}$ where e_s represents the number of erroneous searches. For $B > 1$, FDR is defined as $FDR = \frac{1}{N}\sum(\overline{e}/B)$ where $\overline{e}$ represents the number of faulty target detections in a single search run ($0 \leq \overline{e} \leq B$).

All simulations are performed for specific parameters of $p = 0.9$, $q = 0.2$ to represent an informative sensor with probability of detection near one and probability of false alarms near zero. The value of $\Theta^+ = 0.99$ is used to show that 99% confidence in the search result is required to stop the search. Degrading the quality of the sensor (p and q) increases the number of time steps to locate the target (cf. [73]). In all simulations, the nearest neighbor heuristic[1] is used for solving TSP and a heuristic based on genetic algorithm[2] is used for solving MTSP and thus algorithms proposed in this research inherit all the limitations of TSP, MTSP and applied heuristics.

The proposed algorithms are compared with traditional sweep search or lawnmower-type search [22] where UAVs move straight from one boundary of the search region to the other. To avoid multiple UAVs following the same path concurrently, some initial randomness has been introduced in the sweep search. Each UAV starts from the ground station in a random direction (either along the rows or along the columns of the search region) and follows that direction for a random number of cells (less than L or W). Then the UAVs continue with the traditional sweep pattern in the opposite direction. Uncoordinated sweep search (US) does not use information sharing and coordinated sweep search (CS) shares and merges information using the belief update strategy. The proposed algorithms are also compared with distributed cooperative search (DCS) [132, 92], a recent sophisticated algorithm for minimizing the time of multi-UAV cooperative search.

5.4.2 Threshold Selection

Simulations are performed for three different threshold values $M = 1$, $M = 2$ and $M = 3$ in CCT and two different threshold values $\Theta^- = 0.5$ and $\Theta^- = 0.1111$ in CDI, DDI and DCI. Setting $\Theta^- = 0.5$ means that the UAVs include a cell c in $\mathbf{S}$ if $P_c^t \geq 0.5$. Otherwise, c is removed from the $\mathbf{S}$ which means that a single observation in cell c is sufficient to include or remove it for further search. Similarly, setting $\Theta^- = 0.1111$ (a cell is a candidate if $P_c^t \geq 0.1111$) in CDI, DDI and DCI, corresponds to two negative observations in a cell to remove it from $\mathbf{S}$. Fig. 5.4 shows the effect of different threshold values (M

[1]http://www.mathworks.co.uk/matlabcentral/fileexchange/35178-tspsearch

[2]http://www.mathworks.co.uk/matlabcentral/fileexchange/19049-multiple-traveling-salesmen-problem-genetic-algorithm

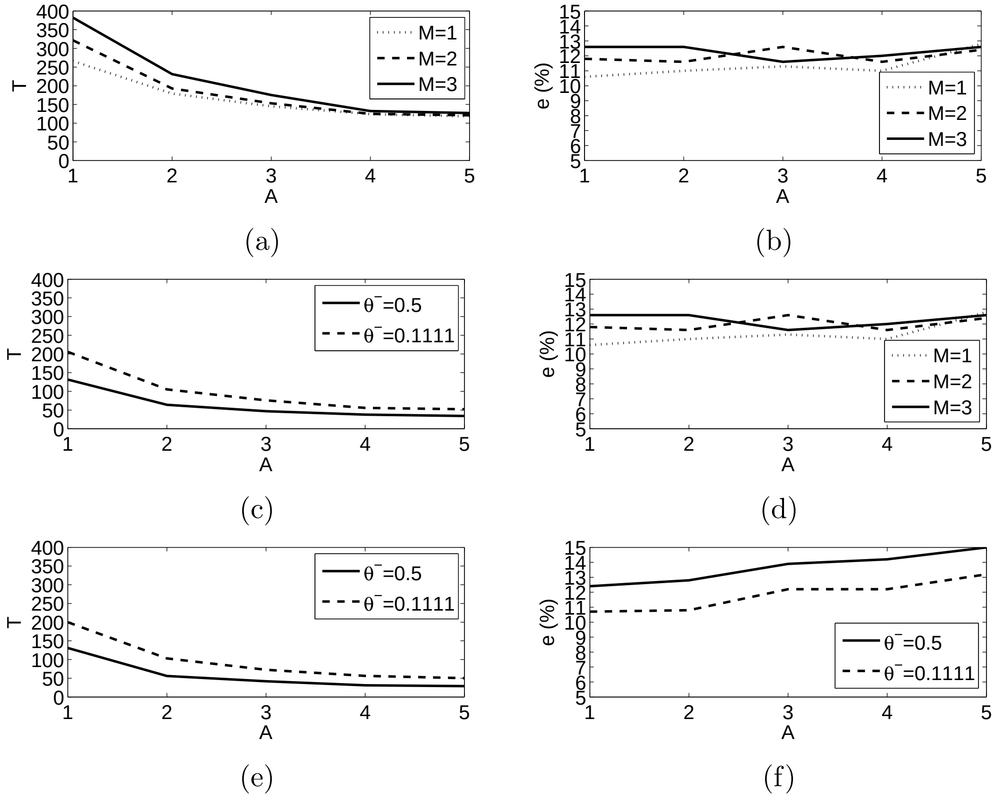


Figure 5.4: The effect of increasing the number of UAVs (A) and increasing the threshold values (M and Θ^-) on the search time (a) CCT, (c) CDI and (e) DDI and the effect of increasing the number of UAVs and increasing the threshold values on the search errors (b) CCT, (d) CDI and (f) DDI ($p = 0.9, q = 0.2, \Theta^+ = 0.99, B = 1, r = 14$).

and Θ^-) on CCT, CDI and DDI. DCI and DDI generate the same results when communication range is unlimited (see Section 5.4.3) and that is why Fig. 5.4 does not show results for DCI. Since increasing M or decreasing Θ^- increases the number of independent observations the path lengths of the UAVs have to increase as well which in turn increases the search time. Fig. 5.4 shows the trade-off between search time and search errors as the number of UAVs increase.

5.4.3 The Effect of Communication Range

Varying the communication range for CCT does not affect the search time, as UAVs do not share information and make decisions while they are traversing the paths. Similarly, the effects of communication range on the performance of CDI are small, as overlapping paths for information merging are unlikely and UAVs cannot fully benefit from information merging. However, the communication range influences the search time of DDI and DCI (Fig. 5.5).

If no communication among the UAVs is possible, CDI performs better than DDI and DCI. The reason is that the length of path $\mathbf{r}_i$ for the UAV U_i does not exceed the length of $\mathbf{r}_i^0$ in subsequent iterations of CDI. While it is more likely that the length of $\mathbf{r}_i$ in subsequent iterations of DDI exceeds the length of $\mathbf{r}_i^0$ to consider the whole search region Ω. This increase in the length of paths does not facilitate other UAVs in searching the whole region Ω and, thus, increases the overall search time. If communication is possible, the overlap in paths in subsequent iterations of DDI increases the frequency of information exchange and merging among the UAVs. This increase in information merging reduces the number of candidate cells and the length of paths leads to a reduction in the overall search time of DDI (Fig. 5.5). This improvement increases as the communication range enlarges but saturates as the UAV network becomes fully connected.

A UAV running DCI algorithm (no direct communication among the UAVs) is isolated if it is not in communication range with the ground station, even if there are other UAVs in its neighborhood. This makes the search using DCI algorithm slower than the search using DDI algorithm, as shown in Fig. 5.5. As compared to CDI, the improvement in DDI is greater than 13% for 100 cells and full communication. This improvement enhances with increasing number of cells (Section 5.4.5). Moreover, it is clear from Fig. 5.5 that DDI and DCI generate exactly the same results when either $r == 0$ or r covers the whole search region. In such cases, keeping information merging either centralized or distributed makes no difference. Therefore, results for DCI algorithm are not shown in the remaining sub-sections where full communication is assumed.

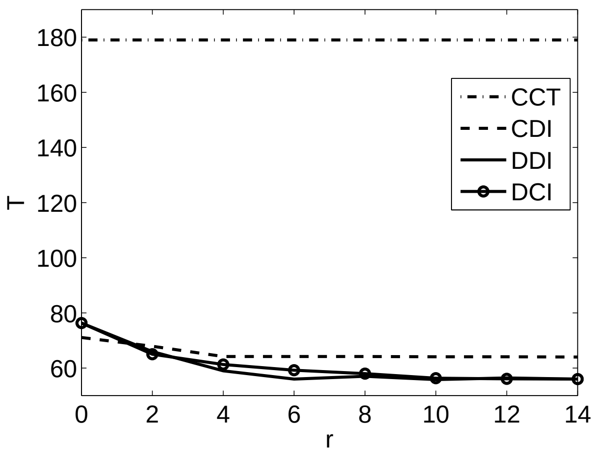


Figure 5.5: The effect of increasing communication range (r) on average time steps (T) for $A = 2,\ B = 1,\ p = 0.9,\ q = 0.2,\ \Theta^+ = 0.99$ and $\Theta^- = 0.5$.

5.4.4 Comparison of Algorithms

The goal is to illustrate the impact of information merging and decision making on cooperative search. As shown in Fig. 5.6, CCT, CDI and DDI significantly reduce the search time as compared to US, CS, and DCS. An interesting observation is that CS performs better than CCT for larger numbers of UAVs, which is caused by the longer waiting times for UAVs at each iteration in CCT. As the number of UAVs increases DCS outperforms US, CS and CCT due to its ability to frequently update the search action on-line and independent of other UAVs or the ground station. Due to the coordinated decision-making, information merging and more intelligent search action selection CDI and DDI clearly outperform the other algorithms.

The proposed algorithms reduce the number of candidate cells in each iteration and increase the number of observations only in candidate cells. The increase of observations only in candidate cells makes the distribution of observations in the search region very skewed towards a subset of cells in the search region. This skewed distribution of observations increases the chances of finding the target without covering the whole search region multiple times and causes a reduction in search time. The distributions of observations in terms of standard deviation of the observations per cell for CCT, CDI, DDI and DCS algorithms are shown in Fig. 5.7. The standard deviation of number of observations per cell σ for no target in the search region and 140 time steps is computed regardless of which UAV visited the cell. It is clear from Fig. 5.7

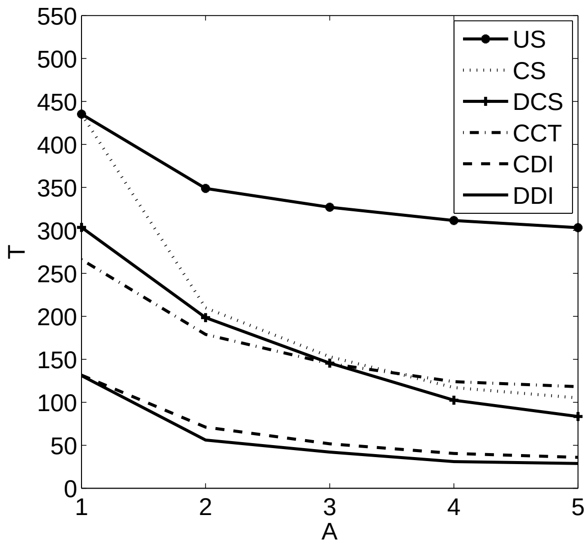


Figure 5.6: Comparison of algorithms for $B = 1$, $p = 0.9$, $q = 0.2$, $\Theta^+ = 0.99$, $M = 1, \Theta^- = 0.5$, and $r = 14$.

