Decentralized Control of Three Dimensional Mobile Robotic Sensor Networks

by

Valimohammad Nazarzehi had

2016
Abstract

Decentralized control of mobile robotic sensor networks is a fundamental problem in robotics that has attracted intensive research in recent decades. Most of the existing works dealt with two-dimensional spaces. This report is concerned with the problem of decentralized self-deployment of mobile robotic sensor networks for coverage, search, and formation building in three-dimensional environments. The first part of the report investigates the problem of complete sensing coverage in three-dimensional spaces. We propose a decentralized random algorithm to drive mobile robotic sensors on the vertices of a truncated octahedral grid for complete sensing coverage of a bounded 3D area. Then, we develop a decentralized random algorithm for self deployment of mobile robotic sensors to form a desired geometric shape on the vertices of the truncated octahedral grid. The second part of this report studies the problem of search in 3D spaces. We present a distributed random algorithm for search in bounded three dimensional environments. The proposed algorithm utilizes an optimal three dimensional grid for the search task.

Third, we study the problem of locating static and mobile targets in a bounded 3D space by a network of mobile robotic sensors. We introduce a novel decentralized bio-inspired random search algorithm for finding static and mobile objects in 3D areas. This algorithm combines the Levy flight random search mechanism with a 3D covering grid. Using this algorithm, the mobile robotic sensors randomly move on the vertices of the covering grid with the length of the movements follow a Levy flight distribution.

This report studies the problem of 3D formation building in 3D spaces by a network of mobile robotic sensors. Decentralized consensus-based control law for the multi-robot system which results in forming a given geometric configuration from any initial positions in 3D environments is proposed. Then, a decentralized random motion coordination law for the multi-robot system for the case when the mobile robots are unaware of their positions in the configuration in three dimensional environments is presented. The proposed algorithms use some simple consensus rules for motion coordination and building desired geometric patterns. Convergence of the mobile robotic sensors to the given configurations are shown by extensive simulations. Moreover, a mathematically rigorous proof of convergence of the proposed algorithms to the given configurations are given.
# Contents

Abstract i

1 Introduction 1
   1.1 Distributed 3D Complete Sensing Coverage and Forming Specific Shapes ......................... 3
   1.2 Distributed 3D Dynamic Search Coverage ........................................... 6
   1.3 Distributed Bio-inspired Random Search for Locating Static Targets 7
   1.4 Distributed Bio-inspired Algorithm for Search of Moving Targets 9
   1.5 Decentralized Three-dimensional Formation Building .............................. 10
   1.6 Report Contributions .................................................................. 11
   1.7 Report Organization .................................................................... 13

2 Distributed Complete Sensing Coverage and Forming Specific Shapes 15
   2.1 Problem Statement ........................................................................... 16
   2.2 Decentralized Algorithm for Self deployment of Mobile sensors for Complete coverage ...................................................... 22
      2.2.1 Simulation Results .................................................................. 24
   2.3 Self-deployment in a Given Three-dimensional Formation ...................... 28
      2.3.1 Simulation Results .................................................................. 30
   2.4 Summary ......................................................................................... 33

3 Distributed 3D Dynamic Search Coverage 35
   3.1 Problem Formulation ......................................................................... 36
   3.2 Decentralized Random Search Algorithm ............................................ 39
      3.2.1 Building a Common Three-dimensional Covering Grid 39
      3.2.2 Randomized Grid Search ......................................................... 41
   3.3 Simulation Results .......................................................................... 42
   3.4 Summary ......................................................................................... 47

4 Distributed Bio-inspired Random Search for Locating Static Targets 48
   4.1 Problem Formulation ......................................................................... 49
   4.2 Decentralized Bio-inspired Random Search Algorithm .......................... 50
   4.3 Simulation Results .......................................................................... 52
   4.4 Summary ......................................................................................... 56
CONTENTS

5 Distributed Bio-inspired Algorithm for Search of Moving Targets 58
  5.1 Problem Statement ........................................... 59
  5.2 Distributed Bio-inspired Levy Walk Random Search Algorithm .. 62
  5.3 Simulation Results .......................................... 64
  5.4 Summary ........................................................... 66

6 Decentralized Three Dimensional Formation Building 69
  6.1 Problem Formulation .......................................... 70
    6.1.1 Multi-Robot Model ..................................... 70
    6.1.2 Formation Building Problem ............................ 71
  6.2 Decentralized Consensus Based Formation Building Algorithm ... 72
  6.3 Decentralized 3D Formation Building with Anonymous Robots .. 79
  6.4 Simulation Results .......................................... 80
  6.5 Summary ........................................................... 84

7 Conclusions and Future Work 87
  7.1 Future Work ...................................................... 88
List of Tables

2.1 Space filling polyhedrons’ volumetric quotient .................. 21
2.2 Comparison of four three-dimensional covering grids ............. 28

4.1 Comparison of different search styles for finding sparsely located
targets ........................................................................... 56

6.1 Simulations Parameters ................................................ 82
List of Figures

2.1 Target region ($\mathbb{M}$) represented by a cube, the set $V$ denoted by . and the set $\tilde{V}$ by $\ast$ ........................................... 17
2.2 Sensing range denoted by $R_s$ and communication range by $R_c$ .... 17
2.3 A truncated octahedron ........................................... 18
2.4 A hexagonal prism ............................................. 18
2.5 A rhombic dodecahedron ........................................ 18
2.6 Four space filling polyhedra with their circumspheres ................. 20
2.7 Consensus to the vertices of the common grid ........................ 25
2.8 Mobile sensors movement to cover the monitoring region (Trunc-
cated octahedron based grid) ........................................ 26
2.9 Intruder detection: the sensing ranges of the mobile sensors shown by syphers and intruders represented by arrows ................. 27
2.10 Mobile sensors movement to form the spherical pattern ............ 29
2.11 Mobile sensors movement to form the cuboid pattern ............... 31
2.12 Mobile sensors movement to form the torus pattern .................. 32
2.13 Mobile sensors movement to form the ellipsoid pattern ............... 33
3.1 Boundaries of $\mathbb{M}$ denoted by , vertices of the grid by o, targets by star ......................................................... 37
3.2 Mobile sensors denoted by sphere (sensing range), targets denoted by stars, detected area shown by yellow tape .................. 38
3.3 Mobile sensors movement to the vertices of the common grid ....... 43
3.4 Mobile sensors movement to cover monitoring region (Truncated octahedron based grid) ........................................ 44
3.5 Searching time versus number of mobile sensors ..................... 46
3.6 Robustness against failure ......................................... 46
4.1 Vertices of the grid denoted by o, mobile sensors search trajectories represented by $-$ and targets by star .......................... 49
4.2 Trajectories of four mobile sensors performing grid-based Levy flight random search for locating clustered targets. mobile sensors’ trajectories denoted by $-$, vertices of the common grid by o, Targets by star ................................................................. 53
4.3 Levy flight without grid search: trajectories of four mobile sensors performing Levy flight search for three targets. mobile sensors’ trajectories denoted by $-$, targets by star .......................... 54
4.4 Trajectories of mobile sensors performing bio-inspired Levy flight random search for locating sparsely located targets. mobile sensors’ trajectories denoted by -, vertices of the common grid by o, targets by star. .............................................................. 54
4.5 Comparison of random search methods using different number of mobile sensors and targets ........................... 55

5.1 A schematic of mobile sensors/targets .......................... 60
5.2 Mobile sensors denoted by sphere (sensing range), moving targets denoted by stars, detected area shown by yellow tape ........ 62
5.3 Boundaries of the search area (M) represented by –, vertices of a covering 3D grid denoted by o ............................ 63
5.4 Trajectories of two mobile sensors performing grid-based random search for detecting two moving targets: Mobile sensors’ trajectories denoted by solid line; moving targets trajectories represented by dashed line, vertices of the grid by o ............................ 65
5.5 Trajectories of two mobile sensors performing Levy walk random search for detecting two moving targets: Mobile sensors’ trajectories denoted by solid line, Mobile sensors by sphere, Mobile targets’ trajectories represented by dashed line, Mobile targets by + ........................................................ 67

6.1 Formation Building Schematic ................................. 71
6.2 Vectors Setting ....................................................... 75
6.3 Vector field around the switching surface .................... 78
6.4 Trajectories of six mobile robotic sensors during formation building ............................... 81
6.5 Trajectories of mobile robotic sensors during 3D formation building ............................... 83
6.6 Convergence of the robot’s headings’ to the same directions .... 84
6.7 Convergence of the mobile robotic sensors to the given configuration ............................... 85
6.8 Formation building with anonymous robots .................. 86
Chapter 1

Introduction

Contents

1.1 Distributed 3D Complete Sensing Coverage and Forming Specific Shapes ........................................... 3
1.2 Distributed 3D Dynamic Search Coverage ............... 6
1.3 Distributed Bio-inspired Random Search for Locating Static Targets .................................................... 7
1.4 Distributed Bio-inspired Algorithm for Search of Moving Targets ....................................................... 9
1.5 Decentralized Three-dimensional Formation Building 10
1.6 Report Contributions ............................................. 11
1.7 Report Organization .............................................. 13

Wireless sensor networks, which are networks without fixed infrastructures, consist of a number of small, low-power, low-cost nodes called sensor nodes which can vary from few to thousands [126]. These nodes have the ability to interact with their environment through sensors, processing information and communicating this information with their neighbors and communicating via wireless channels over a short distance [65]. The wireless sensor networks have the task of monitoring their environment by detecting the event occurring in the target area [131]. Nowadays, sensor and robotic technology are developed enough to inspire the idea of mobile robotic sensor networks. In mobile robotic sensor networks, sensors attached to autonomous underwater robots (AUVs) or unmanned aerial vehicles (UAVs), can move vertically and horizontally in the ocean and air, and each robot can be viewed as a sensing node [174]. Static sensors deployment may not be possible in some applications where the deployment area is not reachable due to the aggressive environment or existence of mines. Moreover, mobile sensor networks are more flexible in terms of deployment and exploration abilities over static sensor networks [82,89,92,190]. Besides, mobile sensor networks can maximize coverage with limited hardware. Also, they reduce the cost of operation, especially in the three-dimensional environments, in that areas of interest
CHAPTER 1. INTRODUCTION

Mobile wireless sensor networks have been used to help or replace human in different applications, for instance, sensing, monitoring [67, 161], surveillance search and rescue [21] operations as well as exploration in hazardous environments. Furthermore, they are used for intrusion detection, oceanographic data collection [175], ocean sampling, pollution monitoring in coastal areas [51], detection of terrorist threats to ships in ports [176], mine reconnaissance [86], disaster prevention like detection of tsunamis and sending warnings [48], ocean resource exploration [2, 124], detection of the origin of the fire [119], its location [38], temperature, and direction also, they can be used for pollution estimation in urban areas [49, 76].

In the literature, most of the works on mobile wireless sensor networks dealt with two-dimensional settings where all sensor nodes are dispersed in a two-dimensional plane. However, in the real world, most of the sensor networks are deployed in 3D spaces such as wireless sensor networks used on the trees of different heights in a forest, or in a building with multiple floors, or underwater applications for ocean sampling and tsunami warnings, aerial defence systems as well as atmosphere pollution monitoring. Deploying sensor networks in 3D environments introduces new challenges in terms of connectivity, coverage, mobility, communication and coordination [56]. Moreover, there are significant physical, technological, and economic differences between terrestrial and air/underwater sensor networks. Compared to those for 2D sensor networks, 3D deployments are more expensive, higher mobility and areas of interest in ocean/air environments are often large [174]. As a result, deploying three-dimensional sensor networks entail extra computational complexity, and many problems cannot be solved by extensions or generalizations of 2D methods [72]. In this report, we study three-dimensional mobile sensor networks to develop new approaches for deployment of mobile sensor networks for coverage, search, and formation building in 3D spaces.

The centralized and decentralized approach are two commonly accepted methods for controlling a network of mobile sensors. The centralized approach is based on the assumption that a central station is available to control a whole group of agents [13, 80]. In contrast, the distributed approach does not require a central station for control. The distributed approach is believed more promising due to many unavoidable physical constraints such as limited resources and energy, short wireless communication ranges, narrow bandwidths, and large sizes of mobile sensors to manage and control [183]. Furthermore, the distributed method has many advantages in realizing cooperative group performances. Also, it is scalable and robust [32].

As a distributed solution to multi-agent coordination, consensus problems was introduced in [90, 167, 169]. In [128, 169], the authors defined consensus as reaching an agreement on a common value of a certain quantity of interest means by a team of agents through negotiating with their neighbours. For more details and progress reports about consensus see survey paper [116, 127] and references therein. In this report, we propose decentralized control algorithm for coverage,
CHAPTER 1. INTRODUCTION

Mobile sensor networks are deployed in an area of interest either deterministically or randomly. The choice of the deployment scheme depends on the type of sensors, applications and the environment that the sensors will operate in [187]. Deterministic node deployment is necessary when sensors are expensive or when their operation is significantly affected by their position or in a controlled and human-friendly environment where the shape of the network, the location of the sensor node is known in advance. In this type of deployment, mobile sensor nodes are deployed in a predefined manner and the placement of sensor nodes for coverage pattern is much easier than random coverage [189]. On the other hand, for many cases, the randomized sensors placement becomes the only feasible option [151]. This is particularly true for the cases where the location and topology change from time to time. Also, for large, dangerous and inaccessible region where human intervention is not possible such as a battle field or a disaster region, intrusion detection, disaster recovery and forest fire detection. In the random deployment scenarios, mobile sensors may be dropped by a low-flying airplane, helicopter or an unmanned aerial or underwater vehicle [177]. As a result, a self-deployment is needed after being distributed in an area of interest, to organize sensor nodes and to place nodes in the appropriate positions [28].

The autonomous deployment of mobile wireless sensor networks is a process in which the mobile sensor nodes adjust their positions dynamically according to a certain algorithm, until the predefined requirement is achieved [103]. In this report, we assume the mobile sensor networks are randomly deployed in a given three-dimensional area. This initial deployment does achieve neither area coverage, nor network connectivity. Thus, we propose distributed self-deployment algorithms to achieve the predefined coverage, search and formation building requirements. Note that the relocation process should take into account the energy cost of mechanical movement and the communication messages involved in directing the motion [187].

The following section presents a literature review on the relevant topics in this report.

1.1 Distributed 3D Complete Sensing Coverage and Forming Specific Shapes

Coverage is an important issue in a sensor network, and is usually treated as a measure of the quality of service. Three types of coverage are defined by Gage: blanket coverage, sweep coverage and barrier coverage [41, 43, 63]. In [44, 46, 137, 154], a distributed law was presented for a group of self-deployed mobile sensors, to perform coverage in a two-dimensional space. A distributed algorithm for blanket coverage of a bounded two-dimensional area was studied in [138] that drives the network of mobile wireless sensors to form a grid consisting of equilateral triangles. All of the mentioned algorithms are based on the two-dimensional space assumption where all mobile sensors are dispersed in a two
CHAPTER 1. INTRODUCTION

dimensional plane [10, 30, 44, 45, 47, 52, 135, 137, 138]. However, in real life, most of the mobile robotic sensor networks are deployed in three-dimensional spaces such as wireless sensor networks deployed on the trees of different heights in a forest, or in a building with multiple floors, or underwater applications for ocean sampling and tsunami warnings, aerial defense systems as well as atmosphere pollution monitoring. For two-dimensional sensor networks coverage means arranging mobile sensors in a line or in a plane, while in three-dimensional cases, in order to detect any penetrating object entering in specific region or gather scientific data from a given three-dimensional space, mobile sensors should cover a three-dimensional space.

Deploying mobile sensor networks in three-dimensional environments introduce new challenges in terms of connectivity, coverage, communication and coordination [56]. Moreover, there are significant physical, technological, and economic differences between terrestrial and air/underwater sensor networks. three-dimensional networks require extra computational complexity, and many problems cannot be solved by extensions or generalizations of 2D methods [72]. Furthermore, the problem of self-deployment of a three-dimensional robotic sensor network is more challenging than a two-dimensional network in the sense that there is not any mathematically proved optimal way to tessellate three-dimensional spaces by the minimum number of robotic sensors [72, 88].

A centralized approximate solution to the problem of assigning sensors to grid positions in three-dimensional environments was presented in [18, 19]. In [11, 12], the authors studied the problem of constructing connected and full covered optimal three-dimensional static wireless sensor networks. In [2], a distributed technique for self-reconfiguration of underwater wireless sensor networks was proposed. However, based on that method, the sensors can only move vertically to adjust depth for maximum coverage. The coverage problem for 3D sensor networks with high deployment density was studied by Watfa and Commuri [179]. Their proposed scheme chooses a subset of sensors to be alive and provide full coverage to the whole 3D region. Their method covers the target area, but the node redundancy is the major flaws of this scheme. In [123], the idea is to adjust the depths of sensors to provide coverage after their initial random deployment at the bottom of the ocean. The deployment is controlled by a central station as a result; the approach is not fully distributed.

Another common method of mobile sensor network deployment for coverage is virtual force algorithm which is based on virtual forces between robots [28, 192]. Using virtual repulsive and attractive forces the mobile robotic sensors are forced to move away or towards each other so that full coverage is achieved. This method has several undesirable limitations. Firstly, using this approach, all robots remain in constant motion and slowly converge to a steady-state. As a result, it depends on mobility, which is a high power consumption task. Moreover, using this method equilibrium steady-state cannot always be predicted, or the system may converge to local-minima [159]. In [20], Bartolini et al. presented the security vulnerabilities of deployment algorithms based on virtual force approach. For more about coverage consider the survey papers and references
CHAPTER 1. INTRODUCTION

therein [59, 173, 178, 191].

In [112], we proposed a decentralized dynamic search coverage algorithm for coverage of 3D spaces. Using that algorithm, a few number of mobile wireless sensors should perform the coverage task for exploring large 3D spaces. As a result, that method is not a good option for the cases where we need to monitor a whole area at the same time or for time-sensitive coverage tasks such as radiation, chemical, and biological agents detection where using that method; the target region is covered only part of the time.