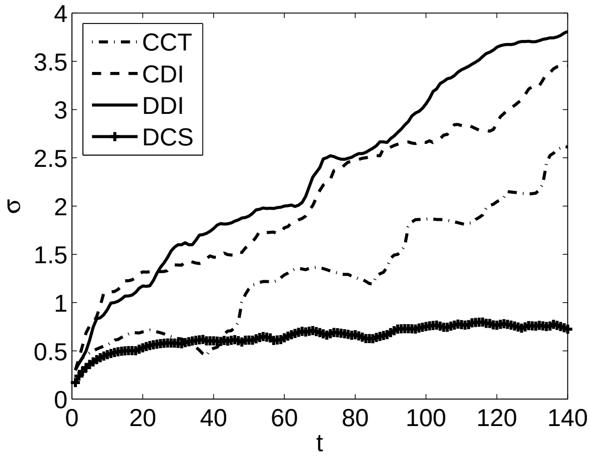


Figure 5.7: Standard deviation of the number of observations per cell for $E = 140$ time steps and a single simulation run for $A = 3,\ B = 0,\ p = 0.9,\ q = 0.2,\ \Theta^{+} = 0.99,\ M = 1,\ \Theta^{-} = 0.5$ and $r = 14$.

that DDI introduces abrupt variation in number of observation per cell as time increases, which causes significant reduction in search time.

5.4.5 The Effect of Search Region Size

Increasing the size of search region, on average, increases the search time of all algorithms. Fig. 5.8 shows the effects of increasing the size of the search region on performance of proposed algorithms. Simulations in this section are performed for nine different search region sizes, starting from $\mathbf{C} = 4 \times 4$ cells to $\mathbf{C} = 20 \times 20$ cells. Both the number of rows and number of columns were incremented by two to get different search region sizes. The increase in the size of the search region results in an increase in the variation in lengths of paths and number of candidate cells at different iterations of all the proposed algorithms. This variation increases the waiting time in CCT and thus significantly reduces the speed of cooperative search. It is also clear from Fig. 5.8 that the difference in the results generated by CDI and DDI enlarges with the increase in the size of search region.

5.4.6 Multiple Targets

Fig. 5.9 shows the impact of the number of UAVs for three targets ($B = 3$) on T, e and FDR. Increasing the number of UAVs for a fixed number of targets gradually reduces T but has no considerable effect on e and FDR. On the contrary, increasing the number of targets increases T and e but reduces FDR. The reason is that a larger number of UAVs increases the observations per

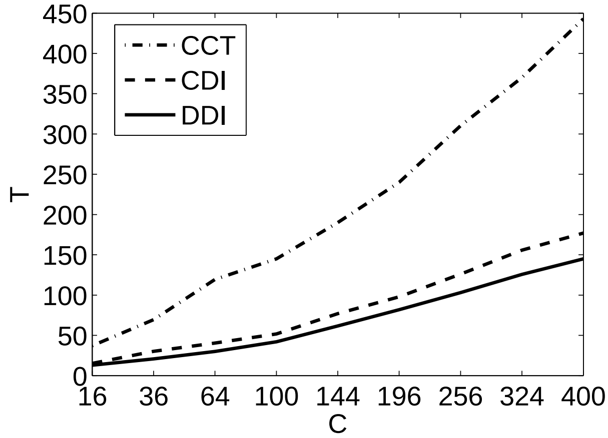


Figure 5.8: The effect of increasing the number of cells in search region for $A = 3,\ B = 1, p = 0.9,\ q = 0.2,\ \Theta^{+} = 0.99,\ M = 1,\ \Theta^{-} = 0.5$, and $r = 14$.

cell which helps to reduce the number of false positives and, thus, FDR. The search error increases since the probability of missing one out of B targets also increases.

CDI does not work for multiple targets, if the UAVs are unable to communicate ($r = 0$). The reason is that each UAV tries to find B targets and not all of them might be present in its assigned initial path (cluster of cells). In that case, each UAV may never terminate the search. Similarly, the communication range does not affect CCT as explained in Section 5.4.3. Therefore, the effect of varying communication range is shown only for DDI (Fig. 5.10). It is evident from Fig. 5.10, that enlarging the communication range reduces the search time and FDR while increasing the search errors. FDR and e converge to a single point in the case of unlimited communication range.

5.5 Summary

In this chapter, we have presented the path planning of UAVs by proposing a method that considers not only the order of vising different regions of the environment but also the number of observation required to make a decision about a given region. We have represented information merging and path planning as two important coordination dimensions for multi-UAV search. We have investigated four search methods—which differ in whether these two coordination dimensions are centralized or distributed—with limited sensing and communication capabilities. We analytically derived the number of independent observations in order to decide on target absence or existence at a given confidence level. In our simulation results, we showed that distributed coordi-

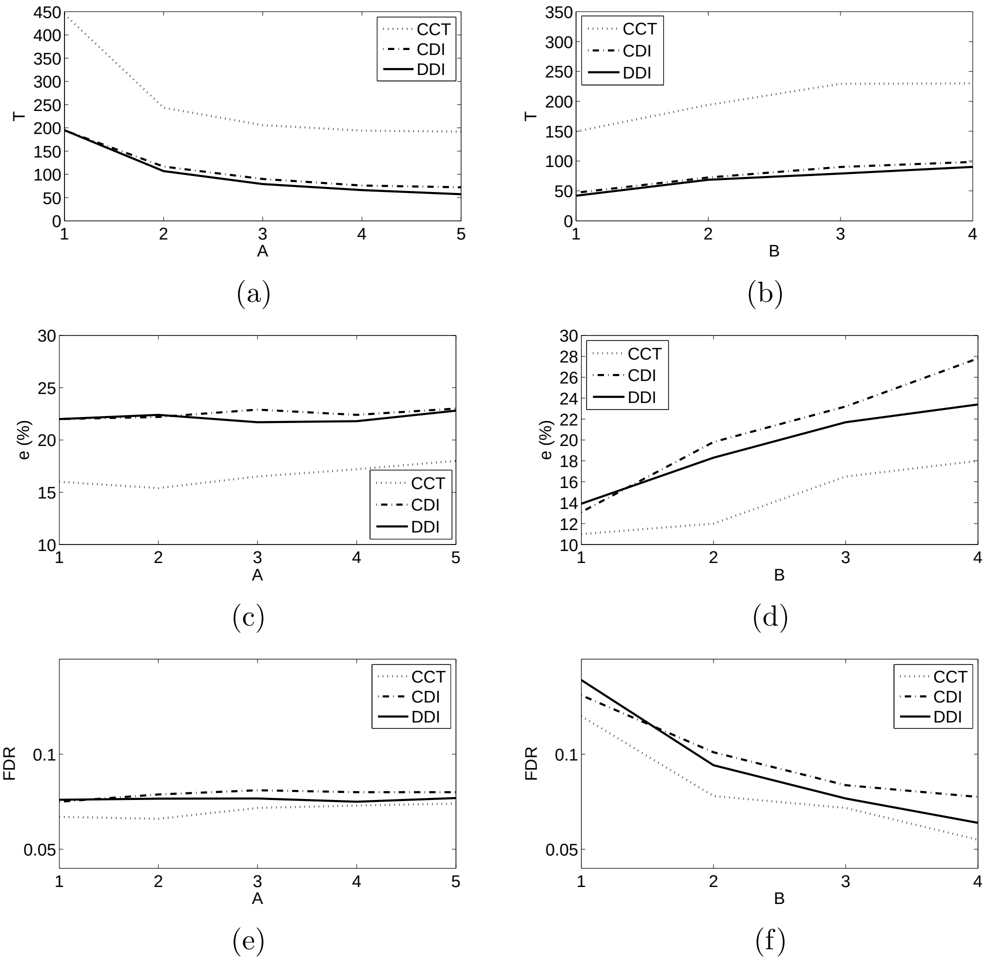
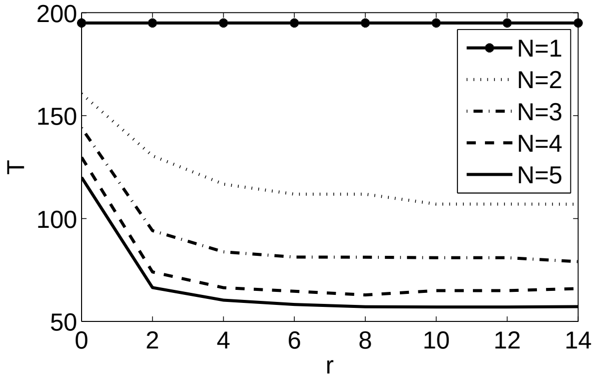


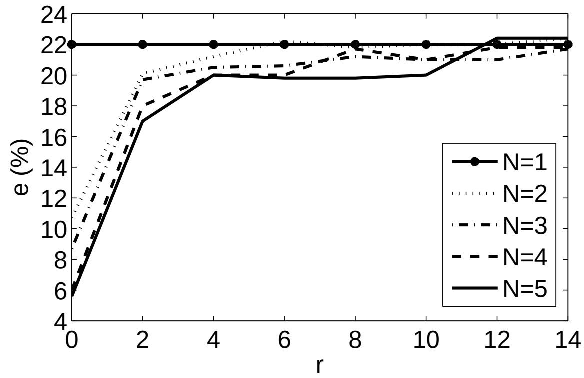
Figure 5.9: The effect of increasing the number of UAVs (with $B = 3$) on (a) T, (c) e, and (e) FDR, and increasing the number of targets (with $A = 3$) on (b) T, (d) e, and (f) FDR ($p = 0.9, q = 0.2, \Theta^+ = 0.99, M = 1, \Theta^- = 0.5, r = 14$).

nation significantly reduces the search time as compared to uncoordinated or centralized coordinated approaches.

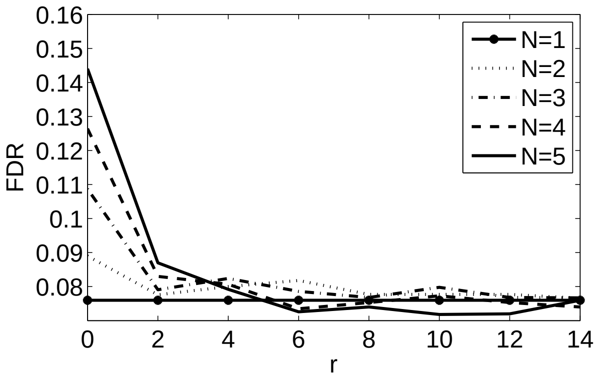
A limitation of the proposed algorithms is their sensitivity to the values of the sensor parameters. Determining these parameters is a challenging task because of their dependency on various factors, e.g., type of the sensor (hardware and software), environmental conditions and altitude of the sensor. Deficiencies in the sensor parameters will naturally have a negative effect on the performance of the proposed algorithms. The actual duration of a single time step depends on the overall processing time. The computation of the TSP/MTSP solution can become the dominant component, especially in case of large search areas. In case of heterogeneous sensors, more sophisticated information merging strategies are required. To avoid collisions, the algorithms keep the MAVs



(a)



(b)



(c)

Figure 5.10: The effect of increasing the communication range ($B = 3, p = 0.9, q = 0.2, \Theta^+ = 0.99$ and $\Theta^- = 0.5$) for DDI.

at slightly different altitudes. This may cause considerable variations in accuracies and the field of views of the sensors. Moreover, the target location accuracy is modelled at the cell level which in turn depends on the sensor's field of view and, thus, the proposed algorithms cannot determine the position of a target within the field of view or the cell.

Chapter 6

Multi-UAV Cooperative Observation

6.1 Overview

This chapter discusses the second application of aerial surveillance using a team of small-scale UAVs, i.e., cooperative observations of moving targets (Section 2.3 and Subsection 3.3.2). The chapter presents a centralized algorithm for multi-scale observation of multiple moving targets using a team of UAVs. We extend the conventional fixed-altitude or fixed-FOV-size problem (defined in Subsection 3.3.2) to multiscale observations using a multi-UAV system with noisy sensors. The proposed algorithm is appropriate when UAVs can observe targets at different elevations with the objective of jointly maximizing duration and resolution of observation for each target. The UAVs share the workload using a greedy assignment of locations and targets to UAVs. To model observations at large spatial scales (low resolution) versus observations at a small spatial scales (high resolution), we use a quad-tree to help in defining the tradeoff between the visibility and the quality of observations of multiple moving targets. We consider cases where there is uncertainty in the target observations (i.e., measurement noise), the number of targets is larger than that of the UAVs and the combined FOVs of the sensors cannot cover the whole search region.

The rest of the chapter is organized as follows. Section 6.2 discusses the proposed approach for multi-scale observation of moving targets. In Section 6.3, we present the simulation results to validate the proposed work.

6.2 Quad-tree based Space Discretization

We discretize and model the 3D space for the movement of UAVs as a quad-tree $\mathbf{V}$ [32] with K nodes (Fig. 6.1a). Let d denote the depth of the tree where the root node is at $d = 1$ and the leaf nodes are at maximum depth of $d = \varepsilon$. In the proposed framework, the topology of the tree is fixed (nodes cannot be added or deleted) and complete (all its leaves are at the same depth). Except the root and the leaf nodes, each node k has five adjacent nodes, i.e., k_0 (parent node) and four children nodes k_1 (north west), k_2 (north east), k_3 (south east), k_4 (south west). The levels of the quad-tree are related to the minimum allowable altitude as

$$z = 2^{\varepsilon - d} z_0. \tag{6.1}$$

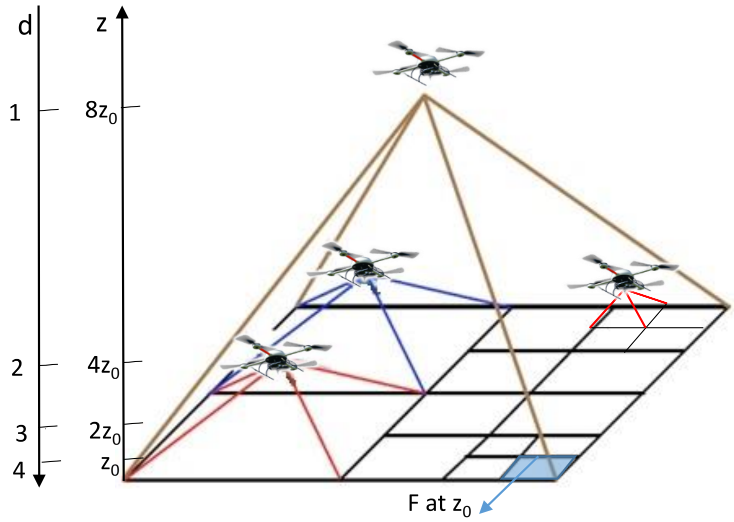
By considering the value of z in Eq. 3.12 and dividing by z_0 yields the normalized value $s_i \leq 1$ for the resolution of observation made by UAV U_i at depth d_i

$$s_i = \frac{1}{2^{\varepsilon - d_i}}. \tag{6.2}$$

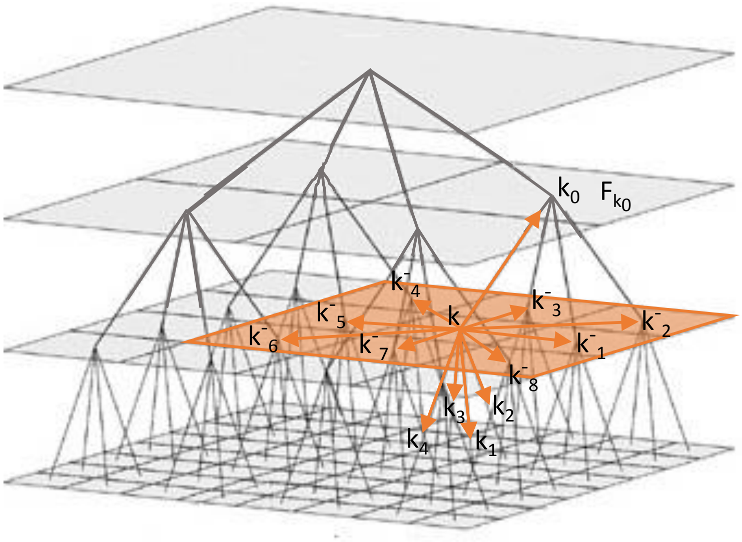
Each node represents an allowable location for the movement of UAVs, such that $\mathbf{y}_i^t = k$ for $k = \{1, 2, ..., K\}$. Every node is associated with a FOV. Any UAV U_i that hovers at node k will always have a specific FOV, denoted as $F_i = F_k \subset \Omega$. If k is an internal node and $k_1, ..., k_4$ are its children, then the various F_{k_i} are obtained by splitting the F_k into four equally sized squares. Therefore $F_k = \cup_{i=1,2,3,4} F_{k_i}$ and $F_{k_i} \cap F_{k_j} = \emptyset$ where k_i and k_j are siblings. It is obvious that a UAV at node k with FOV F_k is already observing $F_{k_1}, F_{k_2}, F_{k_3}, F_{k_4}$ with resolution s_k. A UAV can only take observation when located at a node of $\mathbf{V}$ as shown in Fig. 6.1a.

The root of the quad-tree $\mathbf{V}$ is centered at Ω, such that $F_{\mathbf{V}} = \Omega$ and leaf nodes are at $z = z_0$. This centralized quad-tree is used as a coordination mechanism among UAVs. The key purpose of the quad-tree is to reduce the movement options from 27 in an unconstrained neighborhood cube to only 14 (including the current position). These fourteen nodes include the current node k, 8 nearest nodes on the same level of the quad-tree $\hat{k}_1, ...\hat{k}_8$, the parent node k_0, and 4 children nodes $k_1, ..., k_4$. The exceptions are the root node (5 movement options), the leaf node (10 movement options), and nodes on $d = 2$ (9 movement options). The fourteen movement options for a UAV hovering at node k are shown in Fig. 6.1b. We do not need 27 movement locations, because increasing the altitude reduces the number of movement locations as $\bigcup\limits_{k \in \mathbf{K}_d} F_k = \Omega$, where $\mathbf{K}_d$ is the set of nodes at any given depth d.

Fig. 6.2 shows the overall process in a single time step. The UAVs observe the targets' actual states $\mathbf{X}^t$ at time step t and generate the measurements $\mathbf{Z}^t$.



(a)



(b)

Figure 6.1: Model for the UAV movement: (a) Relationship of search region Ω, FOV (F), altitude (z), lowest altitude (z_0), and depth (d) of the quad-tree. (b) Fourteen-neighborhood (orange arrows and square around the current node k).

Information about the target locations $\mathbf{Z}^t$ and current states of all the UAVs $\mathbf{Y}^t$ at time step t are used to update the centralized quad-tree $\mathbf{V}$. This updated quad-tree and the current states of all the UAVs are then used to move each UAV to one of 14 neighboring locations.

The objective of the UAV movement decision is to appropriately identify the nodes of the $\mathbf{V}$ that maximize the number of targets under observation and their resolution. These nodes are identified and assigned to the UAVs as

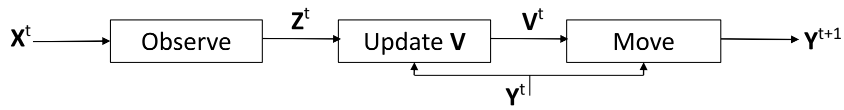


Figure 6.2: Block diagram of the overall process in a single time step.

waypoints at each time step.

The ground station maintains the centralized quad-tree $\mathbf{V}$. In addition to $\mathbf{y}_i$, each node of the quad-tree $\mathbf{V}$ maintains the following value

$$v_k = \frac{\alpha b_k + (1-\alpha) b_k s_k}{1 + A_k}, \tag{6.3}$$

where b_k is the number of targets visible under node k, A_k is the number of UAVs hovering at node k and s_k is the resolution (Eq. 6.2) of observation taken by a UAV hovering at node k. An increase in the depth of node k or the number of targets visible from node k increases the value of v_k. The term A_k in the denominator introduces the spread among the UAVs. In our proposed algorithm, the ground station identifies and assigns nodes to the UAVs that maximizes the team's dispersion pattern, number of targets under observation, and resolution of observation. The movement decision takes into account two sub-goals, namely maximization of the number of targets under observation and resolution maximization. While the UAVs can be trapped in a local maximum, the effects of this local maximum are likely to be temporary as targets move.

The value of v_k cannot be determined accurately when F_k contains one or more targets and F_k is not observed by any UAVs. The states of such unobserved targets are estimated using Eq. 3.2 and the last known locations of these unobserved targets. To include uncertainty in these estimated target states we use the process noise covariance matrix $\overline{\mathbf{Q}}$ which is larger in magnitude than $\mathbf{Q}$. We assume a random location in the unobserved part of Ω as an estimate of the target for which no information is available.

The centralized controller determines the new positions for all the UAVs, as presented in Algorithm 6. At each time step, the new positions are determined sequentially starting from U_1 to U_A (line 15 in Algorithm 6). To find the new position for a UAV U_i two steps are required. First, states for the unobserved targets, if any, are estimated (line 16 to line 20) and measurements $\mathbf{z}_{ij}^t$ (Eq. 3.7) about the target locations are made. These measurements are used to calculate the value of b_k. Second, the value v_k for all the nodes in $\mathbf{V}$ is updated (line 21 to 23). The values of b_k and s_k required to update v_k are determined from UAV observations (Eq. 3.7) and estimated states of unobserved targets (line 16 to line 20). The value of A_k is updated in each iteration of the outer loop (line 15). Third, the new position $\mathbf{y}_i^{t+1}$ for UAV U_i is determined (line 27). This new position is one of fourteen adjacent positions (including the current node) that has the maximum value of v_k. If more than one node has the same value of v_k, priority is given to the node at higher depth level (to reduce the altitude of the UAV). If more than one node on one level of the quad-tree has same value of v_k, priority is given to the node that comes first in the anti-clockwise direction. These priorities are taken into account in line 25 and line 26 of Algorithm 6 by sorting the 14 nodes around node k into a temporary array *temp*. The node in *temp* with a maximum value of v_k is determined as the new position for UAV U_i (line 27). After loop termination (line 29), the ground station has new positions $\mathbf{Y}^{t+1}$ for all the UAVs. These new positions/states are sent to UAVs for taking further observations.

6.3 Simulation Results

We perform simulations for a region Ω of $L \times W = 4096 \times 4096$ m^2 with area of a cell as 1 m^2. We consider mission duration of $E = 1000$ time steps, $\mathbf{Q} = 0.1 \times I_{4\times4}$, $\overline{\mathbf{Q}} = 1 \times I_{4\times4}$, target velocity of $\dot{\mathbf{x}} = 5$ meters per time step, and a quad-tree $\mathbf{V}$ of five levels ($\varepsilon = 5$). Knowing the dimensions of the Ω, and ε the $\mathbf{V}$ is initialized with $v_k = 0, k = 1, 2, ..., K$. We initialize the location of each UAV from the root of the quad-tree, unless otherwise stated. Fig. 6.3 shows the search region with simulated paths (using Eq. 3.2) of $B = 8$ targets, initialized at random locations and random directions.