Our coverage problem is to deploy a network of mobile sensors for complete sensing coverage such that every point of a given bounded three-dimensional region is sensed by at least one mobile sensor. In this report, we present a distributed method for self-deployment of a team of mobile robotic sensors for complete sensing coverage in a bounded three-dimensional region. The mobile sensors use the proposed distributed control approach to communicating with their neighboring mobile robotic sensors to coordinate their motions and to archive a common goal.

To cover a bounded three-dimensional area with the minimum number of mobile wireless sensors, we take advantage of vertices of a three-dimensional truncated octahedral grid proposed in [4]. Unlike [4], here we assume that there is not a common coordinate system among all mobile sensors, therefore, we use consensus variables to build a common coordinate system for the mobile wireless sensors.

To verify the effectiveness of the proposed truncated octahedral grid, we compare four different three-dimensional space filling polyhedral grids and we show that a truncated octahedral grid provides better 3D covering grid than other 3D grids because it can cover a bounded 3D area by a fewer number of vertices compare to the other grids. As a result, using this grid can minimize the time of coverage and the number of mobile sensors.

Without connectivity the mobile sensor nodes cannot exchange information with their neighbours. As a result, connectivity is an important issue in wireless sensor networks. A network is considered to be connected if there is, at least, one path between any two sensor nodes in the target area [91]. Here, maintaining connectivity is essential for continuous data gathering from all mobile sensors.

As a result, we propose a decentralized control algorithm to deploy the minimum number of mobile sensors for full coverage and connectivity in 3D areas.

The problem of self-deployment of mobile wireless sensors in three-dimensional spaces for either sampling or sensing coverage of desired phenomena has enormous potential applications. For example, it can be applied to detect intruders entering a protected region. Also, it can be used for environmental monitoring of disposal sites on the deep ocean floor [77], sea floor surveying for hydrocarbon exploration [26], sensing biological and chemical phenomena in different depths, equipment monitoring and leak detection [68]. Furthermore, the results of this part are also applicable to monitoring of any three-dimensional spaces, such as airborne applications, space exploration, greenhouse gas monitoring and storm tracking study [3, 7].
1.2 Distributed 3D Dynamic Search Coverage

The problem of search by mobile wireless sensor networks is an emerging field of research which is very close to the dynamic coverage because in both cases the goal is to monitor a wide area by a few number of mobile sensors [88]. To accomplish the search task especially for exploring large, unknown environments, it is necessary to use effective search algorithms [53]. A majority of existing search methods relies on a single mobile sensor. Using a single mobile sensor for search of large 3D environments is not effective especially for the time sensitive search applications. In contrast, a network of mobile sensors can considerably minimize the search time as well as it improves the robustness of the search process. In recent years, several strategies have been presented for search in the literature. However, most of them are restricted to 2D spaces and not applicable for 3D environments [14, 43]. In [39], the authors investigated two search algorithms which are spiral search and informed random search. Both of the mentioned algorithms are based on 2D assumption also, they are heuristic. Furthermore, they assumed that the search area is known to the robots a priori. In [58], Farmani et al. proposed a multi-target localization and tracking algorithm for multiple UAVs.

Some research focuses on partially unknown environments based on the assumption that the mobile sensors knows the target locations a priori [54,73] but here we assume that the mobile sensor network is not aware of the locations of targets. In [81], a random search algorithm based on Levy flight, for the search of targets in a bounded 3D environment was proposed. While [39] is based on the assumption that the mobile sensor network knows the search environment and the targets’ locations, in [35] it is assumed that the sensor network knows the number of the targets’ but it is unaware of the targets’ locations. Here, we consider the case where the deployed mobile sensor network knows neither targets’ locations nor the number of targets and the targets are randomly placed around the search area. In this report, we introduce a distributed grid-based random algorithm for search in 3D environments by a mobile sensor network where the search environment is unknown to the mobile sensors a priori. To minimize the time of the search the mobile sensors should avoid repeated exploration. As a result, the mobile sensors communicate with their neighbours to share their information while doing search task. Based on this algorithm, each mobile sensor builds a map of the explored area and shares it with other sensors passing within its communication range to minimize the time of the search. The proposed algorithm is based on the decentralized communication method which is an appropriate option for the case where the mobile sensors have limited communication ranges. Here, the mobile sensor network accomplishes the search task using vertices of a cubic and a truncated octahedral grid. The presented search algorithm uses the vertices of the mentioned 3D grids for the search of a given 3D environment. We demonstrate that the truncated octahedral grid provides better 3D covering than a cubic grid, as a result using this grid can minimize the time of the search. A practical application of the proposed search algorithm
is target searching in a three-dimensional aerial and aquatic environment. For example, it can be used for the search of sea mines, sources of pollution, black boxes from downed aircraft.

1.3 Distributed Bio-inspired Random Search for Locating Static Targets

In many coverage applications, it is not necessary to monitor the whole area of interest. Monitoring only some specific points in the area of interest is sufficient and acceptable. For such cases, we assume only a limited number of discrete objects or target points need to be monitored. As a result, full coverage of a given area is not required. In such cases, the deployment cost will decrease because of the smaller number of mobile sensors needed compared to the number required to cover the entire area. The problem of locating targets in both two and three-dimensional spaces is a common task in biological and engineering systems. To do the search task, a team of mobile robots equipped with sensing peripherals capability is deployed in the search area to locate targets like sea mines, black boxes from downed aircraft or ships, hazardous chemicals, fire spots in the jungle or to measure a concentration of dangerous substances [24,114,162]. Finding targets in large environments through the use of a single robot is inefficient, and using a team of mobile robots with local communication capability can succeed more quickly. Moreover, the success of robot teams searching for targets where targets availability is unknown and the condition of the environment is unpredictable depends on the efficiency of the search strategy [61]. In environments where the distribution of targets is unknown, a priori or change over time randomized search strategies are more effective than deterministic search [1,156]. Animals perform foraging activities when searching for food sources in nature with no knowledge of the environment and a limited sensory range. Biological creatures, marine predators, fruit flies, and honey bees who search for sources of food, for a new site and a mate are believed to use the Levy flight random motion [69,171,185]. Levy flight is a random walk mechanism that has the Levy probability distribution function in determining the length of the walk. Levy walk random search approach is easy to implement also it does not require communication between searchers. Using this random search method, a searcher performs the search task by searching a small area of space, and then jump to a zone that is likely previously unexplored. This behaviour is similar to that of a foraging animal who searches for food in a given area with a series of small movements, then travels a larger distance to another area to search again. In fact, by performing Levy flight random walk mechanism, forager optimizes the number of targets encountered versus the travelled distance. The idea is that the probability of returning to the previous site is smaller compared to another random walk mechanism. The process of evolution and natural selection has led animals to optimize their foraging strategies. As a result, researchers in this new emerging area are finding much inspiration from biology.
CHAPTER 1. INTRODUCTION

In [61, 62], the authors proposed a search strategy for a team of mobile robotic sensors based on the T cell movement in lymph nodes and they showed that the distributions of the step-sizes taken by T cells are best described by a random walk with the Levy-like distribution. In [162], a bio-inspired random search strategy for the multi-robot system to efficiently localize targets in underwater search scenarios was proposed. Also, the authors showed that how a novel adaptation strategy based on the firefly optimization algorithm can improve the Levy flight performance particularly when targets are clustered. In [163], the authors combined the Levy flight bio-inspired search algorithm to an artificial potential field to improve robots dispersion in the environment to optimize search task by a team of mobile robots [163]. In [81], a randomized algorithm, based on the Levy flight, for locating sparse targets in a bounded three-dimensional aquatic or air environments environments was proposed. Yang et al. developed a decentralized control algorithm of swarm robot inspired by bacteria chemotaxis for target search and trapping [182].

In all of the mentioned works, mobile robotic sensors search an area of interest using a random walk with a Levy-flight distribution. Most of the previous works were based on the assumption that targets are sparsely distributed in the search areas. For many real-world search applications, targets are clustered in a group such as schools of fish, underwater mines clustered in fields and human victims in disaster areas. Furthermore, some of the previous studies were restricted to two dimensions, and some of them did not consider cooperation and communication amongst mobile robotic sensors [64]. In this report, we propose a novel decentralized grid-based Levy flight random search algorithm. Here, we combine the bio-inspired Levy flight random search mechanism with a 3D covering grid proposed in [112] to optimize the search procedure.

In the former works, the Levy flight random walks took place in a continuous space, but here the random walk occurs on a discrete grid. Unlike [112], in this study the mobile robotic sensors move randomly on the vertices of the covering grid with the length of motion follow a Levy flight distribution. Based on this method, the Levy flight distribution will generate the length of the movement, while the vertices of the covering truncated octahedral grid will improve the dispersion of the deployed mobile robotic sensors. As a result, it optimizes the search task by reducing the search time. To stop the search and reduce the cost of the operation, each mobile sensors communicates to other mobile sensors within its communication range to broadcast information regarding detected targets [107]. Performance of this grid-based random search method is verified by extensive simulations also to evaluate the performance of the proposed grid-based Levy flight algorithm we compare it to other random search strategies for locating sparse and clustered distributions of targets. Furthermore, we give a mathematically rigorous proof of the convergence of the proposed algorithm with probability 1 for any number of mobile robotic sensors and targets.
CHAPTER 1. INTRODUCTION

1.4 Distributed Bio-inspired Algorithm for Search of Moving Targets

The success of a network of mobile robotic sensors searching for targets in an unknown three-dimensional environment depends on the efficiency of the employed search strategy [157]. There are two target search approaches named random search and deterministic search. Random search is just opposite to the determined search in that no predefined information is available about the location of moving targets, their distribution and movement pattern. In environments where the distribution of targets is unknown a priori or changes over time randomized search strategies are more effective [1, 156].

Viswanathan et al. [170] discovered that the foraging behaviour of animals follows a certain random search strategy that is an optimal Markovian search strategy based on Levy processes. Markovian search strategies have many advantages as they are easy to implement and do not require communication between independent searchers. The Levy walk (Levy flight) is the trajectory of the searcher containing piecewise linear pieces of Levy distributed lengths, and has been exhibited in several natural systems. Based on the Levy walk method, a mobile robotic sensor searches a small area of space and then jumps to a region that is not previously explored and starts again. This behavior is similar to that of a foraging animal who searches for food in a given area with a series of small movements, then travels a larger distance to another area to search again. In the bio-inspired Levy flight search approach, the length of the movement can be determined by the distribution of the targets. As a result, this method is flexible and suitable for the scenario where the animal has to adapt to the environmental changes when resources are sparsely distributed within unpredictable environments [88]. The process of evolution and natural selection has led animals to optimize their foraging strategies. As a result, in recent years, many search algorithms have been inspired by the Levy walk random search method [61, 62].

In [162], the authors proposed a bio-inspired random search strategy for the multi-robot system to efficiently localize targets in underwater search scenarios. Also, they showed how a novel adaptation strategy based on the firefly optimization algorithm can improve the Levy flight performance particularly when targets are clustered. In [163], the authors combined the Levy flight bio-inspired search algorithm to an artificial potential field scheme to improve robots dispersion in the environment to optimize search task by a team of mobile robots [163]. In [81], a randomized algorithm, based on the Levy flight, for locating sparse targets in a three-dimensional bounded aquatic or air environments was proposed. In [157], the authors show that using the Levy distribution exhibits advantages over a continuous sweeping search for moving targets in 2D environments. Most of the previous works are restricted to two dimensional spaces, and some of them do not consider cooperation and communication amongst robots [64]. Furthermore, all of the former studies are based on the assumption that targets are stationary. Therefore, their results are not applicable for detecting mobile targets in three dimensional spaces. There are many examples of moving targets such as
a running vehicle or an intruder entering a protected area. In this report, we study the problem of detecting mobile targets moving randomly in a bounded 3D environment by a network of mobile sensors [108].

1.5 Decentralized Three-dimensional Formation Building

The problem of 3D formation building of a network of mobile robotic sensors capable of local communication is to coordinate that network of mobile robots such that they can realize some desired formations and keep it over time [17,37]. A formation of mobile robots can be used to inspect oil pipelines in underwater environments where a team of mobile robots is required to build a desired formation and hover above the pipeline at different depths to monitor pipelines and detect any abnormalities [74]. In such missions, a formation of autonomous robots is preferred over a single robot as it can provide better redundancy and flexibility. Moreover, it can accomplish the monitoring task more rapidly and cost-effectively. Furthermore, a group of mobile robots moving together in a given configuration can form an efficient data acquisition network for gas and oil exploration, environmental monitoring in both aerial and underwater environments, security patrols, mapping of the seabed, search and rescue in hazardous environments [25, 78, 79, 168, 186]. Besides, such formation can be used in airborne applications such as space exploration, storm tracking, greenhouse gas monitoring and air quality monitoring [7].

There are two approaches to the problem of formation control of a group of mobile robots [22, 42, 117]. The centralized approach is based on the assumption that a leader is available to control the whole group of mobile robots. As a result, the leader should have access to all robots information for coordination of their motions and other robots follow the leader. In contrast, the distributed approach does not require a central station for formation building. Decentralized scheme is preferable to the centralized ones for many reasons. First of all, it provides robustness with respect to mobile robots failures also they are scalable. Secondly, in practice, the sensing and communication ranges of autonomous robots are limited. Consequently, mobile robots do not have access to all information of every robot in the workspace but only of the robots within their neighborhood. Therefore, the design of the controller for each robot has to be based on the local information. In recent years, consensus strategies have been applied to achieve the formation control of multiple mobile robots, which focus on driving the kinematics of all mobile robots to a common value. Using this method, a team of mobile robots communicate to their neighbors to reach an agreement on a common value [120].

Decentralized formation control has been studied in a large number of recent publications [55, 75, 87, 120, 133, 144, 145, 149]. In [15, 16, 148], a distributed formation building algorithm for formation building in a given geometric pattern has been proposed. Most of the above-cited papers on formation control are based on the
2D assumption. This assumption is justified for applications where mobile robots are deployed on the earth surface. Therefore, this 2D space hypothesis may no longer be valid in the sea and air where mobile robots are distributed over a 3D space.

Recently, the problem of formation building in a 3D space has been analysed in [27,57,150,188]. However, in most of the previous works the mobile robots were modelled as simple linear models, which have not taken into account constraints on the robot’s linear and angular velocities. Distributed Geodesic Control Laws for flocking of nonholonomic agents in 3D environments was proposed in [105]. By using that algorithm, all mobile robots will finally move in the same direction without any predefined configuration. In [113], we presented preliminary results of decentralized 2D formation building in three-dimensional environments by some simulations. In this note, we extend the results in [113] by extension from a 2D formation building to a 3D formation building in 3D environments. In particular, here we introduce a random formation building algorithm for anonymous robots. In this report, we present a decentralized consensus-based control algorithm for the problem of 3D formation building for nonholonomic mobile robots in 3D environments. Based on the proposed algorithm, the mobile robots only communicate with their neighbors located in their communication range to build and maintain a desired 3D configuration. As a result, this method is a good option for the cases where the mobile robots have limited communication range due to the economic reasons or because of physical constraints (underwater environments). Unlike previous works [130], using this algorithm, the mobile robots move in not only the same direction in 3D spaces but also they finally build a given 3D geometric configuration and move with the same speed. Furthermore, in this study the motion of each robot is modeled by a nonlinear model. Also, we take into account the standard constraints on robots’ angular acceleration and linear velocity [93]. The performance of the proposed three dimensional formation building algorithm is validated by numerical simulations also, we give mathematically rigorous proof of convergence of the proposed algorithms to the given geometric configurations.

Here, we study the problem of formation building for the case when the mobile robots are unaware of their positions in the configuration. We present a decentralized random motion coordination law for a network of mobile robots so that the mobile robots reach consensus on their positions and form a desired three-dimensional geometric pattern from any initial position.

1.6 Report Contributions

The contributions of the report are described as follows:

1. This report introduces a novel distributed control algorithm for self-deployment of mobile robotic sensor networks for complete sensing coverage of bounded three-dimensional spaces. Moreover, it develops a distributed control law for coordination of the mobile robotic sensors such that they form a given 3D shape at vertices of a truncated octahedral grid from any initial po-
CHAPTER 1. INTRODUCTION

Simulation results show that the 3D truncated octahedral grid outperforms other 3D grids in terms of complete sensing coverage time and a minimum number of mobile robotic sensors required to cover the given 3D region. The proposed control algorithms are based on the consensus approach that is simply implemented and computationally effective. The control algorithms are distributed, and the control action of each mobile sensor is based on the local information of its neighbouring mobile sensors. The effectiveness of the proposed control algorithms are confirmed by extensive simulations. Also, we give mathematically rigorous proof of convergence with probability 1 of the proposed algorithms.

2. The report proposes a set of distributed random laws for search coverage in unknown bounded three dimensional environments. The mobile sensors utilize a truncated octahedral grid for the search process. Furthermore, they exchange information regarding detected targets with their neighbouring mobile sensors to minimize the search time. The performance of the proposed algorithm is confirmed by extensive simulations and the convergence with probability 1 for any number of mobile sensors has been mathematically proved.

3. This report presents a novel decentralized bio-inspired grid-based search algorithm to drive a network of mobile sensors for locating clustered and sparsely distributed targets in bounded 3D areas. The proposed method combines the bio-inspired Levy flight random search mechanism for determining the length of the walk with vertices of a covering truncated octahedral grid to optimize the search procedure by improving dispersion of the mobile robotic sensors. The proposed approach has the advantage that it does not need centralized control system also it is scalable. Moreover, this report compares the proposed bio-inspired grid-based random search scheme with other random search methods.

4. This report considers the problem of detecting mobile targets moving randomly in a 3D space by a network of mobile sensors. It proposes a bio-inspired random search mechanism supplemented with a grid-based distribution strategy for the search in bounded 3D areas. Based on this algorithm, the mobile robots do the search task by randomly moving to the vertices of a common 3D covering grid from any initial position. The proposed algorithm uses some simple consensus rules for building a common covering grid. Also, it uses a random walk search pattern drawn by a Levy flight probability distribution to do the search task. The performance of the proposed algorithm is evaluated by simulations and comparison to the Levy walk random search method. Also, we give mathematically rigorous proof of convergence with probability 1 of the proposed algorithm.