We show the observations and their associated depths of the quad-tree (d) for eight targets ($B = 8$) and a team of three UAVs ($A = 3$) in Fig. 6.4a and Fig. 6.4b. The results are the average of $N = 100$ runs of simulations for different target tracks. Higher values of α affect the movement of UAVs to increase the number of targets under observation but do not care for quality of observation. Reducing the value of α affects the movement of UAVs by forcing them towards leaf nodes $d = 5$. Fig. 6.4b shows that, on average, all the targets are observed at high resolution.

The time evolution of paths for $A = 3$ UAVs observing $B = 8$ targets is shown in Fig. 6.5. The UAVs U_1 and U_3 start observing targets with highest resolution during the first $E = 100$ time steps. The UAV U_2 cannot reduce the altitude because it would make more targets unobserved. Throughout the mission, the UAVs vary their altitudes to avoid empty FOV and large number of targets being unobserved.

The effect of changing the number of UAVs and targets on the performance measure (Eq. 3.15) is shown in Fig. 6.6. The figure shows how the approach scales with the number of targets and UAVs. The increase in the number of UAVs (A) for a given value of B and a given value of α always increases the collective time and resolution of observation (g). Increasing the number of

Algorithm 6 UAVs movement.

A: number of UAVs
B: number of targets
$\mathbf{x}_j^t$: state of target T_j at time step t
$\mathbf{y}_i^t$: state of UAV U_i at time step t
Φ: transition matrix (Eq. 3.2)
γ: $\mathcal{N}(0, \overline{\mathbf{Q}})$ process noise (Eq. 3.2)
A_k: number of UAVs at node k
b_k: number of targets visible from node k
s_k: resolution associated with node k (Eq. 6.2)
temp: temporary array to store 14 nodes priority-wise
$\mathbf{w}$: temporary variable to store the current state of U_i
Initialize quad-tree $\mathbf{V}$ by setting $v_k = 0$ for $k = 1, ..., K$
Initialize the UAV states $\mathbf{Y}^0$
procedure MOVEUAVS($\mathbf{V}, \mathbf{Y}^t$)
    for $i = 1 : A$ **do**
        for $j = 1 : B$ **do**
            if $O_{ij}^t == 0$ **then**
                $\mathbf{x}_j^t = \Phi \mathbf{x}_j^{t-1} + \gamma$
            end if
        end for
        for $k = 1 : K$ **do**
            $v_k = \frac{\alpha b_k + (1-\alpha) b_k s_k}{1 + A_k}$
        end for
        $\mathbf{w} \leftarrow \mathbf{y}_i^t$
        temp $\leftarrow [\mathbf{w}_1,\ \mathbf{w}_2,\ \mathbf{w}_3,\ \mathbf{w}_4,\ \mathbf{w},\ \mathbf{w}_1^{-}\ \mathbf{w}_2^{-}\ \mathbf{w}_3^{-}\ \mathbf{w}_4^{-}\ \mathbf{w}_5^{-}$
            $\mathbf{w}_6^{-}\ \mathbf{w}_7^{-}\ \mathbf{w}_8^{-},\ \mathbf{w}_0]$
        $\mathbf{y}_i^{t+1} \leftarrow$Node in *temp* with maximum value of v
    end for
    Output $\mathbf{Y}^{t+1}$
end procedure

targets (B) results in the following four different trends, which are caused by different values of A and α: (i) smaller values of both α and A result in slower increase of g; (ii) the combination of smaller value of α and larger value of A decreases g; (iii) the combination of larger value of α and smaller value of A increases g; and (iv) larger values of both α and A decrease g. It is clear from Fig. 6.6 that the value of g for different values of A converges as we decrease the value of the ratio A/B. This convergence is faster for higher values of α. A decrease of α compels the UAVs to reduce their altitudes resulting in loosing of targets from observation, which reduces not only the value of g but also the

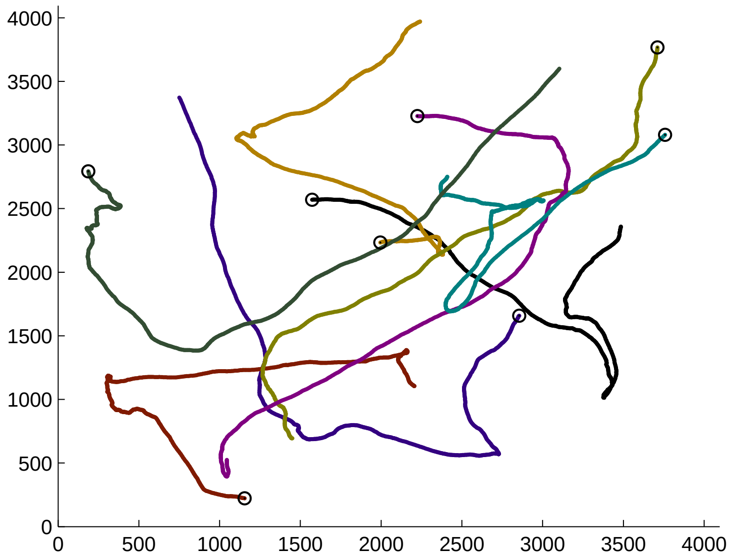


Figure 6.3: Simulated paths of $B = 8$ targets for $E = 1000$ time steps. Black circles show the starting points of the paths.

convergence to same value of performance measure. Therefore, a decrease in α for given values of A and B decreases the value of g.

We show the effect of the quad-tree size on the performance measure (Eq. 3.15) in Fig. 6.7. We perform the simulation for the quad-tree ranging in size from one level (only one node, $\varepsilon = 1$) to seven levels ($\varepsilon = 7$, 4096 leaf nodes and 21845 total nodes). It is clear from Fig. 6.7 that the size of the quad-tree does not affect the performance for $\alpha = 1$. Because the UAVs will always increase altitude to observe the whole region. However, for smaller values of α, increase in size of the quad-tree abruptly increases the performance. We find that allowing more locations for the UAV movement can maximize both the collective time and collective resolution of observation.

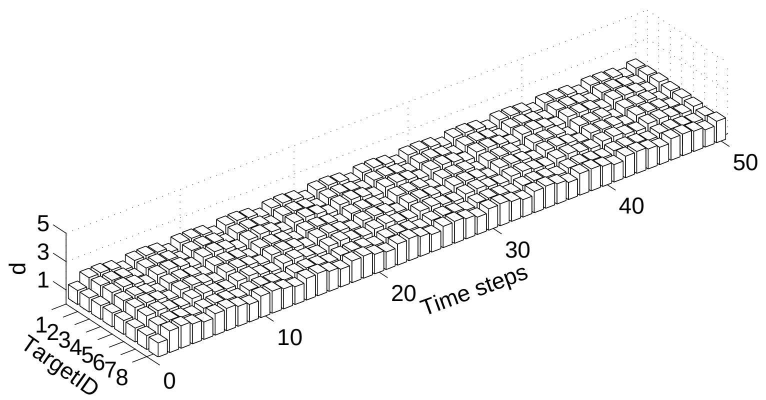
The UAV location initialization also affects the performance of our proposed approach. The effect of UAV location initialization is shown in Fig. 6.8. We perform one simulation by initializing all the UAVs at the root node of $\mathbf{V}$ and one simulation by initializing all UAVs at random leaf nodes. We plot

$$g_t = \frac{1}{B} \sum_{j=1}^{B} \left(\alpha \bigvee_{i=1}^{A} O_{ij}^t + (1-\alpha) \bigsqcup_{i=1}^{A} r_{ij}^t \right) \tag{6.4}$$

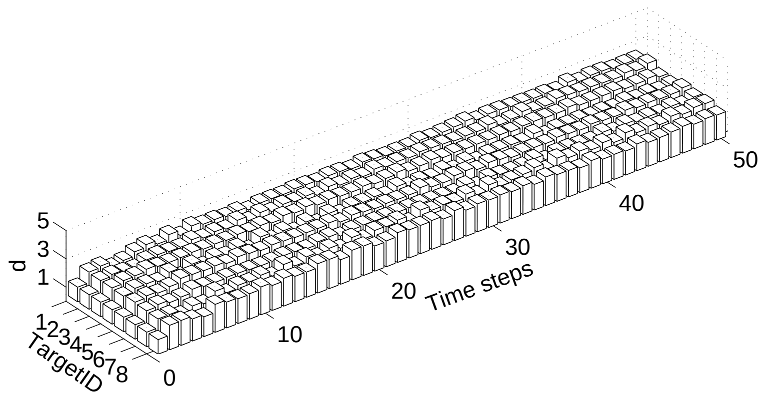
for each time step, t, to show the instantaneous performance of our proposed approach. Fig. 6.8 shows that initialization at root node is better for immediate performance (notice first 50 time steps in Fig. 6.8). As time passes, performance due to both types of initializations converges to the same value.

The effect of observation noise (Eq. 3.7) is shown in terms of covariance matrix, which is $\mathbf{O} = u \times I_{4\times4}$. We increase the observation noise by increasing the value of u. Fig. 6.9a and Fig. 6.9b show the effects of observation noise on g and cost C, respectively. We are interested only in the cost of moving from one depth of the quad-tree to another. The cost incurred by MVA U_i at time step t is defined as

$$C_i^t = \varepsilon - d_i, \tag{6.5}$$

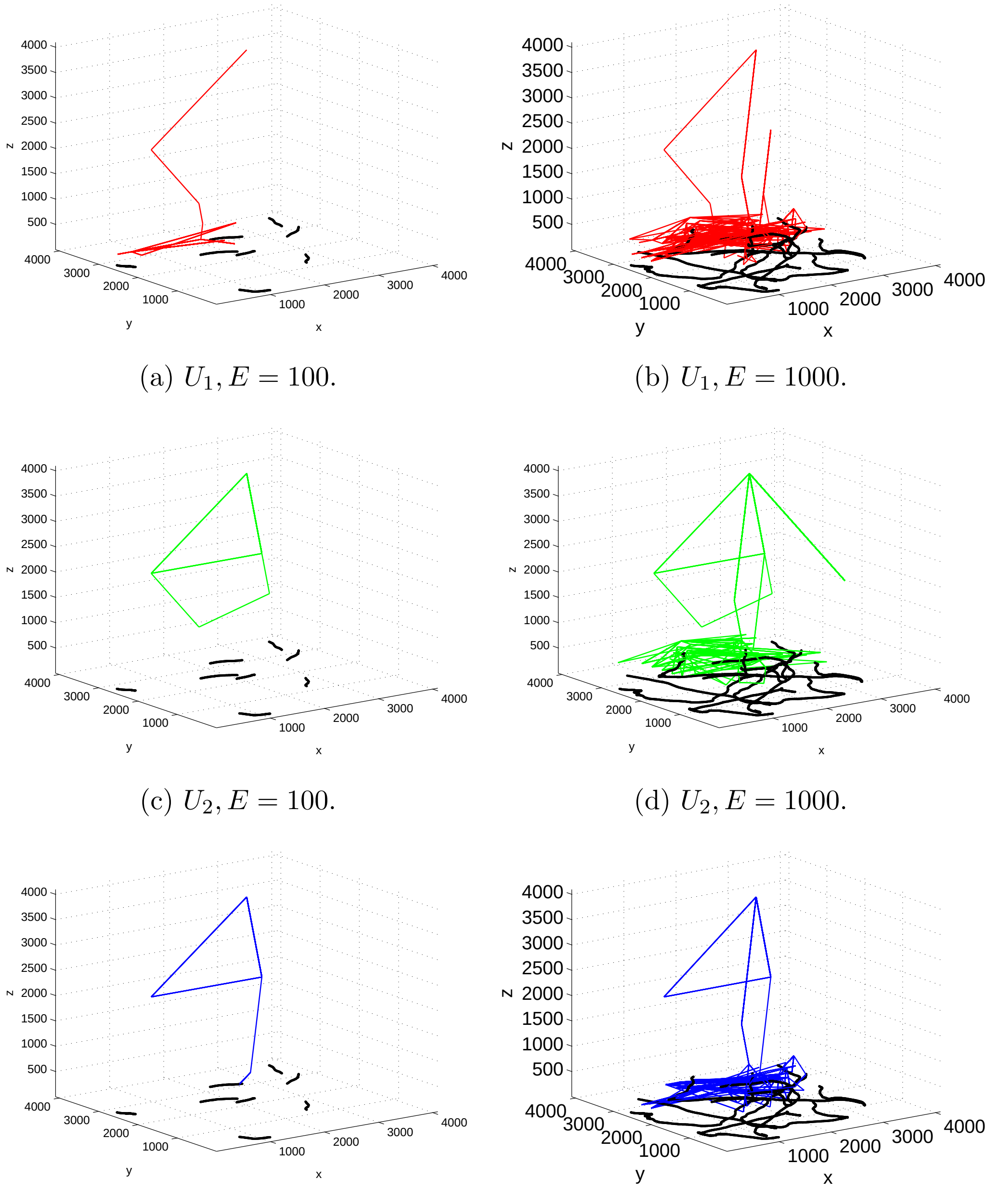


(a) $\alpha = 0.9$.