5. This report presents a decentralized control law for formation building in three dimensional environments. It uses nonlinear standard kinematics equations with hard constraints on the robot angular and linear velocity to describe the robot motion in 3D spaces. Then, it proposes a random
formation building algorithm for the problem of formation building with anonymous robots in 3D environments. This algorithm is an appropriate option for the case where robots do not know a priori its position in the configuration. The proposed control laws are based on the consensus approach that is simply implemented and computationally effective. The proposed control algorithms are decentralized and the control action of each robot is based on the local information of its neighbouring robots. Using this algorithm, all mobile robots eventually converge to the desired geometric configuration with the same direction and the same speed. The performance of the proposed decentralized control laws is confirmed by extensive simulations. Furthermore, it gives a mathematically rigorous analysis of convergence of the mobile robots to the given configurations.

6. In this report, we have developed an effective, flexible and robust search scheme for target searching tasks, where the environment model and target locations are unknown to the mobile sensor network.

7. Unlike many other studies in this area which present heuristic based strategies, in this report mathematically rigorous analysis is available for the proposed coverage, search and formation building algorithms.

1.7 Report Organization

The rest of this report is organized as follows:

- **Chapter (2)** introduces a distributed random algorithm to drive a network of mobile robotic sensors on the vertices of a truncated octahedral grid for complete sensing coverage of a bounded 3D area. Moreover, it provides a decentralized algorithm for self-deployment of mobile robotic sensors to form a desired 3D geometric shape on the vertices of a truncated octahedral grid.

- **Chapter (3)** presents a distributed random algorithm for search in bounded three dimensional environments by a network of mobile sensors. The proposed algorithm utilizes an optimal three dimensional grid pattern for the search task.

- **Chapter (4)** extends the distributed algorithm proposed in the previous chapters by introducing a novel bio-inspired decentralized grid-based random search algorithm for finding randomly located objects in 3D areas by a mobile sensor network.

- **Chapter (5)** presents a distributed bio-inspired Levy flight algorithm for search of moving targets in three dimensional spaces. Using this algorithm, the mobile robots do the search task by randomly moving to the vertices of a common 3D covering grid from any initial position.
- Chapter (6) proposes a decentralized consensus-based control law for a team of mobile robots which result in forming a given geometric configuration from any initial positions in 3D environments. Then, it presents a decentralized random motion coordination law for the multi-robot system for the case when the mobile robots are unaware of their positions in the configuration in three dimensional environments.

- Chapter (7) summarizes the work presented in this report and gives directions for the future research.
Chapter 2

Distributed Complete Sensing Coverage and Forming Specific Shapes

Contents

2.1 Problem Statement .................................................. 16
2.2 Decentralized Algorithm for Self deployment of Mobile sensors for Complete coverage .................. 22
  2.2.1 Simulation Results ............................................. 24
2.3 Self-deployment in a Given Three-dimensional Formation ......................................................... 28
  2.3.1 Simulation Results ............................................. 30
2.4 Summary ............................................................... 33

In this chapter, we present a decentralized control algorithm for self-deployment of mobile sensors for complete sensing coverage in a bounded three-dimensional region. The proposed decentralized control scheme uses distributed communication and control mechanisms for the cooperation of the mobile sensors. Based on this method, the mobile sensors exchange information using a wireless network [111]. In recent years, distributed coordination of networks of mobile robotic sensors has attracted many researchers because based on this approach sensor networks are not required to access information from all other agents also the computational complexity of decentralized systems is independent of the number of agents in the network. Furthermore, this approach is not sensitive to the loss of central leaders, and it scales well with increasing the number of mobile sensors. Moreover, it is flexible, robust and reliable. Besides, the decentralized control method is a good option for the cases where the mobile wireless sensors have limited communication ranges due to the economic reasons or because of physical constraints (underwater environments). Based on this scheme, the mobile sensors communicate with their neighboring mobile sensors to coordinate their motions and to archive a common goal.
To cover a bounded three-dimensional area with the minimum number of mobile wireless sensors, we take advantage of vertices of a three-dimensional truncated octahedral grid proposed in [4]. Unlike [4], here we assume that there is not a common coordinate system among all mobile sensors, therefore we use consensus variables to build a common coordinate system for the mobile sensor network. To verify the effectiveness of the proposed truncated octahedral grid, we compare four different three dimensional space filling polyhedral grids and we show that the truncated octahedral grid provides better covering grid than other three-dimensional grids because it can cover a bounded three-dimensional area by the fewer number of vertices compare to the other grids. As a result, using this grid we can cover the target area quicker and by less number of mobile sensors compare to the other grids.

At first part of this chapter, we introduce a distributed random algorithm for complete coverage of an unknown bounded three-dimensional area. At second part, we develop a distributed control law for coordination of motion of the mobile sensors such that they form a given three-dimensional shape at vertices of a truncated octahedral grid from any initial positions. The performance of the proposed algorithms are verified by simulations also we give a mathematically rigorous proof of the convergence of the proposed algorithm with probability 1 for any number of mobile sensors and any sensor initial conditions [109].

The rest of this chapter is organized as follows: The problem is formulated in Section 2.1. In Section 2.2, a distributed random algorithm for self-deployment of a mobile robotic sensor network on vertices of a truncated octahedral grid in a bounded region is presented. In Section 2.3, the results of section 2.2 are developed for self-deployment of mobile wireless sensors in a given three-dimensional shape. Numerical simulation results that validate the proposed algorithms are presented in Section 2.2.1, Section 2.3.1. Conclusion is summarized in Section 2.4.

2.1 Problem Statement

Our coverage problem is to deploy a network of mobile sensors for complete sensing coverage such that every point inside a given bounded three-dimensional region is sensed by at least one mobile sensor. The objective of this chapter is to design a distributed random control algorithm to drive a team of mobile sensors for complete sensing coverage of a given bounded three-dimensional area. The three-dimensional bounded regions \( M \subset \mathbb{R}^3 \) to be covered is shown in Fig.2.1. We assume the mobile wireless sensor network consisting of \( n \) mobile sensors. Let \( p_i(.) \in \mathbb{R}^3 \) be the Cartesian coordinates of the sensor \( i \). We assume every mobile sensor has a limited sensing range, denoted by \( R_s \). Sensing is omnidirectional and the sensing region of each mobile sensor can be represented by a sphere of radius \( R_s \), having the mobile sensor at its centre, see Fig.2.2.

Assumption 2.1.1. We assume a spherical sensing model such that each mobile sensor has a sensing range of \( R_s > 0 \) can reliably detect any object located within
Figure 2.1: Target region $(M)$ represented by a cube, the set $V$ denoted by . and the set $\hat{V}$ by *.

Figure 2.2: Sensing range denoted by $R_s$ and communication range by $R_c$. 
(a) Figure 2.3: A truncated octahedron.

(b) Figure 2.4: A hexagonal prism

(a) Figure 2.5: A rhombic dodecahedron

(b) a distance of $R_s$ from the sensor. In other words, mobile sensor $i$ has the ability
to identify objects in a sphere of radius $R_s$ defined by:

$$ S_{i,R_s} = \{ p \in \mathbb{R}^3; \| (p - p_i(K)) \| \leq R_s \} $$

(2.1)

We assume every mobile sensor has a limited communication range, denoted by $R_c$. Communication is omnidirectional, and the communication region of each mobile sensor is represented by a sphere of radius $R_c$, having the mobile sensor at its centre, see Fig.2.2.

**Assumption 2.1.2.** We assume a spherical communication model where each mobile sensor has a communication range of $R_c > 0$ can reliably communicate with any mobile sensors located within a distance of $R_c$ from the mobile sensor as shown in Fig.2.2. Mathematically speaking, mobile sensor $i$ can obtain information on its neighbours in a sphere of radius $R_c$ defined by:

$$ S_{i,R_c} = \{ p \in \mathbb{R}^3; \| (p - p_i(K)) \| \leq R_c \} $$

(2.2)

**Definition 2.1.1.** The mobile sensors are called homogenous if they have the identical sensing ability, computational ability, and the capacity to communicate.

Here, we assume the mobile sensors are homogenous. Also, we assume that the mobile sensors can detect the boundaries of the region $M$. For a given time $K > 0$, each mobile sensor communicates with its surrounding neighbours in the communication range at the discrete time instance $K = 0, 1, 2, ..$ for the coordination of their motions [84].

**Definition 2.1.2.** Space-filling polyhedron is a polyhedron that can be used to fill a three-dimensional space without overlaps or gaps.

The figures of three space-filling polyhedral are shown in Fig.2.3, Fig.2.4 and Fig.2.5. As shown in Fig.2.3(b), a truncated octahedron has 14 faces (6 squares and 8 regular hexagonal), and the length of the edges of the squares and the hexagons are the same. Fig.2.4(b) shows that a hexagonal prism is a prism composed of six rectangular sides and two hexagonal bases. Moreover, as shown in Fig.2.5(b), a rhombic dodecahedron is a convex polyhedron with 12 congruent rhombic faces. As mentioned before, we assumed that the mobile sensors’ communication and sensing models are spherical also it is clear that spheres do not tessellate three-dimensional spaces without overlap or gap. Therefore, we introduce a space-filling polyhedron that best approximates a sphere such that if each mobile sensor is modelled by that polyhedron, then the number of mobile sensors required to cover a three-dimensional volume is minimized. Fig.2.6(a), Fig.2.6(b), Fig.2.6(c) and Fig.2.6(d) show four space filling polyhedra and their circumspHER. Finding such optimal space filling polyhedrons is still an important open mathematical problem. This problem is similar to the century old Lord Kelvin problem [166]. In 1887, Kelvin proposed the truncated octahedrons as the solution to the problem of finding a space-filling arrangement of similar cells of equal volume with minimal surface area. Different space filling polyhedrons
can be compared using a criterion called volumetric quotient. The volumetric quotient is defined as [5]:

$$V_q = \frac{V_p}{V_s}$$  \hfill (2.3)

Where $V_p$ is the volume of a polyhedron and $V_s$ is the volume of its circumsphere. The volumetric quotient of a truncated octahedron is bigger than a cube, a hexagonal prism and a rhombic dodecahedron as shown in Table.2.1. As a result, we expect that using a truncated octahedral grid for coverage leads to a fewer number of vertices to be covered and minimizes the time of coverage and the number of mobile sensors for covering a three-dimensional space [6]. In the
CHAPTER 2. DISTRIBUTED COMPLETE SENSING COVERAGE AND FORMING SPECIFIC SHAPES

Table 2.1: Space filling polyhedrons’ volumetric quotient

| Space filling polyhedron       | Volumetric quotient |
|--------------------------------|---------------------|
| Truncated octahedron           | 0.68                |
| Cube                           | 0.36                |
| Hexagonal prism                | 0.47                |
| Rhombic dodecahedron           | 0.47                |

following, we assume that vertices of the truncated octahedral grid are the centres of truncated octahedrons of that grid.

Definition 2.1.3. Consider a truncated octahedral grid cutting $M$ into equal truncated octahedrons with the sides of $2R_s\sqrt{10}$. Let $V$ be the infinite set of centres of all the truncated octahedrons of this grid. The set $\hat{V} = V \cap M$ is called a truncated octahedral covering set of $M$ (Fig. 2.1).

Definition 2.1.4. A sensor network is said to be connected if any sensor node can communicate with other sensor nodes directly or indirectly.

Connectivity is a fundamental issue in wireless sensor networks because without connectivity the mobile sensor nodes cannot exchange information with their neighbours. For our coverage algorithm, connectivity is crucial as without connectivity the mobile sensors cannot reach consensus on the vertices of the common grid. As a result, they cannot build a common covering grid. For the truncated octahedron grid, the communication range must be at least 1.78 times the sensing range to maintain connectivity among nodes [6]. Mathematically speaking:

$$\frac{R_c}{R_s} \geq \frac{4}{\sqrt{5}}. \quad (2.4)$$

It is worth to note that for the cases where the communication range is between 1.41 and 1.78 times the sensing range, then a hexagonal prism placement strategy or a rhombic dodecahedron placement strategy should be used to maintain connectivity in the network.

In this chapter, we present a decentralized control law to drive the mobile sensors to the vertices of the covering truncated octahedral grid. The aim of this algorithm is to deploy the minimum number of mobile wireless sensors such that any point inside the bounded three-dimensional space is within the sensing range of at least one mobile sensor. In other words:

$$\forall \ p \in M, \ \exists \ v \in \hat{V} : \|p - v\| \leq R_s \quad (2.5)$$

It means that the monitoring region is covered completely if for any point inside the monitoring region $M$ there exists a mobile sensor in the sensing range of that point. Also, all mobile sensor can communicate with each other. First, we introduce a distributed random algorithm to drive the mobile sensors to the
vertices of a truncated octahedral grid for complete coverage of an unknown bounded 3D area. Then, we propose a three-stage algorithm for coordination of motion of the mobile sensors, so that drive the mobile sensors on the vertices of a common truncated octahedral grid to form a desired 3D geometric shapes from any initial positions.

Let \( N_i(K) \) be the set of all mobile sensors \( j \neq i \) and \( j \in 1, 2, ..., n \) that at time \( K \) belong to the sphere \( S_{i,R_c(K)} \) also it is supposed that mobile sensor \( i \) has \( |N_i(K)| \) number of neighbours at time \( K \). We employ the notion of graph to describe the relationships between neighbours, because it can change over time. For any time \( K > 0 \), the relationships between neighbours are described by a simple undirected graph \( G(K) \) with vertex set \( 1, 2, ..., n \) where \( i \) corresponds to sensor \( i \). The vertices \( i \) and \( j \) of the graph \( G(K) \), where \( j \neq i \), are connected by an edge if and only if, the mobile wireless sensors \( i \) and \( j \) are neighbours at time \( K \). Here, we impose the following condition on the connectivity of the graph [75, 138].

**Assumption 2.1.3.** There exists an infinite sequence of contiguous, non-empty, bounded time intervals \( [K_j, K_{j+1}) \), \( j = 0, 1, 2, ... \) starting at \( K_0 = 0 \) such that across each \( [K_j, K_{j+1}) \), the union of the collection \( \{G(K) : K \in [K_j, K_{j+1})\} \) is a connected graph [138].

### 2.2 Decentralized Algorithm for Self deployment of Mobile sensors for Complete coverage

In this section, we present an algorithm for self deployment of the mobile sensor network for complete sensing coverage of a bounded three-dimensional area. We use a node placement strategy proposed in [4] to build a covering truncated octahedral grid’s vertices and to find the locations where the mobile sensors should be placed.

\[
V_1 = (x + (2\alpha_1 + \alpha_3)\frac{2R_s}{\sqrt{5}}, y + (2\alpha_2 + \alpha_3)\frac{2R_s}{\sqrt{5}}, z + \alpha_3\frac{2R_s}{\sqrt{5}}) \quad (2.6)
\]

Here, \( \alpha_1, \alpha_2 \) and \( \alpha_3 \in \mathbb{Z} \) and \( \mathbb{Z} \) is the set of all integers. The distance between two neighbouring nodes is \( \frac{2R_s}{\sqrt{5}} \). The inputs to our algorithm are sensing range \( R_s \) and the coordinates of the point \((X, Y, Z)\), which act as a seed for the growing the truncated octahedral grid. We use consensus variables to build a common coordinates \((X, Y, Z)\) for all mobile sensors. As a result, we will build a common truncated octahedral covering grid for all mobile sensors without the need to know a common point a priori. Initially, the mobile sensors do not have a common coordinate system. Therefore, we assume each mobile sensor has consensus variables \( X_i(K), Y_i(K) \) and \( Z_i(K) \) in its own coordinate system. Here, nodes (the mobile sensors) can reach a consensus on the frame of reference \((X, Y, Z)\) by local estimation of distances to neighbours.

We suppose that, for any mobile sensors, the vector \( V_1 = (x + (2\alpha_1 + \alpha_3)\frac{2R_s}{\sqrt{5}}, y + (2\alpha_2 + \alpha_3)\frac{2R_s}{\sqrt{5}}, z + \alpha_3\frac{2R_s}{\sqrt{5}}) \) determines the location where nodes should be placed.
It is obvious that any truncated octahedral grid is uniquely defined by a point \( q_i(K) = (X_i(K), Y_i(K), Z_i(K)) \) and \( R_s \). Thus, any \( q_i(K) = (X_i(K), Y_i(K), Z_i(K)) \) and \( R_s \) uniquely define a truncated octahedral covering set of \( M \), which will be denoted as \( \hat{V}[q, R_s] \). The scalar parameters \( \alpha_1, \alpha_2, \alpha_3 \) and \( R_s \) along with the three-dimensional consensus variables \( q_i(K) = [X_i(K), Y_i(K), Z_i(K)] \) characterize the coordinates of a vertex of the grid. The mobile sensors will start with different values of the coordination variables \( X_i(0), Y_i(0) \) and \( Z_i(0) \), then eventually converge to some consensus values \( X_0, Y_0 \) and \( Z_0 \) which define a common coordinate system for all mobile sensors.

Let \( p \) be a mobile sensor position, and \( C[q](p) \) defines the closest vertices of the truncated octahedral grid \( \hat{V}[q, R_s] \) to \( p \). The following rules for updating the consensus variables and the mobile sensors coordinates are proposed:

\[
X_i(K + 1) = \frac{X_i(K) + \sum_{j \in N_i(K)} X_j(K)}{1 + |N_i(K)|} \tag{2.7}
\]

\[
Y_i(K + 1) = \frac{Y_i(K) + \sum_{j \in N_i(K)} Y_j(K)}{1 + |N_i(K)|}
\]

\[
Z_i(K + 1) = \frac{Z_i(K) + \sum_{j \in N_i(K)} Z_j(K)}{1 + |N_i(K)|}
\]

\[
p_i(K + 1) = C[q_i(K), R_s](p_i(K)) \tag{2.8}
\]

The rules (2.7) and (2.8) are local rules for updating the consensus variables of each mobile wireless sensor based on the average of its own and the consensus variables values of its neighbours. In other words, mobile sensors use the consensus variables to achieve the consensus on the vertices of the truncated octahedral grid. Finally, the mobile wireless sensors converge to the vertices of a truncated octahedral grid.