(b) $\alpha = 0.1$.

Figure 6.4: Observations and their associated quad-tree depths for $B = 8$ targets and $A = 3$ UAVs.



(a) $U_1, E = 100$. (b) $U_1, E = 1000$.

(c) $U_2, E = 100$. (d) $U_2, E = 1000$.

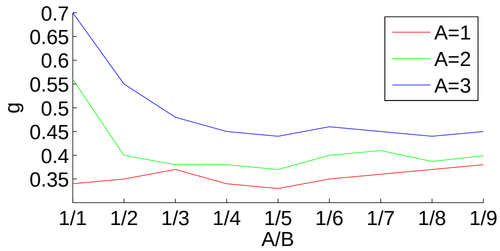
(e) $U_3, E = 100$. (f) $U_3, E = 1000$.

Figure 6.5: Sample paths of $A = 3$ UAVs ($\alpha = 0.1$).

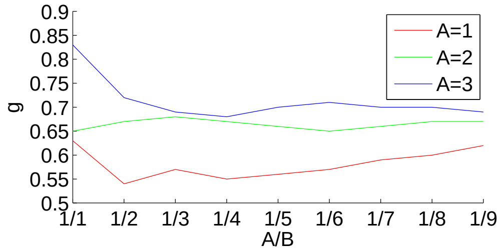
and the collective cost is

$$C = \sum_{t=1}^{\Gamma} \sum_{i=1}^{A} C_i^t. \tag{6.6}$$

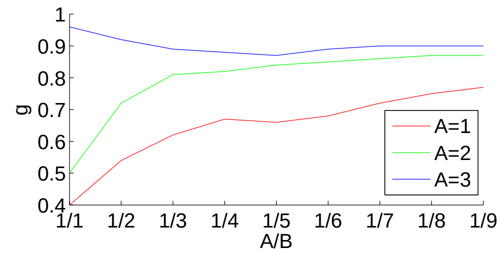
The results in Fig. 6.9a and Fig. 6.9b are average values of 100 simulation runs. While an increased sensor noise does not affect g, it increases the movement



(a) $\alpha = 0.1$



(b) $\alpha = 0.5$



(c) $\alpha = 0.9$

Figure 6.6: The effect of the ratio A/B on the resolution of observation for different values of α.

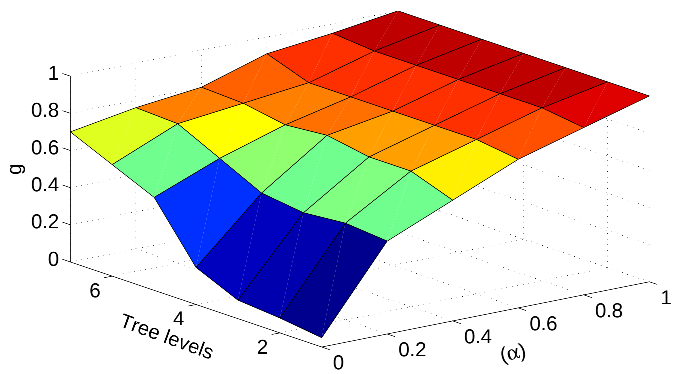
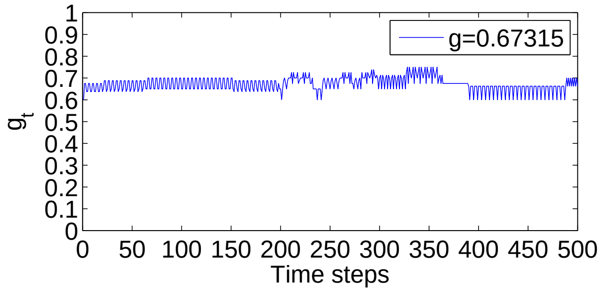
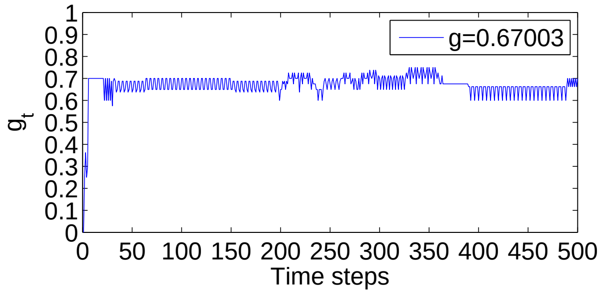


Figure 6.7: The effect of the quad-tree size on g ($A = 3, B = 8$).

cost by moving the UAVs upwards.

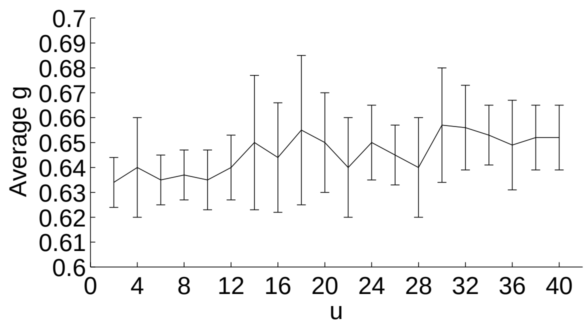


(a) UAVs start at the root node.

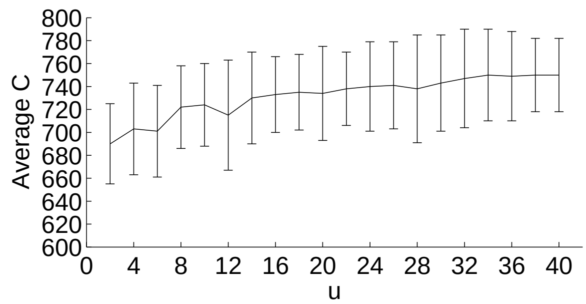


(b) UAVs start at randomly selected leaf nodes.

Figure 6.8: The effect of UAV location initialization ($A = 3,\ B = 8,\ \alpha = 0.5$).



(a)



(b)

Figure 6.9: The effect of observation noise $\mathbf{O} = u \times I_{4\times4}$ ($A = 3,\ B = 8,\ \alpha = 0.5$).

6.4 Summary

We presented a quad-tree based centralized movement strategy for a team of UAVs to maximize the collective time and quality of observation for multiple moving targets. We also compared two mobility options on the quad-tree for movement of UAVs. This movement strategy enables a team of UAVs to work together towards a common goal of maximizing observation of large group of moving targets. The proposed method is suitable not only for aerial robots that can move in 3D space, but also for sensors that can control the location and zoom level of their FOVs. We presented only centralized movement strategy, which may not be feasible in the presence of communication range and bandwidth limitations. The detection error as presented in Chapter 4 and Chapter 5 is a major sensing limitation, which needs to be investigated for multi-scale observation of moving targets. We have considered only one reason of variations in the size of FOV. However, there are other reasons of variations in the size and shape of FOV, which can be considered to improve the multi-scale cooperative observation of moving targets.

Chapter 7

Conclusion and Future Work

This chapter concludes the thesis by a summary of its contributions and an outlook to future research directions.

7.1 Summary

The objective of this research effort is to develop approaches for a team of multiple autonomous and mobile robots, especially resource-limited small-scale UAVs, to perform aerial surveillance of predefined targets in a given environment. The focus is on two applications of aerial surveillance: (i) cooperative search of stationary targets, and (ii) cooperative observation of moving targets. Coordination among the individual team members, which refers to sharing of information and joint decision-making, is a key towards the success of such surveillance systems.

The thesis presents several contributions to the field of multi-UAV aerial surveillance.

First, a survey of the approaches in a larger domain of multi-robot systems for the surveillance of pre-defined targets in a given environment has been presented. Various factors and application scenarios have been identified that affect the performance of multi-robot surveillance systems. Research works from the last 20 years have been classified to understand the state-of-the-art on the stated topic. This larger domain of multi-robot systems has been limited to resource-limited multi-UAV systems for the development of coordination approaches.

Second, a formalization of the problem has been proposed that models the UAV team, the targets, the environment, and limitations in sensing and communication. The UAVs need to communicate with each other to share information and to make joint decision about their paths. The research problem is that of coordinated and dynamic UAV placement. The UAVs must know how to coordinate for determining their time varying locations in order to achieve the application specific goal.

Third, a distributed strategy for merging delayed and incomplete information about the environment has been presented. The delayed and incomplete information is a result of sensing and communication limitations of UAVs. A trade-off in time of cooperative search and detection errors has been shown. Additionally, the effects of various sensing and communication parameters on cooperative search have been presented.

Fourth, the analytic analysis of the number of observations required for collecting information has been performed. This analysis helps in declaring with minimum efforts the absence or existence of a target in a region. This number of required observations has been integrated into a formal system model for cooperative search with constraints in sensing, information exchange, and network connectivity.

Fifth, an iterative use of the required number of sensor observation in the Travelling Salesman Problem (TSP) and Multiple Travelling Salesmen Problem (MTSP) formulation has been proposed for autonomous path planning of UAVs.

Sixth, an exploration of the algorithmic design space has been performed and different algorithms have been developed to analyze the effects of centralized and distributed coordination in the presence of sensing and communication limitations.

Finally, the application of UAVs for observing multiple moving targets with different resolutions has been proposed. The UAVs have the ability to hover and move in 3D environment, which enables them to observe the environment at different spatial scales (resolutions). This scale of observations has been associated to the quality of observation of a target. A key contribution is to use the quad-tree data-structure for modelling the environment and movement of UAVs. This modelling has helped in the dynamic sensor placement of UAVs to maximize the collective time and quality of observation for multiple moving targets. The proposed method is suitable not only for aerial robots that can move in 3D space, but also for sensors that can control the location and zoom level of their FOVs.

7.2 Future Directions

The latest developments in technology and the changed acceptance of robots by human society stimulate major advances in various robotic applications. We can identify the following trends. First, the results of research are turning into products and services, which are influencing many aspects of human life. Second, we see a tremendous growth in the number of small and low cost robots for accomplishing a task. This increase in the number and reduction in size and cost of robots emphasizes the need for advanced methods of coordination among the robots. Third, the level of cognition and autonomy, i.e., the ability to plan and execute tasks in response to the high level commands, in modern robots are increasing. This significantly influences the research, development, and market of robots. Fourth, we see a closer interaction of the modern robots with people. Previously, robots were confined to isolated places in manufacturing plants and now the robots cooperate and share space with people. This close interaction with the people increases the performance of robotic applications but also introduces safety, ethical, legal and societal issues. Finally, other robotic technologies are emerging, which include bio-inspired, nano and cloud robots. A more in-depth discussion about current trends in robotics can be found, for example, in the European Strategic Research Agenda [41]. Considering the recent trends in robotics, the work presented in this thesis shows several directions for future research on multi-robot systems, especially multi-UAV surveillance systems.