**Theorem 2.2.1.** : Suppose that assumption 2.1.3 hold and the mobile sensors move according to the distributed control algorithm (2.7), (2.8). Then, there exists a truncated octahedral grid \( \hat{V} \) such that for any of the mobile wireless sensors there exists a \( \hat{v} \in \hat{V} \) such that:

\[
\lim_{K \to \infty} p_i(K) = \hat{v} \tag{2.9}
\]

Based on the law (2.7) and the assumption 2.1.3, the consensus variables converges to some constant values as follows:

\( X_i(K) \to X_0, Y_i(K) \to Y_0 \) and \( Z_i(K) \to Z_0 \).

(See [36, 75] for the proof of convergence of consensus variables to some constant values). Since \( R_s, \alpha_1, \alpha_2 \) and \( \alpha_3 \) are common to all mobile sensors, therefore:

\[
(x_i + (2\alpha_1 + \alpha_3)\frac{2R_s}{\sqrt{5}}, y_i + (2\alpha_2 + \alpha_3)\frac{2R_s}{\sqrt{5}}, z_i + \alpha_3\frac{2R_s}{\sqrt{5}}) \to
\]
(x_0 + (2\alpha_1 + \alpha_3)\frac{2R_s}{\sqrt{5}}, y_0 + (2\alpha_2 + \alpha_3)\frac{2R_s}{\sqrt{5}}, z_0 + \alpha_3\frac{2R_s}{\sqrt{5}}).

(2.10)

It means that, after convergence of the consensus variables to some constant values, all mobile sensors will have a common truncated octahedral grid. Then, based on the rule (2.8), all mobile sensors move to the vertices of the common truncated octahedral grid. As a result, at the end of this stage, all mobile sensors are derived to some vertices of the truncated octahedral grid. This completes the proof of Theorem 2.2.1.

By using this algorithm, many of the vertices of the truncated octahedral grid remain unoccupied. To occupy all vacant vertices of the grid, we use the following random algorithm for spreading mobile sensors in the whole area [138]

\[
p((K + 1), i) = \begin{cases} 
  s & \text{with probability } \frac{1}{|\mathcal{S}(p_i(K))|} \\
  \forall s \in \mathcal{S}(p_i(K))
\end{cases}
\]

(2.11)

Where \( \mathcal{S}(p_i(K)) \) is defined as a set consisting of \( V = p_i(K) \) and all its unoccupied neighbours belong to \( \hat{V} \). Furthermore, let \( |\mathcal{S}(p_i(K))| \) represents the number of elements in \( \mathcal{S}(p_i(K)) \). It is clear that \( 1 \leq |\mathcal{S}(p_i(K))| \leq 27 \).

The relation (2.11) implies that the mobile wireless sensors move randomly to occupy unoccupied vertices of the truncated octahedral grid. This continuous until all vertices of the grid are occupied. It is obvious that all unoccupied vertices will becomes covered if the number of mobile sensors be more than the number of vertices of the covering truncated octahedral grid.

**Theorem 2.2.2.** Suppose \( N_m \) and \( N_\hat{v} \) be the number of mobile sensors and the number of vertices in \( \hat{V} \), respectively and \( N_m \geq N_\hat{v} \). Also, assume the mobile sensors move according to the law (2.11). Then with probability 1 there exists a time \( K_0 \geq 0 \) such that for any \( \hat{v} \in \hat{V} \) the relationship \( \hat{v} = p_i(K) \) holds for some \( i = 1, 2, ..., n \) and all \( K \geq K_0 \).

Here, the random movement of the mobile sensors is Markov process as the mobile sensors follow the transition rule proposed in (2.11) that only depends on the current state of the mobile sensors, does not depend on their history. Moreover, the proposed control law defines an absorbing Markov chain. An absorbing Markov chain is a Markov chain in which every state can reach an absorbing state. An absorbing state is a state that, once entered, cannot be left. In this case, the state when different mobile sensors are located in different vertices of the covering grid is an absorbing state. These absorbing states can be reached from any initial state with a non-zero probability. This implies that with probability 1, one of the absorbing states will be achieved. This completes the proof of Theorem 2.2.2.

### 2.2.1 Simulation Results

In this section, we present some examples of computer simulations to illustrate the proposed decentralized coverage algorithm. The simulations were carried
(a) Mobile sensors' initial deployment: mobile sensors denoted by *, monitoring region represented by a cube

(b) Mobile sensors movement to the vertices of the Truncated octahedron based grid

Figure 2.7: Consensus to the vertices of the common grid
Figure 2.8: Mobile sensors movement to cover the monitoring region (Truncated octahedron based grid)
CHAPTER 2. DISTRIBUTED COMPLETE SENSING COVERAGE AND FORMING SPECIFIC SHAPES

Figure 2.9: Intruder detection: the sensing ranges of the mobile sensors shown by syphers and intruders represented by arrows

out using MATLAB R2014b. In the first set of simulations, our objective is to obtain complete sensing coverage within a bounded three-dimensional space ($M$). Fig.2.7-Fig.2.8 illustrate the evolution of the mobile wireless sensors’ positions over time. We assume that 100 mobile wireless sensors are spread randomly around an area of interest as shown in Fig.2.7(a), represented by *. For the initial configuration, it is assumed that the locations of these mobile sensors are randomly and independently distributed in a cube with a side of $10R_s$. It is clear that this initial random deployment doesn’t achieve optimal coverage of the area of interest. Therefore, the mobile sensors use our proposed deployment algorithm to ensure the area of interest is covered, and the mobile sensor network is connected.

At first stage, the mobile wireless sensors move to the monitoring region according to the algorithms (2.7),(2.8) to form a covering grid as shown in Fig.2.7(b). At the end of this stage, all mobile wireless sensors are at the vertices of the truncated octahedral grid. Next, the random algorithm (2.11) has been applied to spread mobile wireless sensors to obtain complete sensing coverage. Fig.2.8(a) shows the mobile wireless sensors locations after 7 steps. Finally, Fig.2.8(b) represents the sensors location after 17 steps when all vertices of the truncated octahedral grid have been occupied, and the region $M$ has been completely covered by the mobile wireless sensors.

Next, we consider four intruders are moving to the target area and employ them to evaluate the performance the proposed sensing coverage algorithm. Fig.2.9 shows intruders’ position at the initial deployment and the arrows indicate their headings. Our simulation demonstrates the deployed mobile sensor network have
Table 2.2: Comparison of four three-dimensional covering grids

| Space filling polyhedron       | Number of vertices | Complete sensing coverage (steps) |
|--------------------------------|--------------------|----------------------------------|
| Truncated octahedron           | 100                | 17                               |
| Cube                           | 172                | 28                               |
| Hexagonal prism                 | 140                | 23                               |
| Rhombic dodecahedron           | 142                | 22                               |

detected the intruder after entering the target region.

In the following, we carried out several simulations to compare the performance of four space filling polyhedra grids (Truncated octahedron, Hexagonal prism, Cube and Rhombic dodecahedron) for complete sensing coverage of the same area. As mentioned before, we expected that a polyhedron with the higher volumetric quotient needs, the smaller number of vertices to cover a three-dimensional space. The volumetric quotient of a truncated octahedron is bigger than aforementioned space filling polyhedrons. It means that the number of mobile sensors require to cover a three-dimensional area by a truncated octahedral grid is less than the number of mobile sensors required by other grids.

Table 2.2 shows the average complete sensing coverage steps, and the number of mobile wireless sensors need for full sensing coverage of the same area using Truncated octahedral, Hexagonal prism, Cubic, and Rhombic dodecahedron grids. As shown in table.2.2, the average coverage time (step) of the truncated octahedral grid is less than that of the other three-dimensional grids. Because the number of vertices to be covered is less than other grids. Moreover, truncated octahedral grid has the smaller overlapping area, hence this grid requires the less number of mobile sensors for covering the same area. In conclusion, simulation results show that the truncated octahedron grid is the best among the three kinds of grids as it has the smallest overlapping area. Hexagonal prism and Rhombic dodecahedron provide relatively good performance while the cubic grid is the worst among all since it has the biggest overlapping area.

2.3 Self-deployment in a Given Three-dimensional Formation

In this section, we develop a randomized distributed control law for the mobile sensor network which results in forming a given geometric formation on vertices of a truncated octahedral grid. Using this algorithm, the mobile sensors coordinate their motions to achieve a desired geometric pattern. To do that, we apply the control law (2.7),(2.8) which drive all mobile wireless sensor on vertices of a common covering truncated octahedral grid. As mentioned before, after applying
Figure 2.10: mobile sensors movement to form the spherical pattern

control law (2.7), (2.8) the mobile sensors converge to some consensus values \((x_0, y_0, z_0)\) which define the origin of the common coordinate system for the mobile sensor network. We introduce the following rules for the mobile wireless sensors to achieve a consensus on the directions of the common coordinate system as follows:

\[
\theta_i(K + 1) = \frac{\theta_i(K) + \sum_{j \in N_j(K)} \theta_j(K)}{1 + |N_j(K)|} \tag{2.12}
\]
\[
\psi_i(K + 1) = \frac{\psi_i(K) + \sum_{j \in N_i(K)} \psi_j(K)}{1 + |N_j(K)|}
\]

Where \(\theta_i(t)\) and \(\psi_i(t)\) denote the pitch and the yaw angles of the mobile wireless sensors, respectively. Based on this control law, the mobile wireless sensors start with random initial values for \(\theta_i(k)\) and \(\psi_i(k)\) and eventually converge to some constant values \(\theta_0\) and \(\psi_0\). It is obvious that the consensus values \(x_0, y_0, z_0, \theta_0\) and \(\psi_0\) define a specific orthogonal coordinate system \((x,y,z)\) on the three-dimensional space. Using this three dimensional coordinate system, we can define three dimensional geometric patterns by some relationship of the type \(f(x,y,z) \in \mathbb{R}^3\) as follows:

1. Sphere: \(x^2 + y^2 + z^2 < r\);
2. Cuboid: \(c_{x1} < x < c_{x2}, c_{y1} < y < c_{y2}, c_{z1} < z < c_{z2}\);
3. Torus: \(x = (c + a \cos(\theta)) \cos(\varphi), y = (c + a \cos(\theta)) \sin(\varphi), z = \sin(\theta)\).

Where \(a, c \in \mathbb{R}\) and \(\theta, \varphi \in [0, 2\pi]\);
4. Ellipsoid: \(\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1\). Where \(a, b, c \in \mathbb{R}\).

Note that; it is assumed that the mobile wireless sensor network knows the following desired three-dimensional geometric patterns a priori. Now, we introduce a three-stage algorithm for the mobile wireless sensor network so that drive the mobile sensors on the vertices of a truncated octahedral grid to form the desired three-dimensional geometric shapes as follows:

(St1): At first stage, we use control law (2.7), (2.8), (2.12) to drive the mobile sensors at vertices of a common truncated octahedral grid and to build a common coordinate system for all mobile sensors.

(St2): At second stage of the algorithm, the mobile sensors move to the closest vertices of three dimensional truncated octahedral grid located inside the given geometric shape.

(St3): To form the desired geometric pattern, at third stage we apply the control law (2.11) to randomly drive the mobile wireless sensors to cover the given three-dimensional pattern.

**Theorem 2.3.1.** Suppose that assumption 2.1.3 hold and the desired three dimensional geometric configuration is given by \(f(x,y,z) \in \mathbb{R}^3\). After applying the three stage algorithm St1, St2 and St3, with probability 1 there exists a time \(k_0 \geq 0\) such that for any \(i = 1, 2, ..., n\) there exists a vertex \(v\) of the truncated octahedron grid set belonging to the \(f(x,y,z)\) such that \(v = p_i(k)\) for all \(k \geq k_0\).

Theorem 2.3.1 immediately follows from Theorem 2.2.1.

### 2.3.1 Simulation Results

In this section, we illustrate simulation results for the problem of driving the mobile sensors to form a specific geometric pattern in three-dimensional spaces. We suppose that 80 mobile wireless sensors are distributed randomly in a three-dimensional area. First, we consider a spherical region defined by \(x^2 + y^2 + z^2 < r\). Based on St1 the mobile sensors move according to the control law (2.7),(2.8)
Figure 2.11: Mobile sensors movement to form the cuboid pattern

to the vertices of a common truncated octahedral grid as shown in Fig.2.10(a). Then, all mobile sensors located outside of the given sphere move to the inside of the given sphere as shown in Fig.2.10(b). Fig.2.10(c) shows the sensors locations after 5 steps. The final formation of the mobile sensors is shown in Fig.2.10(d) when all mobile sensors have spread inside the given sphere on different vertices of the truncated octahedral grid. At the next stage, the mobile sensors form a cuboid formation defined by $c_{x_1} < x < c_{x_2}, c_{y_1} < y < c_{y_2}, c_{z_1} < z < c_{z_2}$. Fig.2.11(a)- Fig.2.11(d) shows different stages of the formation building inside the cuboid region.
The next set of simulations are carried out to show the effectiveness of the proposed algorithm to form a complicated torus and ellipsoid patterns shown in (Fig.2.12(a)- Fig.2.12(d)) and (Fig.2.13(a)- Fig.2.13(d)), respectively. As shown in the simulations, for all of the given geometric patterns the mobile wireless sensors build a covering truncated octahedral grid after few steps and then they converge to the vertices of the truncated octahedral grid. After that, the mobile wireless sensors move inside the given pattern on the vertices of the 3D grid and finally they spread inside the given geometric formation and build the given pattern. As a result, the presented simulation results verify the effec-
CHAPTER 2. DISTRIBUTED COMPLETE SENSING COVERAGE AND FORMING SPECIFIC SHAPES

Figure 2.13: Mobile sensors movement to form the ellipsoid pattern

(a) Sensors denoted by *, vertices of the grid represented by o, the desired formation given by the ellipsoid
(b) Sensors moved inside the ellipsoid
(c) Sensors' movement to cover vacant vertices inside the ellipsoid
(d) complete sensing coverage

tiveness of the proposed distributed formation building control law for various given patterns.

2.4 Summary

In this chapter, a distributed control algorithm for self-deployment of a mobile sensor network for complete sensing coverage of a bounded three-dimensional space was presented. The proposed control laws drive the network of mobile
wireless sensors to form a three-dimensional covering grid in the target region then they occupy the vertices of the grid. Also, we developed a distributed control law for coordination of the mobile sensors such that they form a given three-dimensional shape at vertices of a truncated octahedral grid from any initial positions. Simulation results showed that the three-dimensional truncated octahedral grid outperforms other three-dimensional grids in terms of complete sensing coverage time and the minimum number of mobile sensors required to cover completely the given three-dimensional region.

The presented control algorithms are based on the consensus approach that is simply implemented and computationally effective. The control algorithms are distributed and the control action of each mobile wireless sensor is based on the local information of its neighbouring mobile wireless sensors. Therefore, these algorithms could be applicable to the cases where the communication range is limited. The effectiveness of the proposed control algorithms were confirmed by simulations. Also, we give mathematically rigorous proof of convergence with probability 1 of the proposed algorithms.
Chapter 3

Distributed 3D Dynamic Search Coverage

Contents

3.1 Problem Formulation ........................................... 36
3.2 Decentralized Random Search Algorithm .......... 39
  3.2.1 Building a Common Three-dimensional Covering Grid 39
  3.2.2 Randomized Grid Search ................................. 41
3.3 Simulation Results ............................................... 42
3.4 Summary .......................................................... 47

In the previous chapter, it was assumed that we have enough mobile sensors to deploy for the full sensing coverage in a 3D space. In this chapter, we assume the number of mobile sensors is limited. Here, we introduce a distributed grid-based random algorithm for search in 3D environments, where the search environment is unknown to the mobile sensors a priori [112]. Mobile sensors normally rely on absolute positioning or localisation to perform the search task. Absolute positioning needs GPS signal which is usually unfeasible inside or in underwater due to attenuated signals [9]. Robot localization commonly entails odometer sensing and environment maps. Such approaches are not applicable on the low cost large swarms as environment maps may be unknown a priori [33,66]. To do the task of localization and positioning without global information, we employ the concept of distributed processing of local information through a wireless communication network [159]. Our proposed algorithm relies on local sensing and communication as it uses the mobile robotic sensors themselves as sensor nodes. Therefore, this method does not require absolute positioning and localization system. Based on this approach, the mobile sensors communicate with their neighbours to share their information while doing search task. As mentioned before, the proposed algorithm relies on the decentralized communication method which is a real option for the case where the mobile sensors have limited communication ranges. To minimize the time of search and to avoid repeated exploration, each mobile
sensor builds a map of the explored area and shares it with other sensors passing within its communication range. As a result, using this strategy the mobile sensors do the search task cooperatively and time efficiently [112].

In this chapter, the mobile sensor network accomplishes the search task based on a cubic and a truncated octahedral grid. The presented search algorithm uses the vertices of the mentioned 3D grids for the search of a given 3D environment. We demonstrate that the truncated octahedral grid provides better 3D covering than a cubic grid, as a result using this grid can minimize the time of the search. Moreover, simulation results show that the average search time by the proposed grid based random search method is less than the Levy flight random algorithm. Also, we give a mathematically rigorous proof of the convergence of the proposed algorithm with probability 1 for any number of sensors and any sensor initial conditions. A practical application of the proposed search algorithm is target searching in a three-dimensional aerial and aquatic environment. For example, it can be used for the search of sea mines, sources of pollution and black boxes from downed aircraft.

3.1 Problem Formulation

In this chapter, our objective is to design a distributed random search algorithm to drive a network of mobile sensors for search in bounded 3D spaces. We assume the sensor network has no information about the search area ($M \subset \mathbb{R}^3$) and the positions of the targets $T_1, T_2, \ldots, T_l$, but it can detect the boundaries of the search area. Also, we assume the region $M$ is bounded. Here, we adopt the binary sensing model as the sensing model of the mobile sensors. Using this model, the detection of a mobile sensor is defined by 1 or 0. If $\|P_t - p_i\| \leq R_s$, outputs 1, which means that target is within the sensing range of at least one mobile sensor and outputs 0, otherwise.

We define two search scenarios follows:

In the first scenario, we assume that the mobile sensor network knows the total number of targets, but it is not aware of their positions. Consequently, the search should be stopped after finding all targets.

In the second case, it is assumed that the mobile sensors do not know the number of targets. Therefore, they should search entire area to find all possible targets. In other words, for the second case where the mobile sensor network is not aware of the number of neither targets nor their locations exploring the whole space can guarantee the accomplishment of the search task.

Here, our goal is to present a distributed control algorithm to drive the mobile sensors in the search area for identifying all targets in both search scenarios. We propose search algorithms that utilize two different grids (a cubic grid and a truncated octahedral grid) to carry out the search task.

Note that, vertices of a cubic (truncated octahedral) grid are the centres of cubes (truncated octahedrons) of that grid. In 1887, Kelvin [6, 166] proposed the truncated octahedron as the solution to the problem of finding space-filling arrangement of similar cells of equal volume with minimal surface area. Finding
such optimal space filling polyhedra is still an important open mathematical problem. Different 3D grids can be compared using a criteria called volumetric quotient. The volumetric quotient is defined as: $V_q = \frac{V_p}{V_s}$, where $V_p$ is the volume of a polyhedron and $V_s$ is the volume of its circumsphere. The volumetric quotient of a truncated octahedron is about 1.8 times that of a cube. Intuitively, this gives the hope that using a truncated octahedral grid for search leads to a fewer number of vertices to be searched and minimizes the time of the search. In the following, we use the term 3D grid for expression of both cubic grid and truncated octahedral grid.

**Definition 3.1.1.** Consider a cubic (truncated octahedral) grid cutting $M$ into equal cubes (truncated octahedrons) with the sides of $\frac{2R_s}{\sqrt{3}}$ ($\frac{2R_s}{\sqrt{10}}$). Let $\hat{V}$ be the infinite set of centres of all the cubes (truncated octahedrons) of this grid. The set $V = \hat{V} \cap M$ is called a cubic (truncated octahedral) covering set of the search area (Fig.3.1).

At first, the mobile sensor network builds a grid which is common among all mobile sensors, also it covers the search area. Then, they perform the search task by moving randomly to the vertices of the grid from different initial locations. During the search, the mobile sensors keep the information of the previously visited vertices of the grid and the targets such as the position of the detected targets and the visited vertices of the grid, which are representative of the previously searched area and communicates with its neighbours in the communication range at the discrete sequence of times $k = 0, 1, 2, \ldots$, to exchange the search in-
The neighbouring set of mobile sensor $i$ is defined by

$$S_{i,R_c} = \{ p \in \mathbb{R}^3; \| (p - p_i(k)) \| \leq R_c \}$$

(3.1)

Where $R_c$ is the mobile sensors’ communication range and $p_i(k) \in \mathbb{R}^3$ is the Cartesian coordinates of the mobile sensor $i$ at time $k$.

**Definition 3.1.2.** A target is said to be detected by a mobile sensor if it is located within the sensing region of the mobile sensor, which is the sphere of radius $R_s > 0$ centred at the mobile sensor.

The schematic of two mobile sensors searching for two randomly locate targets is shown in Fig.3.2.

Let $N_i(k)$ be the set of all mobile sensors $j, j \neq i$ and $j \in 1, 2, ..., n$ that belongs to the sphere $S_{i,R_c}(k)$ also, we assume that mobile sensor $i$ has $|N_i(k)|$ number of neighbours at time $k$. We take advantage of the notion of graph to define the relationships between neighbouring mobile sensors. As a result, the vertices $i$ and $j$ of the graph $g(k)$ are considered connected if the mobile sensors $i$ and $j$ are neighbours at time $k$. In the following, we impose a condition on the connectivity of the graph.

**Assumption 3.1.1.** There exists non-empty, an infinite sequence of contiguous, bounded time-intervals $[k_m, k_{m+1})$, $m = 0, 1, 2, ...$, such that across each $[k_m, k_{m+1})$, the graph $g(k)$ is connected [138].
3.2 Decentralized Random Search Algorithm

In this section, we introduce a two-stage decentralized grid based random search algorithm for finding targets in a bounded 3D area. At the first phase, the mobile sensors use consensus variables to build a common 3D grid. At the end of this stage, sensors are located on the vertices of the 3D grid. In the second stage, the mobile sensors perform the search task by moving randomly on the vertices of the 3D grid. To avoid unnecessary motion and to save the time and the energy consumption, during the search the mobile sensors share the information about the visited vertices and the detected targets with other mobile sensors passing on their communication range.

3.2.1 Building a Common Three-dimensional Covering Grid

In this section, we propose a distributed consensus algorithm to enable the mobile sensors to reach a common value and to build a common covering grid. Using our proposed distributed consensus algorithms each mobile sensors updates its information based on the information states of its neighbours in such a way that the final information state of each mobile sensor converges to a common value. It is said that agreement or consensus is achieved if the corresponding states of all agents converge to the same value. To make a covering truncated octahedral or cubic grid, we use 3D node placement strategies proposed in [6] as follows:

\[ V_1 = (x + \alpha_1 \frac{2R_s}{\sqrt{3}}, y + \alpha_2 \frac{2R_s}{\sqrt{3}}, z + \alpha_3 \frac{2R_s}{\sqrt{3}}) \]  \hspace{1cm} (3.2)

\[ V_2 = (x + (2\alpha_1 + \alpha_3) \frac{2R_s}{\sqrt{5}}, y + (2\alpha_2 + \alpha_3) \frac{2R_s}{\sqrt{5}}, z + \alpha_3 \frac{2R_s}{\sqrt{5}}) \]  \hspace{1cm} (3.3)

The vectors \( V_1, V_2 \) determine the position of the centres of the cubes and the truncated octahedrons which are used to tessellate the search area, respectively. Here, \( \alpha_1, \alpha_2 \) and \( \alpha_3 \in \mathbb{Z} \) and \( \mathbb{Z} \) is the set of all integers. The inputs to our algorithm are the sensing range \( R_s \) and the coordinates of the point \( (x, y, z) \), which act as a seed for the growing of the 3D grid. In [6], the point \( (x, y, z) \) was considered as an arbitrary known point. In this chapter, we assume that the mobile sensor network does not have information about this point a priori. Therefore, we use consensus variables to build a common coordinates \( (x, y, z) \) for the mobile sensor network, as a result, we will create a common 3D grid for the mobile sensor network. Initially, the sensors do not have a common coordinate system. Therefore, we assume that each mobile sensor has consensus variables \( x_i(k), y_i(k) \) and \( z_i(k) \) in its coordinate system. It is obvious that, any 3D grid is uniquely defined by a point \( q_i(k) = (x_i(k), y_i(k), z_i(k)) \) and \( R_s \). Thus, any \( q_i(k) \) and \( R_s \) uniquely define a 3D grid in \( M \), which will be denoted as \( V[q, R_s] \). The scalar parameters \( \alpha_1, \alpha_2, \alpha_3 \) and \( R_s \) along with the 3D consensus variables \( q_i(k) = [x_i(k) \ y_i(k) \ z_i(k)] \) characterize the coordinates of the vertices
of the grid. The mobile sensors will start with different values of the coordination variables \(x_i(0), y_i(0)\) and \(z_i(0)\), then eventually converge to some consensus values \(x_0, y_0\) and \(z_0\) which define a common coordinate system for the mobile sensor network.

Let \(p_i(k)\) be the coordinate of the mobile sensor \(i\) at time \(k\), and \(C[q](p)\) defines the closest vertices of the 3D grid \(V[q, R_s]\) to \(p\). The following rules for updating the consensus variables and the mobile sensors coordinates are proposed that drives all mobile sensors to the same constant values:

\[
\begin{align*}
    x_i(k + 1) &= \frac{x_i(k) + \sum_{j \in N_i(k)} x_j(k)}{1 + |N_i(k)|} \\
    y_i(k + 1) &= \frac{y_i(k) + \sum_{j \in N_i(k)} y_j(k)}{1 + |N_i(k)|} \\
    z_i(k + 1) &= \frac{z_i(k) + \sum_{j \in N_i(k)} z_j(k)}{1 + |N_i(k)|} \\
    p_i(k + 1) &= C[q_i(k), R_s](p_i(k))
\end{align*}
\]

Based on the rules (3.4) and (3.5), each mobile sensor updates its consensus variables using the average of its own and the consensus variables values of its neighbours located within its communication range.

**Theorem 3.2.1.** Suppose that Assumptions 3.1.1 hold and the mobile sensor network moves according to the laws (3.4), (3.5). Then, there exists a cubic (truncated octahedral) covering set \(V\) such that: \(\forall i = 1, 2, ..., n, \exists v \in V\); \(\lim_{k \to \infty} p_i(k) = v\).

**Proof.** Based on the update law (3.4) and the assumption 3.1.1, the consensus variables converges to some constant values as follows: \(x_i(k) \to x_0, y_i(k) \to y_0\) and \(z_i(k) \to z_0\).

(See [75] for the proof of convergence of consensus variables to some constant values). Since \(R_s, \alpha_1, \alpha_2\) and \(\alpha_3\) are common to all mobile sensors, therefore:

\[
\begin{align*}
    (x_i + \alpha_1 \frac{2R_s}{\sqrt{3}}, y_i + \alpha_2 \frac{2R_s}{\sqrt{3}}, z_i + \alpha_3 \frac{2R_s}{\sqrt{3}}) &\to \\
    (x_0 + \alpha_1 \frac{2R_s}{\sqrt{3}}, y_0 + \alpha_2 \frac{2R_s}{\sqrt{3}}, z_0 + \alpha_3 \frac{2R_s}{\sqrt{3}}).
\end{align*}
\]

\[
\begin{align*}
    (x_i + (2\alpha_1 + \alpha_3) \frac{2R_s}{\sqrt{5}}, y_i + (2\alpha_2 + \alpha_3) \frac{2R_s}{\sqrt{5}}, z_i + \alpha_3 \frac{2R_s}{\sqrt{5}}) &\to \\
    (x_0 + (2\alpha_1 + \alpha_3) \frac{2R_s}{\sqrt{5}}, y_0 + (2\alpha_2 + \alpha_3) \frac{2R_s}{\sqrt{5}}, z_0 + \alpha_3 \frac{2R_s}{\sqrt{5}}).
\end{align*}
\]

It means that, after a while, all mobile sensors will have a common covering 3D grid and, the rule (3.5) guaranties that all mobile sensors move to the vertices of the common 3D grid. \(\square\)
3.2.2 Randomized Grid Search

In this section, we propose a randomized search algorithm to transfer the mobile sensors to visit the vertices of the 3D grid. The 3D grid covers the entire search area, as a result, transmission of the mobile sensors to the vertices of the 3D grid implies that the whole area is searched by the mobile sensor network. We introduce the Boolean variables $b_v(k)$ and $b_T(k)$ which define the states of the vertices of the grid and the states of the targets at the time $k$, respectively. If vertex $v$ has been visited before time $k$ by any of the mobile sensors then, $b_v(k) = 1$ and $b_v(k) = 0$ otherwise. If the target $T$ has been detected by any of the mobile sensors before time $k$ then, $b_T(k) = 1$ and $b_T(k) = 0$ otherwise.

Now the following random algorithm for relocating of the mobile sensors is presented:

$$p_i(k+1) = \begin{cases} \hat{N}_{p_i(k)} & \text{with probability } \frac{1}{\|S(p_i(k))\|} \\ N_{p_i(k)} & \text{with probability } \frac{1}{\|\hat{S}(p_i(k))\|} \\ \text{if } \|\hat{S}(p_i(k))\| \neq 0 \\ \text{and } \|\hat{S}(p_i(k))\| = 0. \end{cases}$$  \quad (3.8)

Where, $S(p_i(k))$ is the set of the closest vertices of the 3D grid to $p_i(k)$ and $N_{p_i(k)}$ is a randomly selected element of $S(p_i(k))$. $\hat{S}(p_i(k))$ is considered as a set of the all elements of $S(p_i(k))$ which have not been already visited by any of the sensors also $\hat{N}_{p_i(k)}$ is its randomly selected element. $\|S(p_i(k))\|$ and $\|\hat{S}(p_i(k))\|$ are considered as the number of the elements in each set.

The algorithm (3.8) implies that the mobile sensors perform the search by moving randomly to the vertices of the covering grid. During the search, they keep the information of the previously searched vertices and share it with their neighbours. The mobile sensors randomly select their next destination from the unvisited vertices in their neighbourhood. In the case where all of the neighbouring vertices are visited they randomly move to one of the neighbouring vertices of the grid. Based on this algorithm, the mobile sensor network constantly moves in the search area.

To stop the search process after search of the entire area when the mobile sensor network is unaware of the number of the targets; the following condition should be satisfied:

$$p_i(k+1) = p_i(k)$$  \quad (3.9)

if $\forall \ v \in \mathcal{V}; \|\hat{S}(p_i(k))\| = 0$

**Theorem 3.2.2.** Suppose that the mobile sensors move according to the law (3.8),(3.9). Then for any number of mobile sensors, with probability 1 there exists a time $k_0$ such that:

$$\forall \ v \in \mathcal{V}, \ b_v(k_0) = 1$$

**Proof.** The algorithm (3.8),(3.9) describe an absorbing Markov chain that con-
sists of many absorbing states (that are impossible to leave) and many transient states. The vertices where the mobile sensors stop are considered as absorbing states also, the vertices that are visited during the search are considered as transient states. Algorithm (3.8) implies that the mobile sensors move randomly to the neighbouring unoccupied vertices of the 3D covering grid; this will continue until all mobile sensors neighbours are visited (an absorbing state). It is obvious that, from any initial state with probability 1, one of the absorbing states will be reached.

For the case where the mobile sensor network knows the number of targets, the search should be stopped after finding all targets. In other words:

$$p_i(k + 1) = p_i(k)$$

if $\forall T = \{T_1, T_2, ... T_l\}, \exists k, i ; \|P_T - p_i(k)\| \leq R_s$

Where, $T = \{T_1, T_2, ... T_l\}$ and $P_T = \{P_{T_1}, P_{T_2}, ... P_{T_l}\}$, are the sets of targets and their positions, respectively.

**Theorem 3.2.3.** Suppose that the mobile sensors move according to the law (3.8),(3.10). Then for any number of mobile sensors and any number of targets, with probability 1 there exists a time $k_0$ such that:

$$\forall t \in T, \ b_T(k_0) = 1$$

**Proof.** Vertices where sensors stop are considered as absorbing states (3.10), also the vertices of the grid which are visited during search are considered as transient states (3.8). As a result, the algorithm (3.8),(3.10) describes an absorbing Markov chain. The mobile sensors randomly move to visit unoccupied vertices; this continues until all targets are detected. Now the proof of this theorem immediately follows from the proof of Theorem 3.2.

### 3.3 Simulation Results

In this section, we present simulation results for different scenarios to explain the proposed algorithm. We consider six mobile sensors with random initial positions for the search task shown in Fig.3.3(a). By using the truncated octahedral grid and the control law (3.4) and (3.5), the mobile sensors move to the vertices of the common 3D grid. At the end of this stage, all mobile sensors are located at the vertices of the 3D grid as shown in Fig.3.3(b). Then, we apply the second stage of the algorithm to transfer the mobile sensors for the random search. Based on the decentralized random control law (3.8), the mobile sensors move randomly to visit unoccupied vertices of the covering grid. Throughout the search, each mobile sensor keeps information about the visited vertices and communicates with its neighbours in the communication range; this optimizes the search task because the mobile sensors do not search the vertices which have
Figure 3.3: Mobile sensors movement to the vertices of the common grid

been already visited by the others.
Figure 3.4: Mobile sensors movement to cover monitoring region (Truncated octahedron based grid)
Fig. 3.4(a) shows the performance of the search algorithm for the first scenario. As mentioned before, for this scenario we assume the mobile sensor network knows nothing about the environment and the positions of targets except the number of targets. In this case, the search goal is to identify two targets which are randomly placed in the search area. In this case, we evaluate our proposed distributed search coverage algorithm based on the time taken to find all targets. As shown, the mobile sensor network has identified the targets after 8 steps. Trajectories of the mobile sensors after 8 steps have been shown in Fig. 3.4(a). As shown, the control law (3.10) terminates the search process after detecting the given number of targets.

For the second scenario where the mobile sensor network is not aware of the number of targets, the whole area should be searched to identify all possible targets. In this case, the control law (3.9) terminates the search process after complete search of the whole area as shown in Fig. 3.4(b). It is obvious that, adding the number of mobile sensor means more vertices of the grid to be visited at the same time. As a result, the increase in the numbers of the mobile sensors makes the search more efficient because increasing the number of mobile sensors will decrease the time taken to the search task for both search scenarios. As it is seen from the Fig. 3.5 the performance of the system consisting of 14 mobile sensors is better than that of 8 mobile sensors and similarly 2 mobile sensors search performance is better than 1 mobile sensor. Also, the figure demonstrates the algorithm becomes faster with more mobile sensors. However, if the number of mobile sensors exceeds a limit, the effectiveness of the algorithm has less impact on the whole search performance.

To compare the performance of the grids, we did several simulations with different number of mobile sensors. Based on the simulations the average search length for the truncated octahedron grid was about 28 percent less than for the cubic grid. In the following, we carried out several simulations to compare the proposed grid based random search method with the randomized algorithms based on Levy flights proposed in [81] for detecting targets in a bounded 3D environment. Simulations result show that the average search time of the proposed grid based search method for two mobile sensors performing a search is far less than that of the Levy flight approach.

In the next set of simulations, we have evaluated the ability of the proposed search algorithm for handling some common uncertainties by assuming that one mobile sensor unexpectedly break during the search task. For this case, we consider six mobile sensors search for three targets. Fig. 3.6 shows that the other mobile sensors still be able to perform the search task in case of failure of one mobile sensors. It means that despite failure of one or more mobile sensor during the process the search task has been accomplished successfully.

Now, we investigate the effect of the size of the communication range on the performance of the proposed search algorithm. We have done three set of simulation. The communication ranges are chosen $R_c$, $\frac{R_c}{3}$ and $3R_c$. Simulation results show that, when the communication range is increased the search performance increases up to a certain point beyond which there is not much change in the
Figure 3.5: Searching time versus number of mobile sensors

Figure 3.6: Robustness against failure
performance of the system. As a result, it is not necessary to have the commun-
ication range to cover all the search area for better performance. Furthermore,
shorter communication range means lower power consumption.