7.2.1 Sensing

The most important challenge is the coordination among the UAVs to share the sensing resources for improving the perception of the environment. This challenge is due to sensing limitations, which are (i) limited sensor performance (limited detection, errors etc.), (ii) limited FOV, and (iii) limited observability (occlusions etc.). The associated hardware and software of a surveillance sensor are not perfect and introduce errors in the sensing process, which affects sensor performance. The sensor's FOV is usually very limited such that the union of the areas of all the sensors' FOVs is smaller than the area to be monitored. This limitation is a major reason for multiple sensor deployment and their motion planning. The challenge of sensing limitation is severe in structured environments when buildings, trees, obstacles etc., occlude the observations. These sensing limitations generate uncertain, unreliable, and outdated information. To deal with this challenge, it requires accurate interpretations of the real output of the sensor (e.g., image or video) and all types of potential errors (e.g., measurement noise, false positive, missed detections) in reporting the locations of the targets or recording different features of the environment. Sharing the sensing resources also requires the advanced methods for merging of information (local and distributed sensor fusion) to enable fusion of real outputs of the sensors. Combining information from multiple sensors either mounted on the same UAV or on different UAVs, has a great potential for increasing the perception of the environment. Information fusion from heterogeneous sensors e.g., vision sensor and laser scanner, vision sensor and GPS sensor, can increase robustness against sensing errors and occluded views. This can also increase the accuracy of distance and orientation estimation of targets and environment features as shown for a static sensor network [23]. It has already been shown that information fusion can improve camera calibration and sensor movement in single robot applications [104]. The same approach can be investigated for cooperative UAVs. Moreover, information fusion can also be employed to improve the UAV state estimation with respect to environment, targets and other UAVs. New techniques are required to enhance sensor fusion in both non-distributed and distributed ways.

7.2.2 Networking

The functionality of multi-UAV systems depends greatly on their networking capabilities and timely information exchange, especially in time critical missions. Here, we highlight some important networking issues. First, the wireless communication among the UAVs is always limited, which hinders the smooth and uninterrupted information exchange among the robots. The permanent or temporary loss of connectivity hinders a UAV from access to updated information by other UAVs, which is important in applications with highly dynamic environments. Second, the communication requirements of UAVs moving in 3D environments [7, 131] are different from those of ground robots. The communication problems severely affect the centralized coordinated UAVs by isolating (permanently or temporarily) one or more robots from the centrally available global information and decisions. However, distributed algorithms, that enable individual UAVs to operate even with partial available information, affect the cooperative UAVs marginally. Reliable networks with guaranteed QoS (bandwidth, delay etc.) are needed to cope with connectivity and time-varying network latency of highly mobile and cooperative UAVs. Delay tolerant networks must be designed to degrade gracefully when the communication is slow, unavailable, and intermittent. The development of protocols/algorithms/applications, which can deal with the various network limitations will advance the multi-UAV surveillance systems.

7.2.3 Path-planning

There is much room for improvement in multi-UAV surveillance systems by developing advanced path-planning algorithms. One example is to analyze the coordinated on-line decision making where each MAV has no path information but decides its movement at each time step. The actual duration of a single time step depends on the overall processing time. Making independent and online decisions of a UAV within a single time step may require additional processing and communication resources. The computation of the TSP/MTSP solution can become the dominant component, especially in case of large search areas. Another example is to develop strong coordination approaches for path-planning where a UAV can provide some feedback to the other team members on accepting/rejecting their decisions. The UAVs can use market based strategies (e.g., negotiations and auctions) to implement this feedback. A third example is to include additional resource constraints such as the flight time of UAVs and to consider alternative inference processes for converting a sequence of locations into paths of the UAVs. Finally, a 3D representation of the environment for movement of both the UAVs and targets and methods for collision avoidance could serve as interesting research directions.

7.2.4 Human Interaction

The potential impact of multi-UAV surveillance system will depend not only on the coordination among UAVs but also on the coordination with the human operators. Automated systems without the interaction of humans are critical since they are faster than humans in calculations and resource allocation in dynamic and time-critical environments. However, such systems may fail to respond to emergent events. Human operators can assist such systems by timely intervention and their knowledge-based reasoning and experience [30]. Humans can intervene to balance the load by assigning different targets or roles to the UAVs. Direct interaction of the human with cooperative UAVs is integrated part of advanced UAVs and will shape emerging applications. On the other hand, remote control can be hard for the operator due to communication failures, limited range of communication, limited battery life, and specific motion constraints on UAV platforms. Methods for robust perception of environment are required for a UAV in order to interact with the human operators. Studies should be conducted to examine how an operator interacts with decentralized UAVs. Based on these studies, novel human computer/automation interaction methods can be developed to provide autonomy for the UAVs and to assist human operator in control tasks of multiple UAVs. These methods should reduce the gap between human expectations and the assumptions made by the UAVs.

Bibliography

[1] Collaborative microdrones (cdrones). `http://uav.lakeside-labs.com/overview/cdrones/`, 2008.

[2] Collaborative aerial robots. `http://www.lakeside-labs.com/research/collaborative-aerial-robots/`, 2013.

[3] Self-organizing intelligent network of uavs (sinus). `http://uav.lakeside-labs.com/overview/sinus/`, 2013.

[4] Perez Alberto, Chamoso Pablo, Parra Victor, and Sanchez Antonio Juan. Ground vehicle detection through aerial images taken by a UAV. In *Proceedings of the IEEE International Conference on Information Fusion (FUSION)*, pages 1–6, 2014.

[5] Javier A Alcazar. A simple approach to multi-predator multi-prey pursuit domain. In Ali A Minai, Dan Braha, and Bar-Yam, editors, *Unifying Themes in Complex Systems*, pages 2–9. Springer Berlin Heidelberg, 2011.

[6] Ross P Anderson and Dejan Milutinovic. A stochastic approach to dubins vehicle tracking problems. *IEEE Transactions on Automatic Control*, 59(10):2801–2806, 2014.

[7] Torsten Andre, Karina Hummel, Angela Schoellig, Evsen Yanmaz, Mahdi Asadpour, Christian Bettstetter, Pasquale Grippa, Hermann Hellwagner, Stephan Sand, and Siwei Zhang. Application-driven design of aerial communication networks. *IEEE Communication Magazine*, 52(5):129–137, 2014.

[8] Tamio Arai, Enrico Pagello, and Lynne E. Parker. Guest editorial: advances in multirobot systems. *IEEE Transactions on Automation, Science and Engineering*, 18(5):655–661, 2002.

[9] Federico Augugliaro, Ammar Mirjan, Fabio Gramazio, Matthias Kohler, and Raffaello Andrea. Building tensile structures with flying machines. In

Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 3487–3492, 2013.

[10] Jacopo Banfi, Jerome Guzzi, Alessandro Giusti, Luca Gambardella, and Gianni A. Di Caro. Fair multi-target tracking in cooperative multi-robot systems. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 5411–5418, 2015.

[11] Randal W. Beard, Timothy W. McLain, Derek B. Nelson, Derek Kingston, and David Johanson. Decentralized cooperative aerial surveillance using fixed-wing miniature UAVs. *Proceedings of the IEEE*, 94(7):1306–1326, 2006.

[12] Tolga Bektas. The multiple traveling salesman problem: an overview of formulations and solution procedures. *Omega: The International Journal of Management Science*, 34:209–219, 2006.

[13] Jean Berger and Jens Happe. Co-evolutionary search path planning under constrained information-sharing for a cooperative unmanned aerial vehicle team. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC)*, pages 1–8, 2010.

[14] Luca F. Bertuccelli and Jonathan P. How. Robust UAV search for environments with imprecise probability maps. In *Proceedings of the IEEE Conference on Decision and Control - European Control Conference (CDC-ECC)*, pages 5680–5685, 2005.

[15] Francisco Bonin-Font, Alberto Ortiz, and Gabriel Oliver. Visual navigation for mobile robots: A survey. *Journal of Intelligent Robotic Systems*, 53:263–296, 2008.

[16] Richard Borie, Sven Koenig, and Craig Tovey. Section 9.5: Pursuit-evasion problems. In J. Gross, J. Yellen, and P. Zhang, editors, *Handbook of Graph Theory*, pages 1145–1165. Chapman and Hall/CRC, 2013.

[17] Frederic Bourgault, Tomonari Furukawa, and Hugh F. Durrant-Whyte. Coordinated decentralized search for a lost target in a Bayesian world. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 48–53, 2003.

[18] Wolfram Burgard, Mark Moors, Dieter Fox, Reid Simmons, and Sebastian Thrun. Collaborative multi-robot exploration. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 476–481, 2000.

[19] Ze-Su Cai, Li-Ning Sun, Hai-Bo Gao, Pu-Cheng Zhou, Song-Hao Piao, and Qing-Cheng Huang. Multi-robot cooperative pursuit based on task bundle auctions. In Caihua Xiong, Yongan Huang, Youlun Xiong, and Honghai Liu, editors, *Intelligent Robotics and Applications*, volume 5314, pages 235–244. Springer Berlin Heidelberg, 2008.

[20] Yongcan Cao, Wenwu Yu, Wei Ren, and Guanrong Chen. An overview of recent progress in the study of distributed multi-agent Coordination. *IEEE Transactions on Industrial Informatics*, 9(1):427–438, 2013.

[21] Yu Cao, Alex Fukunaga, and Andrew Kahng. Cooperative mobile robotics: antecedents and directions. *Autonomous Robots*, 4:1–13, 1997.

[22] Howie Choset and Philippe Pignon. Coverage path planning: the boustrophedon cellular decomposition. In *Proceedings of International Conference on Field and Service Robotics*, pages 1–7, 1997.

[23] Changmook Chun, SeungBeum Suh, Chi won Roh, Yeonsik Kang, Sungchul Kang, Jung yup Lee, and Chang soo Han. Sensor fusion-based line detection for unmanned navigation. In *Proceedings of the IEEE Intelligent Vehicle Symposium*, pages 191–196, 2010.

[24] Timothy H. Chung and Joel W. Burdick. Multi-agent probabilistic search in a sequential decision-theoretic framework. In *Proceedings of the IEEE International Conferences on Robotics and Automation (ICRA)*, pages 146–151, 2008.

[25] Timothy H. Chung and Joel W. Burdick. Analysis of search decision-making using probabilistic search strategies. *IEEE Transactions on Robotics*, 28(1):132–144, 2012.

[26] Timothy H. Chung, Vijay Gupta, Joel W. Burdick, and Richard M. Murray. On a decentralized active sensing stragegy using mobile sensor platforms in a network. In *Proceedings of the IEEE Conference on Decision and Control*, pages 1914–1919, 2004.

[27] Timothy H Chung, Geoffrey A Hollinger, and Volkan Isler. Search and pursuit-evasion in mobile robotics: A survey. *Autonomous Robots*, 31:299–316, 2011.

[28] Richard A Coogle and Ayanna Howard. Robotic resource allocation for the observation of oblating target sources. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, pages 4002–4007, 2014.

[29] Richard A Coogle and Ayanna M Howard. The iceberg observation problem: Using multiple agents to monitor and observe ablating target sources. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, pages 1660–1665, 2013.