3.4 Summary

In this chapter, we presented a set of distributed random laws for search in un-
known bounded three dimensional environments. The mobile sensors utilize a
truncated octahedral grid for the search process. Furthermore, they exchange
information with their neighbouring sensors to minimize the search time. The
performance of the proposed algorithm has been confirmed by extensive simul-
ations and the convergence with probability 1 for any number of mobile sensors
of the proposed algorithm has been mathematically rigorously proved.
Chapter 4

Distributed Bio-inspired Random Search for Locating Static Targets

Contents

4.1 Problem Formulation ......................................... 49
4.2 Decentralized Bio-inspired Random Search Algorithm 50
4.3 Simulation Results ........................................... 52
4.4 Summary ....................................................... 56

In many applications, it is not necessary to cover the whole area of interest. Monitoring only some specific points in the area of interest is sufficient and acceptable. For such cases, we assume only a limited number of discrete target points are to be monitored. The point of interest can be either static or mobile. Here we assume that the targets are stationary which are not able to move. In this chapter, we propose a novel decentralized bio-inspired random search algorithm for the search of randomly located static targets in 3D spaces [107]. Here, we combine the bio-inspired Levy flight random search mechanism [23] with a 3D covering grid proposed in [112] to optimize the search procedure. In previous works the Levy flight random walks took place on a continuous space but here the random walk occurs on a discrete grid. Unlike [112], here we assume the mobile sensors move randomly on the vertices of the covering grid with the length of the movement follow a Levy flight distribution. Based on this scheme, the Levy flight distribution will generate the length of the movement and the vertices of the covering truncated octahedral grid will improve the dispersion of the deployed mobile sensor. As a result, this method reduces the search task and optimizes the search procedure by dispersing efficiently the mobile sensors in the search area. In order to stop search after finding all targets and reduce the cost of operation, each mobile sensor communicates to other mobile sensors within its communication range to broadcast information regarding detected targets. Performance of this bio-inspired random search method is verified by simulations.
Moreover, to evaluate the performance of the proposed grid-based Levy flight algorithm we compare it to other random search strategies for locating sparse and clustered distribution of targets. Furthermore, we give a mathematically rigorous proof of the convergence of the proposed algorithm with probability 1 for any number of mobile sensors and targets.

The remainder of the chapter is structured in the following way: Section 4.1 describes the problem of bio-inspired random search in 3d environments. The proposed algorithm is explained in Section 4.2. Section 4.3 is devoted to evaluate the proposed algorithm by simulations. Section 4.4 concludes this work.

4.1 Problem Formulation

The objective of this chapter is to design a decentralized bio-inspired random search algorithm to drive a network of mobile sensors for target search in a bounded 3D area. We assume that the mobile sensors have no information about the the positions of the targets \( T_1, T_2, \ldots, T_l \) and search area \( (M \subset \mathbb{R}^3) \), but they can detect the boundaries of the search area. Let \( p_i(.) \in \mathbb{R}^3 \) be the Cartesian coordinates of the mobile sensor \( i \) and each mobile sensor has a sensing range of \( R_s > 0 \) and a communication range of \( R_c > \frac{4}{\sqrt{5}} R_s \). Our goal is to develop a distributed grid-based random search algorithm to drive the mobile sensors and to efficiently spread the mobile sensors in the search area for identifying all targets in a bounded 3D space. We take advantage of vertices of a 3D truncated octahedral grid proposed in [4, 7].

At first, we assume there is not a common coordinate system among all mobile sensors. As a result, the mobile sensors use consensus variables to build a common coordinate system and a common covering truncated octahedral grid.
Then, they perform the search task by moving randomly with the random steps length drawn from a Levy flight probability distribution on the vertices of the grid from different initial positions to locate all randomly located targets in the 3D area. During the search, the mobile sensors keep the information of the previously detected targets and communicates with its neighbours in the communication range at the discrete sequence of times \( k = 0, 1, 2, \ldots \), to exchange the search information.

**Definition 4.1.1.** A target is said to be detected by a mobile sensor if the Euclidean distance between the target and the mobile sensor is less than the sensing range. Mathematically speaking, the target \( T_i \) is said to detected by a mobile sensor \( s_i \), if and only if, \( \|P_{T_i} - p_i(k)\| \leq R_s \).

Where, \( P_{T_i} \) is position of target \( T_i \).

**4.2 Decentralized Bio-inspired Random Search Algorithm**

In this section, we present a two-stage decentralized bio-inspired random search algorithm for finding targets in a bounded 3D area. First, the mobile sensors build a common 3D truncated octahedral grid as explained in the previous chapter, section 3.2.1. Then, they perform the search task by moving randomly on the vertices of the 3D grid using random walks with a Levy-flight probability distribution.

Levy flight is a renowned bio-inspired random search mechanism with step lengths taken from a heavy-tailed probability distribution [129]. Using this distribution the probability of returning to a previously visited site is smaller, therefore advantageous when target sites are sparsely and randomly distributed [115, 170].

To optimize the search task, we supplement the bio-inspired Levy flight random search strategy to a covering truncated octahedral grid. It means that, the mobile sensors do the search task by moving randomly to the vertices of the covering grid with steps length drawn form Levy-flight probability distribution. At first we assume the mobile sensors do not have a common coordinate system and a common covering grid therefore the robots use consensus variables to build a common covering grid 3D [112]. In our previous search algorithm, mobile sensors performed the search task by moving randomly on the unvisited vertices in their neighbourhood. To optimize the search task and to minimize the time of search in this section, we propose a novel random search strategy for the mobile sensors. Using this method, the mobile sensors do the search task by moving randomly to the vertices of the covering grid based on the random walk generated by the Levy flight distribution. A Levy flight random walk pattern uses a Levy probability distribution as follows [172]:

\[
P_{\alpha,\gamma}(l) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} e^{-\gamma q^\alpha} \cos(ql) dq \tag{4.1}
\]
Where $\gamma$ denotes the scaling factor and $\alpha$ represents the shape of the distribution. Also, the distribution is symmetrical around $l$. In this chapter, the Levy flight provides a random walk with the random step length is drawn from an approximated Levy distribution proposed in [162] as follows:

$$P_{\alpha}(l) = (l)^{-\alpha} \quad (4.2)$$

Where $l$ is a random number, $\alpha > 0$ defines the length of walk and $1 < \alpha < 3$. Based on our method, a mobile sensor selects next vertices of the grid as follows:

Step 1: The mobile sensor’s direction will be drawn from a uniform distribution of angles in a range of $[0, 2\pi]$. In other words, the mobile sensor chooses a direction uniformly from a sphere.

Step 2: The mobile sensor chooses a step length from a Levy flight distribution where the power law parameter $\alpha$ determines the distance travelled by the mobile sensors between direction changes.

Step 3: The mobile sensor moves to the closest vertices of the grid to the point calculated in Step 2 and 3.

Step 4: If a target is ever within the sensing range of the mobile sensor, then the target is said to be detected. If no target is found, the mobile robotic sensor repeats its search according to steps mentioned above.

Note that the mobile sensors move from one vertex of the grid to another until they detect all targets. During the search process if a new random walk places a mobile sensor out of the boundary of the search area it will be rejected and a new random walk will be drawn from Levy probability distribution.

Let $\hat{V}$ be the set of all vertices of the 3D covering grid and $\hat{v} \in \hat{V}$ is a randomly selected element of $\hat{V}$ with Levy flight probability distribution. We propose the following random algorithm for the mobile sensors to search the 3D area:

$$p_i(k + 1) = \hat{v} \quad (4.3)$$

It means that the mobile sensors use steps 1,2,3 to select an element of $\hat{V}$ in each step and move on it.

Here we assume that the Boolean variables $b_T(k)$ defines the states of the targets at the time $k$. If target $T$ has been detected by any of the mobile sensors before time $k$ then, $b_T(k) = 1$ and $b_T(k) = 0$ otherwise. Throughout the search, each mobile sensor communicates to other mobile sensors in the communication range at the discrete sequence of times $k = 0, 1, 2$, to exchange the information about detected targets. The search process should be stopped after finding all targets.

In other words,

$$p_i(k + 1) = p_i(k) \quad (4.4)$$

if $\forall \ T = \{T_1, T_2, ... T_l\} ; \exists \ k, i \ ; \ |(P_T - p_i(k))| \leq R_s$

Where, $T = \{T_1, T_2, ... T_l\}$ and $P_T = \{P_{T_1}, P_{T_2}, ... P_{T_l}\}$, are the sets of targets and their positions, respectively.

**Theorem 4.2.1.** Suppose that the mobile sensors move according to the law \((4.3),(4.4)\). Then for any number of mobile sensors and any number of targets,
with probability 1 there exists a time $k_0$ such that:

$$\forall \ t \in T, \ b_T(k_0) = 1$$

**Proof:** The algorithm (4.3),(4.4) describes an absorbing Markov chain that consists of a number of absorbing states (that are impossible to leave) and many transient states. Vertices, where mobile sensors stop, are considered as absorbing states (4.4). Also, the vertices of the grid which are visited during the search are considered as transient states (4.3). Relation (4.3) implies that the mobile sensors randomly move to visit unvisited vertices, this continues until all targets are detected (an absorbing state). It is clear that from any initial state with probability 1, one of the absorbing states will be reached.

### 4.3 Simulation Results

In this section, we conduct several simulations using MATLAB R2014b to evaluate the performance of the proposed bio-inspired Levy flight search algorithm for different number of mobile sensors and for clustered and sparsely distributed targets in bounded three dimensional environments. At the first set of simulations, we assume that three targets are randomly distributed within the search area, and they are shown as stars, see Fig.4.2(b). The initial positions of all mobile sensors and targets are generated randomly within the search area. For this case, a team of four mobile sensors are deployed in the area to locate clustered targets. Fig.4.2(b) indicates the results of searching for three randomly distributed stationary targets within the search region. As shown in Fig.4.2(a), using the update law proposed in the previous chapter, first the mobile sensors build a common covering truncated octahedral grid and after that all mobile sensors move to the vertices of this grid. Then, the mobile sensors perform the search task by moving randomly by the random walk taken by a Levy-flight probability distribution on the vertices of the grid. Fig.4.2(b) shows the trajectories of the mobile sensors during the search. For this case, the mobile sensors have detected the targets after 15 steps. We performed a simulation to compare the performance of our proposed Levy-flight grid-based search algorithm with Levy flight without grid. Our simulation (Fig.4.3) demonstrates that using Levy flight without a grid, the mobile sensors has detected the targets after 800 steps which is far more than the time taken by Levy flight grid based method.

For the second case, we assume that targets are sparsely distributed in the search environment. Fig.4.4 shows paths of the mobile sensors doing the search task by moving randomly to the vertices of the common truncated octahedral grid to find the targets. For this case, the mobile sensors have detected the targets after 47 steps. Simulation results in Fig.4.2 and Fig.4.4 demonstrate that for both cases the mobile sensors have detected the given targets after a reasonable amount of time (15 and 47 steps) by moving randomly to the vertices of the common covering grid. It means that, simulations verify the effectiveness of the proposed decentralized bio-inspired random search algorithm for finding targets.
Figure 4.2: Trajectories of four mobile sensors performing grid-based Levy flight random search for locating clustered targets. Mobile sensors’ trajectories denoted by -, vertices of the common grid by o, Targets by star in both cases.
Figure 4.3: Levy flight without grid search: trajectories of four mobile sensors performing Levy flight search for three targets. Mobile sensors’ trajectories denoted by -, targets by star.

Figure 4.4: Trajectories of mobile sensors performing bio-inspired Levy flight random search for locating sparsely located targets. Mobile sensors’ trajectories denoted by -, vertices of the common grid by o, targets by star.
The next set of simulations are carried out to compare the proposed grid-based Levy flight search algorithm to the Levy flight search algorithm without using grid, our random search algorithm proposed in the last chapter and to the grid-based Levy walk, neighboring search, and normal distribution methods.
Table 4.1: Comparison of different search styles for finding sparsely located targets

| Type of random walk                             | Average search time |
|------------------------------------------------|---------------------|
| Levy – flight                                    | 1300                |
| Grid – based – Levy – flight                     | 87                  |
| Grid – based – normal – distribution            | 125                 |
| Neighboring – random – search                   | 300                 |

based random search with the step length taken by a normal distribution for detecting targets in a bounded 3D environment. In the following figures and table, for notational convenience we call the proposed grid-based Levy flight search algorithm, the Levy flight search algorithm without using a grid, our random search algorithm proposed in [112] and the grid-based random search with the step length taken by a normal distribution as grid-based Levy flight, Levy flight, neighboring search and normal distribution, respectively.

As these algorithms are stochastic, we run each simulation in 10 trials. Moreover, we compare these random search methods using different number of mobile sensors and targets with different initial positions. Performance of the simulations is measured by the total steps, which is indicative of the total time spent to locate targets. Fig.4.5(a) shows simulation results for finding four clustered targets in a bounded 3D area using 3, 5 and 7 mobile sensors by four random search methods. As shown, the proposed grid-based Levy flight method outperforms the other random search strategies for locating these targets. Fig.4.5(b) displays the simulation results for locating 4, 6 and 8 clustered targets in 3D area using three mobile sensors. This simulation demonstrates that using our proposed algorithm the mobile sensors detected the given number of targets in the least time.

To evaluate the proposed algorithm for locating sparsely located targets, we did several simulations with different number of targets and compared our proposed grid-based Levy walk random search strategy to the others. Table.4.1 demonstrates that the proposed grid-based Levy flight strategy out-performs other random walk strategies in locating targets.

4.4 Summary

In this chapter, we proposed an efficient bio-inspired random search algorithm to drive mobile sensors to find clustered and sparsely distributed targets in bounded 3D areas. The proposed method combines the bio-inspired Levy flight random search mechanism for determining the length of the walk with vertices of a covering truncated octahedral grid to optimize the search procedure.

We showed that moving randomly on the vertices of the covering grid can minimize the time of locating randomly located objects in the search area by im-
proving dispersion of the robots. Simulation results demonstrated that the proposed algorithm outperform to the other random search methods. Also, we gave mathematically rigorous proof of convergence with probability 1 of the proposed algorithm for any number of mobile sensors and targets.
Chapter 5

Distributed Bio-inspired Algorithm for Search of Moving Targets

Contents

5.1 Problem Statement .............................................. 59
5.2 Distributed Bio-inspired Levy Walk Random Search Algorithm ............................................. 62
5.3 Simulation Results .................................................. 64
5.4 Summary ............................................................ 66

In the previous chapter, we introduced a decentralized random search algorithm for detecting static targets in three dimensional spaces. In this chapter, we study the problem of detecting mobile targets in a bounded 3D environment by a network of mobile sensors. Compare to a single mobile sensor, a network of mobile sensors can complete the search task in cheaper, faster, and more efficient way [8]. A network of mobile sensors can handle the uncertainties and accomplish the task even when some mobile sensor fail during the search [34]. As a result, it can significantly improve the efficiency and provide better robustness and adaptability. A mobile or dynamic target has the ability to change its location. A running vehicle can be represented as a dynamic target. The main focus of our proposed search method is the movement of the dynamic target which is complicated compared to the static one [181]. In [83], Koopman showed that the Lawnmower search strategy is optimal with stationary targets in 2D spaces, but the analysis does not apply to the problem of search for moving targets. In [60], authors demonstrated the advantages of random Levy walk search over the lawnmower strategy especially for moving targets in 2D spaces. The work presented here addresses significant gaps in the deployment of a team of mobile robotic sensors for the search of mobile targets in three-dimensional spaces. We introduce a random search algorithm based on the Levy walk in 3D spaces. Furthermore, we take advantage of vertices of a covering grid to efficiently disperse
CHAPTER 5. DISTRIBUTED BIO-INSPIRED ALGORITHM FOR SEARCH OF MOVING TARGETS

the mobile robot for the search task.
Here, we use our algorithm presented in the previous chapter for identifying moving targets by combining the bio-inspired Levy flight random search mechanism with a 3D covering grid proposed in [112]. Unlike the last chapter, in this chapter we assume targets are moving in the search space randomly. Using proposed algorithm, the mobile robotic sensors randomly move on the vertices of the covering grid with the length of the movement follow a Levy flight distribution. The vertices of the covering truncated octahedral grid optimize the search task by efficiently spreading mobile sensors in the search area. To stop the search procedure, each sensor communicates to the other sensors in its communication range to broadcast information regarding detected targets. The performance of this grid-based random search method is verified by extensive simulations.

To evaluate the performance of the proposed grid-based Levy walk algorithm we compare it to the Levy walk random search strategy for detecting moving targets. Furthermore, we give a mathematically rigorous proof of the convergence of the proposed algorithm with probability 1 for any number of mobile sensors and moving targets. The proposed algorithm is distributed, and it relies only on local sensing and communication, and does not require absolute positioning system.

The rest of the chapter is structured in the following way. Section 5.1 describes the problem of search by a network of mobile sensor sensors in 3D environments. The proposed algorithm is described in Section 5.2. Section 5.3 presents simulation results and section 5.4 concludes this work.

5.1 Problem Statement

In this chapter, we study the problem of detecting mobile targets moving in a bounded 3D area using a network of mobile robotic sensors. Let \( l \) mobile robotic sensors are deployed to detect \( m \) moving targets in the search area. Also, let \( p_i(\cdot) \in \mathbb{R}^3 \) be the Cartesian coordinates of the mobile sensor \( i \). Let \( T_{m_1}, T_{m_2}, \ldots, T_{m_l} \) be \( l \) number of mobile targets in the search area that the mobile sensors have no information about their positions, the spatial distribution and the movement of the targets but they know the total number of mobile targets. Suppose that the search area \( M \subset \mathbb{R}^3 \) is unknown to the mobile robotic network, but the mobile robotic sensors can detect the boundaries of \( M \). The problem of detecting moving targets depends not only on the mobility behavior of the robotic sensors but also on the movement of the mobile targets themselves.

In practice, mobile targets can have a wide variety of movement patterns. In this work, we consider the case where mobile robotic sensors and targets move randomly in the search area with Levy walk/normal probability distributions, respectively. Fig.5.1(a) shows trajectories of mobile sensors/targets moving randomly in the search area. Here we assume the moving targets select a random location to travel to next location depending on a normal probability distribution \( p(x) \). To find the next location for a mobile target, a point is chosen uniformly
(a) Trajectories of mobile sensors/targets performing grid-based Levy flight random search: Mobile sensors’ trajectories denoted by -, Mobile sensors’ denoted by sphere, Mobile targets’ trajectories represented by –, Mobile targets’ by star

(b) Mobile sensors represented by their sensing sphere, communication between robots denoted by —

Figure 5.1: A schematic of mobile sensors/targets
at random on the surface of a unit sphere. This point establishes the direction for the step. Second, the step length $\lambda$ is determined from a probability density. We take advantage of simple kinematic model to describe the mobile targets’ motions in the three-dimensional environment as follows:

$$
x_t(k + 1) = x_t(k) + \lambda \cos(\theta) \cos(\psi)
$$
$$
y_t(k + 1) = y_t(k) + \lambda \cos(\theta) \sin(\psi)
$$
$$
z_t(k + 1) = z_t(k) + \lambda \sin(\theta)
$$  \hspace{1cm} (5.1)

Where $x_t(k)$, $y_t(k)$ and $z_t(k)$ are the Cartesian coordinates of the target. As mentioned earlier, the mobile targets chooses randomly their directions ($\theta$ and $\psi$) between $[0, 2\pi)$.

We assume the mobile robotic sensors are equipped with short range distance sensors. These sensors have a maximum sensing range $R_s > 0$. It means that the mobile robotic sensors can only sense the environment and detect mobile targets within their sensing area, which is the sphere of radius $R_s$ centered at the sensor. Moreover, we assume that a mobile target will be identified when it falls into the sensing range of a mobile robotic sensor. Schematic of two mobile sensors moving randomly to find two moving targets is shown in Fig.5.2. To back a moving mobile sensor to the search area when a mobile sensor reaches the boundary of the search area, it is given a new random heading that would direct it back to the search area. Here, our search goal is, to detect every moving target by at least one of the mobile robotic sensors and the target searching task is accomplished when all moving targets are detected.

We use a bio-inspired distributed random search algorithm proposed in the last chapter to efficiently spread the mobile sensors in the search area for identifying moving targets in a bounded 3D space. To disperse the mobile sensors in the search area, we take advantage of vertices of a 3D truncated octahedral grid proposed in [4]. To detect moving targets, the mobile sensors build a common covering truncated octahedral grid. Then, they perform the search task by moving randomly with the step length taken by a Levy flight probability distribution on the vertices of the grid from different initial locations. As shown in Fig.5.1(b), the mobile sensors are modeled as spheres having a sensing radius of $R_s$ that are randomly moving within the search area. During the search procedure, the mobile sensors keep the information of the detected mobile targets in their memory and share it with other sensors passing in their communication range. As shown, mobile sensors 1 and 5 as well as mobile sensors 1 and 3 are in their communication range; as a result, they update each other by exchanging the search information. Therefore, the mobile sensors 3 and 5 communicate indirectly through mobile sensor 1 and this will optimize the search task, because the mobile sensors do not search for the mobile targets that have been previously detected by other sensors.
mobile sensors denoted by sphere (sensing range), moving targets denoted by stars, detected area shown by yellow tape

5.2 Distributed Bio-inspired Levy Walk Random Search Algorithm

In this section, we present a distributed grid-based random search algorithm for detecting moving targets in bounded 3D spaces. Based on this algorithm, the mobile robotic sensors build a common 3D truncated octahedral grid. Then, they perform the search task by moving randomly on the vertices of the 3D grid using random walks with a Levy-flight probability distribution. Here, the mobile sensors are constrained to operate within a bounded search region, constructed as the cuboid area shown in Fig.5.3. To keep the mobile sensors/targets in the search area at all times at each step, the length of the next Levy walk is calculated to ensure that the travel from the current position to the next position will not place the mobile sensors/targets outside of the search area. If this new length places the mobile sensor out of the bounded search area, that value is rejected, and a new length is drawn. The process is repeated for each step until an acceptable length is provided.

Levy flight is a renowned bio-inspired random search mechanism with step lengths taken from a heavy-tailed probability distribution [122]. Using this distribution the probability of returning to a previously visited site is smaller. Therefore, advantageous when target sites are sparsely and randomly distributed [170]. To optimize the search task, we supplement the bio-inspired Levy flight random
search strategy to a covering truncated octahedral grid. It means that mobile sensors do the search task by moving randomly to the vertices of the covering grid with random movements with Levy-flight probability distribution. As mentioned earlier, at first we assume that the mobile sensors do not have a common coordinate system and a common covering grid therefore the sensors use consensus variables to build a common covering grid 3D as explained in chapter 3, section 3.2.1 [112]. In our previous algorithm sensors performed the search task by moving randomly on the unvisited vertices in their neighbourhood. To optimize the search task and to minimize the time of the search, in this section, we propose a novel random search strategy for the mobile sensors. Using this method, the sensors do the search task by moving randomly to the vertices of the covering grid based on the random walk generated by the Levy flight distribution. A Levy flight random search uses a Levy probability distribution as follows [172]:

$$P_{\alpha,\gamma}(l) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} e^{-\gamma q^\alpha} \cos(q l) dq$$  \hspace{1cm} (5.2)$$

Where $\gamma$ denotes the scaling factor and $\alpha$ represents the shape of the distribution. Also, the distribution is symmetrical around $l$. In this chapter, the Levy flight provides a random walk with the random step length is drawn from an approximated Levy distribution proposed in [162] as follows:

$$P_{\alpha}(l) = (l)^{-\alpha}$$  \hspace{1cm} (5.3)$$
Where \( l \) is a random number, \( \alpha > 0 \) defines the length of walk and \( 1 < \alpha < 3 \).

Let \( \hat{V} \) be the set of all vertices of the 3D covering grid and \( \hat{v} \in \hat{V} \) is a randomly selected element of \( \hat{V} \) with Levy flight probability distribution. We use the following random algorithm proposed in chapter 4 section 4.2, for the mobile sensors to search the 3D area:

\[
p_i(k + 1) = \hat{v}
\]

Here we assume that the Boolean variables \( b_T(k) \) defines the states of the moving targets at the time \( k \). If a moving target \( T \) has been detected by any of the mobile sensors before time \( k \) then, \( b_T(k) = 1 \) and \( b_T(k) = 0 \) otherwise. Throughout the search, each mobile sensor communicates to other sensors in the communication range at the discrete sequence of times \( k = 0, 1, 2 \), to exchange the information about detected mobile targets. The search process should be stopped after finding all targets.

In other words,

\[
p_i(k + 1) = p_i(k) \quad (5.5)
\]

\[\text{if } \forall T = \{ T_1, T_2, ..., T_m \}, \exists k, i ; \left\| (P_T - p_i(k)) \right\| \leq R_s\]

Where, \( T = \{ T_1, T_2, ..., T_m \} \) and \( P_T = \{ P_{T_1}, P_{T_2}, ..., P_{T_m} \} \), are the sets of mobile targets and their positions, respectively.

**Theorem 5.2.1.** Suppose that the mobile sensors move according to the law (5.4), (5.5), and the mobile targets move based on the law (5.1). Then for any number of mobile sensors and any number of moving targets, with probability 1 there exists a time \( k_0 \) such that:

\[\forall t \in T, \quad b_T(k_0) = 1\]

**Proof:** The algorithm (5.4) describes an absorbing Markov chain that consists of a number of absorbing states (that are impossible to leave) and many transient states. Vertices where mobile sensors stop are considered as absorbing states (5.5). Also, the vertices of the grid which are visited during search are considered as transient states (5.4). Relation (5.4) implies that the mobile sensors randomly move to visit unvisited vertices, this continues until all mobile targets are detected (an absorbing state). It is obvious that, from any initial state with probability 1, one of the absorbing states will be reached.

### 5.3 Simulation Results

In this section, we carry out a simulation study to evaluate the performance of the proposed bio-inspired distributed laws for search of mobile targets in bounded three dimensional environments. As mentioned before, the mobile targets can adopt a wide variety of movement patterns. In this work, we consider the mobile targets choose a direction uniformly from a sphere. Then, it chooses a step length from a normal distribution. The initial positions of all mobile sensors and targets are randomly generated within the search area. As these algorithms are
Figure 5.4: Trajectories of two mobile sensors performing grid-based random search for detecting two moving targets: Mobile sensors’ trajectories denoted by solid line; moving targets trajectories represented by dashed line, vertices of the grid by o
stochastic, we run each simulation in 20 trials.
In the first set of simulations, we study the detection time of the mobile targets. It is obvious that, detection time depends on the mobility behaviour of the mobile sensors and on the movement of the moving targets themselves. We assume that a team of two mobile sensors are deployed in the area to detect two mobile targets moving randomly in the search area. As mentioned earlier, a mobile target is detected when it falls into the sensing range of a mobile sensor. As shown in Fig.5.4(a), using the updating law proposed in 3.2.1, the mobile sensors build a common covering truncated octahedral grid then all mobile sensors move to the vertices of this grid. Subsequently, the mobile sensors perform the search task by moving randomly with the random walk taken from a Levy-flight probability distribution on the vertices of the grid.

Fig.5.4(b) shows trajectories of the mobile sensors doing the search task by moving randomly to the vertices of the common truncated octahedral grid to find the moving targets. As shown in Fig.5.4(b), the mobile sensors have detected the targets after 27 steps.

Note that, to back a moving target to the search area when a moving target reaches the boundary of the search area, it is given a new random heading that would direct it back into the search area.

The second set of simulations are carried out to compare the proposed grid based Levy walk method to the Levy walk method without grid for detecting two moving targets. Fig.5.5 shows the simulation results for detecting two moving targets in a 3D area using two mobile sensors using Levy walk random search method. Performance of the simulations are measured by the total steps, which is indicative of the total time spent to detect moving targets. Simulation results demonstrate that using the proposed grid-based random algorithm the mobile sensors detect the given number of targets in less time compare to the Levy flight method without grid as using the common grid improves mobile sensors dispersion in the search area.

In the next set of simulations, we have evaluated the ability of the proposed search algorithm for handling some uncertainties by assuming that two mobile sensors unexpectedly fail during the search task. For this case, we consider four mobile sensors search for two targets. Our simulations show that the other mobile sensors continue to perform the search task despite failure on some mobile sensors.

5.4 Summary

In this chapter, we considered the problem of detecting mobile targets moving randomly in a bounded 3D space by a mobile sensor network. We presented a bio-inspired grid-based random search algorithm to drive mobile sensors for search of moving targets in the search areas. The proposed search method combines the bio-inspired Levy flight random search mechanism for determining the length of the walk with a covering truncated octahedral grid to optimize the search procedure. The presented algorithm is decentralized, and the control ac-
Figure 5.5: Trajectories of two mobile sensors performing Levy walk random search for detecting two moving targets: Mobile sensors’ trajectories denoted by solid line, Mobile sensors by sphere, Mobile targets’ trajectories represented by dashed line, Mobile targets by +
tion of each sensor is based on the local information of its neighbors. Simulation results demonstrated that the proposed grid based algorithm outperforms to the Levy walk search method. Also, we gave mathematically rigorous proof of convergence with probability 1 of the proposed algorithm for any number of mobile sensors and targets.
Chapter 6
Decentralized Three Dimensional Formation Building

Contents

6.1 Problem Formulation ........................................ 70
   6.1.1 Multi-Robot Model ...................................... 70
   6.1.2 Formation Building Problem .............................. 71
6.2 Decentralized Consensus Based Formation Building Algorithm ........................................ 72
6.3 Decentralized 3D Formation Building with Anonymous Robots ........................................ 79
6.4 Simulation Results ............................................ 80
6.5 Summary ......................................................... 84

In this chapter, we study the problem of 3D formation building in 3D environments [113]. Our goal is to design a distributed control algorithm to coordinate a group of mobile robotic sensors such that they achieve a particular geometric pattern [37]. In particular, here, we introduce a random formation building algorithm for anonymous robots in 3D spaces. First, we present a decentralized consensus-based control algorithm for the problem of 3D formation building for nonholonomic mobile robotic sensors in 3D environments. Based on the proposed algorithm, the mobile robotic sensors only communicate with their neighbors located in their communication range to build and maintain a desired 3D configuration. As a result, this method is a good option for the cases where the mobile robotic sensors have limited communication range due to the economic reasons or because of physical constraints (underwater environments). Unlike previous works [130], using this algorithm, the mobile robotic sensors move in not only the same direction in 3D spaces but also they finally build a given 3D geometric configuration and move with the same speed [110]. Furthermore, in this study, the motion of each robot is modeled by a nonlinear model also we take into account the standard constraints on mobile robotic sensors’ angular...
acceleration and linear velocity [93].
At second part, we study the problem of formation building for the case when the mobile robotic sensors are unaware of their positions in the configuration. We present a decentralized random motion coordination law for the multi-robot system so that the mobile robotic sensors reach consensus on their positions and form a desired three dimensional geometric pattern from any initial positions. The performance of the proposed three dimensional formation building algorithm is validated by numerical simulations. Also, we give mathematically rigorous proof of convergence of the proposed algorithms to the given geometric configurations.

The rest of the chapter is organized as follows: In Section 6.1, we introduce a kinematic model for a group of mobile robotic sensors moving in a three-dimensional space. In Section 6.1.2, we formulate the problem. In Section 6.2, a novel consensus-based algorithm for formation building in 3D environments is presented. In Section 6.3, we introduce a 3D decentralized random formation building algorithm for anonymous mobile robotic sensors. Numerical simulation results that validate the presented algorithms are presented in Section 6.4. Finally, in Section 6.5 conclusion is summarized.

6.1 Problem Formulation

In this section, we define the kinematics equations of the motions of the multirobot system in 3D environments and we formulate the problem of formation control. In this chapter, the term mobile robotic sensor is used for expression of both unmanned aerial vehicles and autonomous underwater vehicles. Moreover, the standard inner product is denoted by $<;>$.

6.1.1 Multi-Robot Model

The system under consideration consists of n three degree-of-freedom nonholonomic mobile robotic sensors labelled 1 through n. All these robots move in the three dimensional workspace in continuous time, and they are controlled by bounded $v_i$, pitch ($q_i$) and yaw ($r_i$) rates. Let $\xi_i = \text{col}(x_i(t), y_i(t), z_i(t))$ denotes the column vector of the Cartesian coordinates of the mobile robotic sensor $i$. Also, let $\theta_i(t)$ and $\psi_i(t)$ denote the pitch and the yaw angle, respectively. Moreover, let $v_i(t)$, $q_i(t)$ and $r_i(t)$ be the linear velocity, the pitch and the yaw angular velocities of the robot $i$, respectively. Note that, the mobile robotic sensors considered in this chapter do not have roll motion. Then, the mobile robotic sensors kinematic equations of motion are given by the following equations:

$$\dot{\xi}_i(t) = v_i(t)c_i(t)$$
$$\dot{c}_i(t) = u_i(t)$$

(6.1)
for all $i = 1, 2, ... n$. Here, $c_i(t) \in \mathbb{R}^3$, $\|c_i(t)\| = 1$, is the unit vector along the robots centerline described by:

$$
\vec{c}_i = \begin{pmatrix}
\cos(\theta_i(t)) \cos(\psi_i(t)) \\
\cos(\theta_i(t)) \sin(\psi_i(t)) \\
-\sin(\theta_i(t))
\end{pmatrix}
$$

(6.2)

Also, introduce three-dimensional vectors $V_i(t)$ of the robots velocities by $V_i(t) = v_i(t)c_i(t)$.

$\vec{u}_i(t) \in \mathbb{R}^3$ is the two-degree-of-freedom control. Furthermore, $\vec{u}_i(t)$ and $v_i(t) \in \mathbb{R}$ are the control inputs for all robots. We assume the following constraints on the control inputs:

$$
\left\langle \vec{u}_i(t), c_i(t) \right\rangle = 0 
$$

(6.3)

$$
\|\vec{u}_i(t)\| \leq u_{max}
$$

(6.4)

$$
V_{min} \leq v_i(t) \leq V_{max}
$$

(6.5)

Here $u_{max} > 0$ and also $0 < V_{min} < V_{max}$ are given constants.

### 6.1.2 Formation Building Problem

In this section, we explain the problem of formation building by nonholonomic mobile robotic sensors. Especially, we study the decentralized 3D formation building problem for $n$ mobile robotic sensors moving in three-dimensional envi-
environments such that, after transient state, the mobile robotic sensors constitute the desired formation and move collectively along a desired geometric pattern from any initial position. The desired geometric configuration of the mobile robotic sensors in the formation are characterized by a set of given numbers \(X_1, X_2, \ldots, X_n, Y_1, Y_2, \ldots, Y_n\) and \(Z_1, Z_2, \ldots, Z_n\). For all \(i = 1, \ldots, n\), \(Y_i - Y_j\) and \(Z_i - Z_j\) are the desired horizontal and vertical distances between the \(i\)th and the \(j\)th robots respectively. Also, \(X_i - X_j\) is the desired distance between the \(i\)th and the \(j\)th mobile robotic sensor along X axis.

**Definition 6.1.1.** A navigation law with the given configuration \(C = \{X_1, X_2, \ldots, X_n, Y_1, Y_2, \ldots, Y_n, Z_1, Z_2, \ldots, Z_n\}\) and initial conditions \(\theta_i(0), \psi_i(0), x_i(0), y_i(0), z_i(0)\) and \(v_i(0)\), \(i = 1, \ldots, n\) is said to be globally stabilizing, if there exist a Cartesian coordinate system and \(v_0\) such that the solution of the closed-loop system (6.1) with these initial conditions and the proposed navigation law in this Cartesian coordinate system satisfies:

\[
\lim_{t \to \infty} x_i(t) - x_j(t) = X_i - X_j \tag{6.6}
\]
\[
\lim_{t \to \infty} y_i(t) - y_j(t) = Y_i - Y_j
\]
\[
\lim_{t \to \infty} z_i(t) - z_j(t) = Z_i - Z_j
\]
\[
\lim_{t \to \infty} \psi_i(t) = 0
\]
\[
\lim_{t \to \infty} \theta_i(t) = 0 \tag{6.7}
\]
\[
\lim_{t \to \infty} v_i(t) = v_0 \tag{6.8}
\]

for \(i, j = 1, 2, \ldots, n, i \neq j\). Thus, the control objective is to design a navigation law to satisfy the conditions mentioned above for any initial conditions. The schematic of the mobile robotic sensors during formation building in a given configuration is shown in Fig. 6.1.