[30] Missy Cummings, Jonathan P How, A Whitten, and O Toupet. The impact of human-automation collaboration in decentralized multiple unmanned vehicle control. *Proceedings of the IEEE*, 100(3):660–671, 2012.

[31] Kai Daniel, Sebastian Rohde, Niklas Goddemeier, and Christian Wietfeld. Cognitive agent mobility for aerial sensor networks. *IEEE Sensor Journal*, 11(11):2671–2682, 2011.

[32] Mark de Berg, Otfried Cheong, Marc van Kreveld, and Mark Overmars. *Computational Geometry: Algorithms and Applications*. Springer-Verlag.

[33] M. Bernardine Dias, Robert Zlot, Nidhi Kalra, and Anthony Stentz. Market-based multi-robot coordination: A survey and analysis. *Proceedings of IEEE*, 94(7):1257–1270, 2006.

[34] Yingying Ding and Yan He. Flexible formation of the multi-robot system and its application on cmommt problem. In *Proceedings of the 2nd International Asia Conference on Informatics in Control, Automation and Robotics*, pages 377–382, 2010.

[35] Yingying Ding, Miaoliang Zhu, Yan He, and Jingping Jiang. P-cmommt algorithm for the cooperative multi-robot observation of multiple moving targets. In *Proceedings of the 6th World Congress on Intelligent Control and Automation*, pages 9267–9271, 2006.

[36] Joseph W. Durham, Antonio Franchi, and Francesco Bullo. Distributed pursuit-evasion without mapping or global localization via local frontiers. *Autonomous Robots*, 32:81–95, 2012.

[37] Mohamed Elbanhawi and Milan Simic. Sampling-based robot motion planning: A review. *IEEE Access*, 2:56–77, 2014.

[38] Alberto Elfes. Occupancy grids: A stochastic spatial representation for active robot perception. In *Proceedings of Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 60–70, 1990.

[39] Ahmed M. Elmogy and Fakhreddine O. Karray. Cooperative multi target tracking using multi sensor network. *In ternational Journal on Smart Sensing and Intelligent Systems*, 1(3):716–734, 2008.

[40] Dale Enns, Dan Bugajski, and Steve Pratt. Guidance and control for cooperative search. In *Proceedings of the IEEE American Control Conference (ACC)*, volume 3, pages 1923–1929, 2002.

[41] euRobotics. Strategic research agenda for robotics in europe 2014-2020. `http://www.eu-robotics.net/cms/upload/PPP/SRA2020_SPARC.pdf`, 2014.

[42] Alessandro Farinelli, Luca Iocchi, and Daniele Nardi. Multi-robot systems: A classification focused on coordination. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 34(5):2015–2028, 2004.

[43] Delle Fave, Francesco Maria, Zhe Xu, Alex Rogers, and Nicholas R Jennings. Decentralized coordination of unmanned aerial vehicles for target search using the max-sum algorithm. In *Proceedings of Adaptive Agents and Multi-agent Systems (AAMAS)*, pages 35–44, 2010.

[44] Silvia Ferrari, Rafael Fierro, and Domagoj Tolic. A geometric optimization approach to tracking maneuvering targets using a heterogeneous mobile sensor network. In *Proceedings of the IEEE Conference on Decision and Control*, pages 1080–1087, 2009.

[45] Matthew Flint, Marios Polycarpou, and Eninianuel Fernandez-Gaucherand. Cooperative control for multiple autonomous UAV's searching for targets. In *Proceedings of the IEEE Conference on Decision and Control (CDC)*, pages 2823–2828, 2002.

[46] Erik J. Forsmo, Esten I. Grotli, Thor I. Fossen, and Tor A. Johansen. Optimal search mission with unmanned aerial vehicles using mixed integer linear programming. In *Proceedings of the IEEE International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 253–259, 2013.

[47] Eric W. Frew and Jack Elston. Target assignment for integrated search and tracking by active robot networks. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2354–2359, 2008.

[48] Yinfei Fu, Qing Ling, and Zhi Tian. Distributed sensor allocation for multi-target tracking in wireless sensor networks. *IEEE Transactions on Aerospace and Electronic Systems*, 48(4):3538–3553, 2012.

[49] Yinfei Fu and Le Yang. Sensor mobility control for multitarget tracking in mobile sensor networks. *International Journal of Distributed Sensor Networks*, pages 1–15, 2014.

[50] Tomonari Furukawa, Frederic Bourgault, B. Lavis, and Hugh F. Durrant-Whyte. Recursive Bayesian search-and-tracking using coordinated UAVs for lost targets. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2521–2526, 2006.

[51] Jianhua Ge, Liang Tang, Johan Reimann, and George Vachtsevanos. Hierarchical decomposition approach for pursuit-evasion differential game with multiple players. In *Procceedings of the IEEE Aerospace Conference*, pages 1–7, 2006.

[52] Alvaro E. Gil, Kevin M. Passino, and Jose B. Cruz Jr. Stable cooperative surveillance with information flow constraints. *IEEE Transactions on Control Systems Technology*, 16(5):856–868, 2008.

[53] Chad Goerzen, Zhaodan Kong, and Berenice Mettler. A survey of motion planning algorithms from the perspective of autonomous UAV guidance. *Journal of Intelligent and Robotic Systems*, 100:57 – 65, 2010.

[54] Ben Grocholsky, Alexei Makarenko, and Hugh Durrant-Whyte. Information-theoretic coordinated control of multiple sensor platforms. In *Proceedings of the IEEE International Conferrocs on Robotics and Automation (ICRA)*, pages 1521–1526, 2003.

[55] Guangjie Han, Chenyu Zhang, Lei Shu, Ning Sun, and Qingwu Li. A survey on deployment algorithms in underwater acoustic sensor networks. *International Journal of Distributed Sensor Networks*, 2013:1–11, 2013.

[56] Gabriel M. Hoffmann, Steven L. Waslander, and Claire J. Tomlin. Distributed cooperative search using information-theoretic costs for particle filters, with quadrotor applications. In *Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit*, pages 1–21, 2006.

[57] Jonathan P. How, Cameron Fraser, Karl C. Kulling, Luca F. Bertuccelli, Olivier Toupet, Luc Brunet, Abe Bachrach, and Nicholas Roy. Increasing autonomy of UAVs. *IEEE Robotics and Automation Magazine*, 16:43–51, 2009.

[58] Jinwen Hu, Lihua Xie, Kai-Yew Lum, and Jun Xu. Multiagent information fusion and cooperative control in target search. *IEEE Transactions on Control Systems Technology*, 21(4):1223–1235, 2013.

[59] Jinwen Hu, Lihua Xie, Jun Xu, and Zhao Xu. Multi-agent cooperative target search. *Sensors*, 14(6):9408–9428, 2014.

[60] Todd Humphreys. Cockrell school researchers demonstrate first successful spoofing of uavs. `http://www.engr.utexas.edu/features/humphreysspoofing`, 2012.

[61] Victor M Jiménez and Andrés Marzal. Computing the k shortest paths: A new algorithm and an experimental comparison. In Christos D. Zaroliagis Jeffrey S. Vitter, editor, *Algorithm Engineering (Lecture Notes in Computer Science)*, pages 15–29. Springer Berlin Heidelberg, 1999.

[62] Shiyuan Jin and Zhihua Qu. A heuristic task scheduling for multi-pursuer multi-evader games. In *Proceedings of the IEEE International Conference on Information and Automation (ICIA)*, pages 528–533, 2011.

[63] Boyoon Jung and Gaurav S Sukhatme. A generalized region-based approach for multi-target tracking in outdoor environments. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 2764–2769, 2002.

[64] Boyoon Jung and Gaurav S Sukhatme. A region-based approach for cooperative multi-target tracking in a structured environment. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 2189–2195, 2004.

[65] Boyoon Jung and Gaurav S Sukhatme. A region-based approach for cooperative multi-target tracking in a structured environment. In *Proceedings International Symposium on Distributed Autonomous Robotic Systems*, pages 81–90, 2006.

[66] Seema Kamath, Eric Meisner, and Volkan Isler. Triangulation based multi target tracking with mobile sensor networks. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3283–3288, 2007.

[67] Natallia Katenkaa, Elizaveta Levinab, and George Michailidis. Tracking multiple targets using binary decisions from wireless sensor networks. *Journal of the American Statistical Association*, 108(502):398–410, 2013.

[68] Ben Kehoe, Sachin Patil, Pieter Abbeel, and Ken Goldberg. A survey of research on cloud robotics and automation. *IEEE Transactions on Automation, Science and Engineering*, 12(2):398–407, 2015.

[69] Amirreza M. Khaleghi, Dong Xu, Zhenrui Wang, Mingyang Li, Alfonso Lobos, Jian Liu, and Young-Jun Son. A DDDAMS-based planning and control framework for surveillance and crowd control via UAVs and UGVs. *Expert Systems with Applications*, 40(18):7168–7183, 2013.

[70] Asif Khan and Bernhard Rinner. Coordination in micro aerial vehicles for search operations. In *Proceedings of the Austrian Robotics Workshop*, pages 1–2, 2015.

[71] Asif Khan, Bernhard Rinner, and Andrea Cavallaro. Multi-scale observation of multiple moving targets using micro aerial vehicles. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 1 – 8, 2015.

[72] Asif Khan, Bernhard Rinner, and Andrea Cavallaro. Observing moving targets by cooperative mobile robots: A survey. *IEEE Transactions on Automation Science and Engineering (Submitted)*, pages 1–13, 2015.

[73] Asif Khan, Evsen Yanmaz, and Bernhard Rinner. Information merging in Multi-UAV cooperative search. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, pages 3122–3129, 2014.

[74] Asif Khan, Evsen Yanmaz, and Bernhard Rinner. Information exchange and decision making in micro aerial vehicle networks for cooperative search. *IEEE Transactions on Control of Networked Systems*, In press, 2015.

[75] Jongrae Kim and John L. Crassidis. UAV path planning for maximum visibility of ground targets in an urban area. In *Proceedings of IEEE Conference on Information Fusion*, pages 1–7, 2010.

[76] Min-Hyuk Kim, Hyeoncheol Baik, and Seokcheon Lee. A response threshold model based UAV search planning and task allocation. *Journal of Intelligent and Robotic Systems*, 28(1):132–144, 2013.

[77] Mark P. Kolba, Waymond R. Scott, and Leslie M. Collins. A Framework for Information-Based Sensor Management for the Detection of Static Targets. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 41(1):105–120, 2011.

[78] Andreas Kolling and Stefano Carpin. Cooperative observation of multiple moving targets: an algorithm and its formalization. *The International Journal of Robotics Research*, 26(9):935–953, 2007.

[79] Andreas Kolling and Carpin Stefano. Multirobot cooperation for surveillance of multiple moving targets - a new behavioral approach. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1311–1316, 2006.

[80] Juliane Kuhn, Christian Reinl, and Oskar von Stryk. Predictive control for multi-robot observation of multiple moving targets based on discrete-continuous linear models. In *Proceedings of the 18th World Congress, The International Federation of Automatic Control*, pages 257–262, 2011.

[81] Vijay Kumar and Nathan Michael. Opportunities and challenges with autonomous micro aerial vehicles. *International Journal of Robotics Research*, 31(11):1279–1291, 2012.

[82] Hung Manh Laa and Weihua Shengb. Dynamic target tracking and observing in a mobile sensor network. *Robotics and Autonomous Systems*, 60(7):996–1009, 2012.

[83] Colin Lecher. How will drones change sports. `http://www.popsci.com/article/technology/how-will-drones-change-sports`, 2014.

[84] Geunho Lee, Nak Young Chong, and Henrik Christensen. Tracking multiple moving targets with swarms of mobile robots. *Intelligent Service Robotics*, 3(2):61–72, 2010.

[85] Brenda Livingston. The role of uavs/mavs in domestic surveillance and control. `http://www.rense.com/general47/uav.htm`.

[86] Sean Luke, Keith Sullivan, Liviu Panait, and Gabriel Balan. Tunably decentralized algorithms for cooperative target observation. In *Proceedings of the Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 911–917, 2005.

[87] Christopher Lum, Rolf T. Rysdyk, and Juris Vagners. A search algorithm for teams of heterogeneous agents with coverage guarantees. *AIAA Journal of Aerospace Computing, Information, and Communication*, 7:1–31, 2010.

[88] Luis Merino, Fernando Caballero, Martínez de Dios, Joaquín Ferruz, and Aníbal Ollero. A cooperative perception system for multiple uavs: Application to automatic detection of forest fires. *Journal of Field Robotics*, 23:165–184, 2006.

[89] Nathan Michael, Daniel Mellinger, Quentin Lindsey, and Vijay Kumar. The GRASP multiple micro UAV testbed. *IEEE Robotics and Automation Magazine*, 17(3):56–65, 2010.

[90] Mostafa Mirzaei, Farid Sharifi, Brandon W. Gordon, C. A. Rabbath, and Youmin Zhang. Cooperative multi-vehicle search and coverage problem

in uncertain environments. In *Proceedings of the IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, pages 4140–4145, 2011.

[91] Songhwai Oh, Luca Schenato, Phoebus Chen, and Shankar Sastry. Tracking and coordination of multiple agents using sensor networks: System design, algorithms and experiments. *Proceedings of IEEE*, 95(1):234–253, 2007.

[92] Daniel J. Pack, Pedro DeLima, Gregory J. Toussaint, and George York. Cooperative control of UAVs for localization of intermittently emitting mobile targets. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 39:959–970, 2009.

[93] Lynne E. Parker. Distributed Algorithms for Multi-Robot Observation of Multiple Moving Targets. *Autnomous Robots*, 12:231–255, 2002.

[94] Lynne E. Parker. Decision making as optimization in multi-robot teams. In *Proceedings of International Conference on Distributed Computing and Internet Technology*, pages 1–15, 2012.

[95] Lynne E. Parker and Brad A Emmons. Cooperative multi-robot observation of multiple moving targets. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2082–2089, 1997.

[96] Lynne E. Parker and Claude Touzet. Multi-robot learning in a cooperative observation task. In *In Distributed Autonomous Robotic Systems 4*, pages 391–401. Springer, 2000.

[97] Kevin Passino, Marios Polycarpou, David Jacques, Meir Pachter, Yang Liu, Yanli Yang, Matt Flint, and Michael Baum. Cooperative control for autonomous air vehicles. In Robert Murphey and Panos M. Pardalos, editors, *Cooperative Control and Optimization*, pages 233–271. Springer US, 2002.

[98] Claudio Piciarelli, Lukas Esterle, Asif Khan, Bernhard Rinner, and Gian Luca Foresti. Dynamic reconfiguration in camera networks: a short survey. *IEEE Transactions on Circuits and Systems for Video Technology*, In press, 2015.

[99] Ryan R. Pitre, X. R. Li, and R. Delbalzo. UAV route planning for joint search and track missions: An information-value approach. *IEEE Transactions on Aerospace and Electronic Systems*, 48:2551–2565, 2012.

[100] Markus Quaritsch, Karin Kruggl, Daniel Wischounig-Strucl, Subhabrata Bhattacharya, Mubarak Shah, and Bernhard Rinner. Networked UAVs as aerial sensor network for disaster management applications. *Elektrotechnik und Informationstechnik, Special Issue on Wireless Sensor Networks*, 127(3):56–63, 2010.

[101] Markus Quaritsch, Emil Stojanovski, Christian Bettstetter, Gerhard Friedrich, Hermann Hellwagner, Hermann Hofbaur, Mubarak Shah, and Bernhard Rinner. Collaborative microdrones: applications and research challenges. In *Proceedins the Second International Conference on Autonomic Computing and Communication Systems*, pages 1–7, 2008.

[102] James R. Riehl, Gaemus E. Collins, and Joao P. Hespanha. Cooperative search by UAV teams: A model predictive approach using dynamic graphs. *IEEE Transactions on Aerospace and Electronic Systems*, 47(2):2637–2656, 2011.

[103] Airam Rodriguez, Juan J. Negro1, Mara Mulero1, Carlos Rodriguez, Jesus Hernandez-Pliego, and Javier Bustamante. The Eye in the Sky: Combined Use of Unmanned Aerial Systems and GPS Data Loggers for Ecological Research and Conservation of Small Birds. *PLoS ONE*, 7(12):1–6, 2012.

[104] Davide Scaramuzza, Roland Siegwart, and Agostino Martinelli. A robust descriptor for tracking vertical lines in omnidirectional images and its use in mobile robotics. *The International Journal of Robotics Research*, 28(2):149–171, 2009.

[105] Jürgen Scherer, S. Yahyanejad, S. Hayat, E. Yanmaz, T. Andre, A. Khan, V. Vukadinovic, C. Bettstetter, H. Hellwagner, and Bernhard Rinner. An autonomous multi-UAV system for search and rescue. In *Proceedings of the First Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use: MobiSys2015*, pages 33–38, 2015.

[106] Melanie Schranz and Bernhard Rinner. Resource-aware dynamic clustering utilizing state estimation in visual sensor networks. *Sensors and Transducers*, 191(8):28–39, 2015.

[107] Mac Schwager, Brian J. Julian, Michael Angermann, and Daniela Rus. Eyes in the sky: decentralized control for the deployment of robotic camera networks. *Proceedings of the IEEE*, 99(9):1541–1561, 2011.

[108] Abhijit Sinha, Thia Kirubarajan, and Yaakov Bar-Shalom. Autonomous ground target tracking by multiple cooperative UAVs. In *Proceedings of the IEEE Aerospace Conference*, pages 1–9, 2005.

[109] P. B. Sujit and Debasish Ghose. Search using multiple UAVs with flight time constraints. *IEEE Transactions on Aerospace and Electronic Systems*, 40(2):491–509, 2004.

[110] P. B. Sujit and Debasish Ghose. Self assessment-based decision making for multiagent cooperative search. *IEEE Transactions on Automation Science and Engineering*, 8:705–719, 2011.

[111] Zhijun Tang and Ümit Özgüner. Motion planning for multitarget surveillance with mobile sensor agents. *IEEE Transactions on Robotics*, 21(5):898–908, 2005.

[112] Theodoros Theodoridis and Huosheng Hu. Toward Intelligent Security Robots: A survey. *IEEE Transactions on Systems, Man, and Cybernetics — Part C: Applications and Reviews*, 42(6):1219–1229, 2012.

[113] John Tisdale, Zuwhan Kim, and J. Hedrick. Autonomous UAV path planning and estimation. *IEEE Robotics and Automation Magazine*, 16(2):35 – 42, 2009.

[114] Domagoj Tolic and Rafael Fierro. Adaptive sampling for tracking in pursuit-evasion games. In *Procceedings of the IEEE International Symposium on Intelligent Control*, pages 179–184, 2011.

[115] Domagoj Tolic, Rafael Fierro, and Silvia Ferrari. Cooperative multi-target tracking via hybrid modeling and geometric optimization. In *Proceedings of the Mediterranean Conference on Control and Automation*, pages 440–445, 2009.

[116] Rene Vidal, Omid Shakernia, H Jin Kim, David Hyunchul Shim, and Shankar Sastry. Probabilistic pursuit-evasion games: Theory, implementation, and experimental evaluation. *IEEE Transactions on Robotics and Automation*, 18(5):262–269, 2002.

[117] Patrick Vincent and Izhak Rubin. A framework and analysis for cooperative search using UAV swarms. In *Proceedings of the ACM symposium on Applied computing*, pages 79–86, 2004.

[118] Sonia Waharte and Niki Trigoni. Supporting search and rescue operations with UAVs. In *Proceedings of the IEEE International Conference on Emerging Security Technologies (EST)*, pages 142–147, 2010.

[119] Sonia Waharte, Niki Trigoni, and Simon Justin Julier. Coordinated search with a swarm of UAVs. In *Proceedings of the IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, pages 1–3, 2009.

[120] Barry Brian Werger and Maja J Mataric. Broadcast of local eligibility for multi-target observation. In Lynne E Parker, George Bekey, and Jacob Barhen, editors, *Distributed Autonomous Robotic Systems 4*, pages 347–356. Springer Japan, 2000.

[121] Richard A. Wise and Rolf T. Rysdyk. UAV Coordination for Autonomous Target Tracking. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, pages 1–22, 2006.

[122] Kyle Hollins Wray and Benjamin B. Thompson. An application of multi-agent learning in highly dynamic environments. In *Proceedings of AAAI Workshop on Multiagent Interaction without Prior Coordination (MIPC)*, pages 1–7, 2014.

[123] Pradeep K. Yadav and Wei Meng. Mobile targets localization in a field area using moving Gaussian peaks and probability map. In *Proceedings of the IEEE International Conference on Control and Automation (ICCA)*, pages 416–421, 2014.

[124] Saeed Yahyanejad and Bernhard Rinner. A fast and mobile system for registration of low-altitude visual and thermal aerial images using multiple small-scale uavs. *ISPRS Journal of Photogrammetry and Remote Sensing*, 104:189–202, 2015.

[125] Saeed Yahyanejad, Daniel Wischounig-Strucl, Markus Quaritsch, and Bernhard Rinner. Incremental mosaicking of images from autonomous, small-scale UAVs. In *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 329–336, 2010.

[126] Zhi Yan, Nicolas Jouandeau, and Arab Ali Cherif. A survey and analysis of multi-robot coordination. *International Journal of Advanced Robotic Systems*, 10:1–18, 2013.

[127] Yanli Yang, Ali. a. Minai, and Marios M. Polycarpou. Decentralized cooperative search by networked UAVs in an uncertain environment. In *Proceedings of the IEEE Americal Control Conference (ACC)*, pages 5558–5563, 2004.

[128] Yanli Yang, Marios M. Polycarpou, and Ali A. Minai. Multi-UAV cooperative search using an opportunistic learning method. *Journal of Dynamic Systems, Measurement, and Control*, 129(5):716–728, 2007.

[129] Evsen Yanmaz and Hasan Guclu. Stationary and mobile target detection using mobile wireless sensor networks. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, pages 1–5, 2010.

[130] Benjamini Yoav and Hochberg Yosef. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 57:289–300, 1995.

[131] Seokhoon Yoon and Chunming Qiao. Cooperative search and survey using autonomous underwater vehicles (AUVs). *IEEE Transactions on Parallel and Distributed Systems*, 22(3):364–379, 2011.

[132] George York and Daniel J. Pack. Ground target detection using cooperative unmanned aerial systems. *Journal of Intelligent Robotic Systems*, pages 473–478, 2012.

[133] Dimitri Zarzhitsky, Diana F. Spears, and William M. Spears. Swarms for chemical plume tracing. In *Proceedings of the IEEE Swarm Intelligence Symposium*, pages 249–256, 2005.

[134] Zheng Zeng, Andrew Lammas, Karl Sammut, Fangpo He, Youhong Tang, and Qijin Ji. Path planning for rendezvous of multiple AUVs operating in a variable ocean . In *Proceedins the IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems*, pages 451–456, 2014.

[135] Minyi Zhong and Christos G. Cassandras. Distributed coverage control and data collection with mobile sensor networks. In *Proceedings of the IEEE Conference on Decision and Control (CDC)*, pages 5604–5609, Dec 2010.