### 6.2 Decentralized Consensus Based Formation Building Algorithm

For a given time \(k > 0\), each mobile robotic sensor communicates with its surrounding neighbours in the communication range \(R_c\) at the discrete time instances \(k = 0, 1, 2, \ldots\). We assume a spherical communication model where each robot has the ability to obtain information of its neighbours within a sphere of radius \(R_c > 0\) defined by:

\[
S_{i,R_c} = \{p \in \mathbb{R}^3; ||(p - \zeta_i(k))|| \leq R_c\} \tag{6.9}
\]

Let \(N_i(k)\) be the set of mobile robotic sensors \(j, j \neq i\) and \(j = 1, 2, \ldots, n\) that at time \(k\) belong to the sphere \(S_{i,R_c}\), also we assume that robot \(i\) has \(|N_i(k)|\)
number of neighbours at time $k$. We consider a simple undirected graph $G(k)$ with vertex set $1, 2, ..., n$ to describe the relationships between the neighbouring mobile robotic sensors for any time $k$. The different vertices $i$ and $j$ of the graph $G(k)$ are connected by an edge if and only if, the robots $i$ and $j$ are neighbours at time $k$. In the following, we impose a condition on the connectivity of the graph.

**Assumption 6.2.1.** There exists an infinite sequence of non-empty, contiguous, bounded time-intervals $[k_j, k_{j+1})$, $j = 0, 1, 2, ...$ beginning at $k_0 = 0$ such that across each $[k_j, k_{j+1})$, the union of the collection $\{G(k) : k \in [k_j, k_{j+1})\}$ is a connected graph [138].

We use consensus variables $\tilde{v}_i(k)$ to reach a common speed of the formation, and the consensus variables $\tilde{x}_i(k), \tilde{y}_i(k), \tilde{z}_i(k)$ are used to build a common coordinates of the formation, and also the consensus variables $\tilde{\theta}_i(k)$ and $\tilde{\psi}_i(k)$ are utilized to achieve the common heading of the formation. The robots will start with different values of $\tilde{v}_i(0), \tilde{x}_i(0), \tilde{y}_i(0), \tilde{z}_i(0), \tilde{\theta}_i(0)$, and $\tilde{\psi}_i(0)$, and finally converge to some constant values which define a common formation orientation and speed for all robots.

**Assumption 6.2.2.** The initial values of the consensus variables: $\tilde{\theta}_i(0) \in [0, \pi)$ and $\tilde{\psi}_i(0) \in [0, \pi)$ for all $i = 1, 2, ..., n$.

We propose the following rules for updating the consensus variables:

$$\tilde{\psi}_i(k+1) = \frac{\tilde{\psi}_i(k) + \sum_{j \in N_i(k)} \tilde{\psi}_j(k)}{1 + |N_i(k)|}$$

$$\tilde{\theta}_i(k+1) = \frac{\tilde{\theta}_i(k) + \sum_{j \in N_i(k)} \tilde{\theta}_j(k)}{1 + |N_i(k)|}$$

$$\tilde{x}_i(k+1) = \frac{x_i(k) + \tilde{x}_i(k) + \sum_{j \in N_i(k)} x_j(k) + \tilde{x}_j(k)}{1 + |N_i(k)|} - x_i(k+1)$$

$$\tilde{y}_i(k+1) = \frac{y_i(k) + \tilde{y}_i(k) + \sum_{j \in N_i(k)} y_j(k) + \tilde{y}_j(k)}{1 + |N_i(k)|} - y_i(k+1)$$

$$\tilde{z}_i(k+1) = \frac{z_i(k) + \tilde{z}_i(k) + \sum_{j \in N_i(k)} z_j(k) + \tilde{z}_j(k)}{1 + |N_i(k)|} - z_i(k+1)$$

$$\tilde{v}_i(k+1) = \frac{\tilde{v}_i(k) + \sum_{j \in N_i(k)} \tilde{v}_j(k)}{1 + |N_i(k)|}$$

(6.10)

Based on (6.10), the mobile robotic sensors use the consensus variables to achieve a consensus on the heading, speed and mass centre of the formation.
Lemma 6.2.1. Suppose that assumptions 6.2.1 and 6.2.2 hold and the consensus variables are updated according to the decentralized control algorithm (6.10), then there exist constants \( \theta_0, \psi_0, x_0, y_0, z_0 \) and \( v_0 \) such that:

\[
\begin{align*}
\lim_{k \to \infty} \tilde{\theta}_i(k) &= \theta_0 \\
\lim_{k \to \infty} \tilde{\psi}_i(k) &= \psi_0 \\
\lim_{k \to \infty} x_i(k) + \tilde{x}_i(k) &= x_0 \\
\lim_{k \to \infty} y_i(k) + \tilde{y}_i(k) &= y_0 \\
\lim_{k \to \infty} z_i(k) + \tilde{z}_i(k) &= z_0 \\
\lim_{k \to \infty} \tilde{v}_i(k) &= v_0
\end{align*}
\] (6.11)

(See [75, 133] for the proof of convergence of consensus variables to some constant values).

Notice that, for all \( i = 1, 2, \ldots, n \), we consider a cartesian coordinate system with the \( x \)-axis in the direction of \( \theta_i(k) \). Thus, in this coordinate system \( \theta_i(k) = 0 \).

In the following, we present a decentralized formation building law in 3D spaces for the multi-robot system such that the robots will eventually move in the same direction of the \( x \)-axis in this Cartesian coordinate system, with the same speed and with the given geometric configuration.

We define the functions \( h_i(t) \) as:

\[
h_i(t) = x_i(t) + \tilde{x}_i(t) + X_i + t\tilde{v}_i(t).
\]

Next, we introduce \( T_{x_i}(t) \), \( T_{y_i}(t) \) and \( T_{z_i}(t) \) as:

\[
T_{x_i}(t) = \begin{cases} 
  h_i(t) + c_0 & \text{if } x_i(t) \leq h_i(t) \\
  x_i(t) + c_0 & \text{if } x_i(t) > h_i(t)
\end{cases}
\]

Note that \( c_0 > 0 \) is a constant such that

\[
c_0 > \frac{2V_{\max}}{u_{\max}} \quad (6.12)
\]

\[
T_{y_i}(t) = \begin{cases} 
  y_i(t) + \tilde{y}_i(t) + Y_i
\end{cases}
\]

\[
T_{z_i}(t) = \begin{cases} 
  z_i(t) + \tilde{z}_i(t) + Z_i
\end{cases}
\]

Also, for \( i = 1, 2, \ldots, n \), we consider \( T_i \) as fictitious targets moving with coordinates defined by:

\[
T_i = \begin{pmatrix} 
  T_{x_i}(t) \\
  T_{y_i}(t) \\
  T_{z_i}(t)
\end{pmatrix} \quad (6.14)
\]
CHAPTER 6. DECENTRALIZED THREE DIMENSIONAL FORMATION BUILDING

Furthermore, let $d_i$ be a three-dimensional vector as:

$$d_i = \begin{pmatrix} T x_i(t) - x_i(t) \\ T y_i(t) - y_i(t) \\ T z_i(t) - z_i(t) \end{pmatrix}$$ (6.15)

Which is the distance vector between robot $i$ and the fictitious target.

**Remark 6.2.1.** We assume that each mobile robotic sensor knows the coordination $(x_i, y_i, z_i)$ and the consensus variables of all of its neighboring mobile robotic sensors, also the configuration $C = \{X_1, X_2, \ldots, X_n, Y_1, Y_2, \ldots, Y_n, Z_1, Z_2, \ldots, Z_n\}$ and the constant $c_0$ are known to the multi-robot system a priori.

Since $\dot{c}_i = u_i(t)$ defined by (6.1), then the following correspondence between the control vector $u_i$ from (6.1) and the pitch ($q_i$) and yaw ($r_i$) rates can be defined:

$$\vec{u}_i = \dot{\theta}_i \vec{A}_i + \dot{\psi}_i \vec{B}_i$$ (6.16)

where, $\dot{\theta}_i = q_i$, $\dot{\psi}_i = r_i$ and also $A_i$ and $B_i$ are considered as three-dimensional vectors defined by:

$$\vec{A}_i = \begin{pmatrix} -\sin(\theta_i(t)) \cos(\psi_i(t)) \\ -\sin(\theta_i(t)) \sin(\psi_i(t)) \\ -\cos(\theta_i(t)) \end{pmatrix}$$ (6.17)

$$\vec{B}_i = \begin{pmatrix} -\cos(\theta_i(t)) \sin(\psi_i(t)) \\ \cos(\theta_i(t)) \cos(\psi_i(t)) \\ 0 \end{pmatrix}$$ (6.18)

for all $i = 1, 2, \ldots, n$.

We know that the direction of $\vec{u}_i$ is orthogonal to $\vec{V}_i$ and it can cause acceleration perpendicular to the mobile robotic sensors velocity vector($V_i$). The vector $u_i$ causes a turning motion to parallel the mobile robotic sensors velocity vector to the vector $d_i$. Furthermore, we assume that the control signal($u_i$) is in the same plane with $d_i$ and the velocity vector $V_i$, therefore we can write the direction of
\[ \vec{d}_i = \vec{u}_i - \frac{\langle \vec{d}_i(t), V_i(t) \rangle}{\|V_i\|^2} \hat{V}_i; \] (6.19)

We define \( \vec{u}_i = \frac{\vec{d}_i}{\|\vec{d}_i\|} \) as the unit vector of \( \vec{d}_i \). Now, we introduce the following decentralized control law for formation building in 3D environments:

\[ v_i(t) = \begin{cases} V_{\text{max}} & \text{if } x_i(t) \leq h_i(t) \\ V_{\text{min}} & \text{if } x_i(t) > h_i(t) \end{cases} \] (6.20)

\[ \vec{u}_i(t) = \vec{u}_i(t) \hat{\vec{u}}_i \] (6.21)

Where \( \vec{u}_i = u_{\text{max}} \text{sgn}(V_i, \hat{d}_i) \). \text{sgn}(...) defines the sign of angle between vectors \( V_i \) and \( \hat{d}_i \).

The controller 6.19-6.21 belongs to the class of switched controller systems, see e.g. [94, 136, 143, 153] Also, (6.16) implies that we can calculate the angular velocities as follows:

\[ q_i(t) = \langle u_i(t), A_i(t) \rangle \] (6.22)

\[ r_i(t) = \langle u_i(t), B_i(t) \rangle \] (6.23)

for all \( i = 1, 2, ..., n \).

**Lemma 6.2.2.** Suppose that \( q_i(t) \) and \( r_i(t) \) are updated according to the decentralized control law (6.22),(6.23), then both angular velocities are bounded.

By using (6.21, 6.22, 6.23) we can say that \( \|q_i(t)\| \leq \|\vec{u}_i(t)\| \|\hat{A}_i\| \) and \( \|r_i(t)\| \leq \|\vec{u}_i(t)\| \|\hat{B}_i\| \). Also, it is clear that \( \|\vec{u}_i(t)\| = u_{\text{max}}, \|\hat{A}_i\| = 1 \) and \( \|\hat{B}_i\| = \cos(\theta_i) \). This means that \( q_i(t), r_i(t) \) are bounded, and \( u_{\text{max}} \) is an upper bound to them. Now, we are in a position to present the main result of the chapter.

**Theorem 6.2.1.** Consider the mobile robotic sensors described by the equations (6.1), (6.2) and the constraints (6.3), (6.4), (6.5). Let \( C = \{X_1, X_2, ..., X_n, Y_1, Y_2, ..., Y_n, Z_1, Z_2, \ldots \} \) be a given configuration. Suppose that Assumptions 6.2.1 and 6.2.2 hold, and \( c_0 \) is a constant satisfying (6.11). Then, the decentralized control law (6.19), (6.20) and (6.21) is globally stabilizing with any initial conditions and the configuration \( C \).

**Proof:** Consider a fictitious target \( T_i(t) \) with coordinates defined by \( T_i(t) \) moving in the 3D space. Moreover, let

\[ \bar{T}_i(t) = \begin{pmatrix} \bar{T}x_i(t) \\ \bar{T}y_i(t) \\ \bar{T}z_i(t) \end{pmatrix} \] (6.24)
be another fictitious target moving in the 3D space with coordinates $\widetilde{T}_1(t)$ as follows:

$$\widetilde{T}_x_i(t) = \begin{cases} X_i + X_0 + t\tilde{v}_0 + c_0 & \text{if } x_i(t) \leq X_i + X_0 + t\tilde{v}_0 \\ x_i(t) + c_0 & \text{if } x_i(t) > X_i + X_0 + t\tilde{v}_0 \end{cases}$$

$$\widetilde{T}_y_i(t) = y_i(t) + \tilde{y}_i(t) + Y_i$$

$$\widetilde{T}_z_i(t) = z_i(t) + \tilde{z}_i(t) + Z_i.$$  (6.25)

Based on Lemma 6.2.1,

$$\lim_{t \to \infty} \widetilde{T}_i(t) = T_i(t)$$

The system defined by the equations (6.1),(6.20),(6.21), (6.22), (6.23) represents a closed-loop system with discontinuous right-hand sides. Suppose $\lambda_i(t)$ is the angle between the velocity vector $V_i(t)$ of the robot $i$ and the line-of-sight between the robot and $\widetilde{T}_i$. The equation $\lambda_i = 0$ defines a switching surface of this system. In order to prove convergence to the sliding mode, we show that with this control, $\lambda_i$ will converge to 0 in finite time. In [165], it has been shown that:

$$\dot{\lambda}_i(t) = a(t) - \bar{u}_i(t) - b(t),$$  (6.26)

where

$$|a(t)| \leq \frac{|\sin(\lambda_i(t))||V_i(t)||}{||\tilde{d}_i(t)||};$$  (6.27)

$$|b(t)| \leq \frac{|\sin(\beta_i(t))||V_i^T(t)||}{||\tilde{d}_i(t)||}.$$  (6.28)

$\beta_i(t)$ is the angle between the velocity vector $V_i^T(t)$ of $\tilde{T}_i$ and the line-of-sight from the robot $i$ to $\tilde{T}_i$. Also, $\tilde{d}_i(t)$ is defined as:

$$\tilde{d}_i(t) := \tilde{T}_i(t) - \xi_i(t)$$  (6.29)

(6.2), (6.29) imply that

$$||\tilde{d}_i(t)|| \geq c \quad \forall \quad t \geq 0.$$  (6.30)

It follows from (6.2) that

$$V_i^T(t) = \begin{pmatrix} V_i^{T_x}(t) \\ V_i^{T_y}(t) \\ V_i^{T_z}(t) \end{pmatrix}$$  (6.31)

$$V_i^{T_x}(t) = \begin{cases} \tilde{v}_0 & \text{if } x_i(t) \leq X_0 + X_i + t\tilde{v}_0 \\ v_i(t) & \text{if } x_i(t) > X_0 + X_i + t\tilde{v}_0 \end{cases}$$
Chapter 6. Decentralized Three Dimensional Formation Building

Switching Surface

Figure 6.3: Vector field around the switching surface

\[ V_i^{Ty}(t) = 0 \quad (6.32) \]
\[ V_i^{Tz}(t) = 0 \]

(6.32) and (6.5) imply that

\[ \|V_i^T(t)\| \leq V^M. \quad (6.33) \]

Suppose that in control law (6.21) the distance vector \( d_i \) replaced by \( \tilde{d}_i \). The mathematical relations (6.12), (6.26), (6.33), (6.30) implies that using this control law, there exists a constant \( \epsilon > 0 \) such that

\[ \dot{\lambda}_i(t) < -\epsilon \quad if \quad \lambda_i(t) > 0 \]
\[ \dot{\lambda}_i(t) > \epsilon \quad if \quad \lambda_i(t) < 0. \quad (6.34) \]

It means that \( \dot{\lambda}_i(t)\lambda_i(t) < 0 \). Therefore, there exists a time \( t_0 > 0 \) such that

\[ \lambda_i(t) = 0 \quad \forall \quad t \geq t_0. \quad (6.35) \]

Fig. 6.3 shows the vector field of the closed-loop system around the switching surface \( (\lambda_i(t) = 0) \). (6.34) implies that

\[ \dot{\lambda}_i(t) = 0 \quad \forall \quad t \geq t_0. \quad (6.36) \]

It clearly follows from (6.26), (6.27), (6.28) and (6.35) that

\[ \bar{u}_i(t) = -b(t) \quad (6.37) \]

where \( b(t) \) satisfies the inequality (6.28) for all sliding mode solutions. As a result, for any initial condition, the sliding mode solution is unique and well-defined. Moreover, (6.37), (6.33), (6.12) and (6.30) indicate that the constraint (6.4) holds for any sliding mode solution satisfying (6.34). It follows from (6.35)
that for all $t \geq t_0$ the vector $\tilde{d}_i(t)$ is parallel to the velocity vector $V_i(t)$ as a result, $y_i(t) = Y_0 + Y_i$ and $z_i(t) = Z_0 + Z_i$ for all $t \geq t_0$. Therefore, we can say that the second and third parts of the condition (6) follow from this. The fact that the velocity vector $V_i(t)$ is parallel to the vector $\tilde{d}_i(t)$ for all $t \geq t_0$ and $d_i$ replaced by $\tilde{d}_i$ in the control law (6.21) mean that

$$\tilde{d}_i(t) = \begin{pmatrix} c \\ 0 \\ 0 \end{pmatrix}$$

The first part of the conditions 6.6 follows from this. This completes the proof of Theorem 6.2.1.

### 6.3 Decentralized 3D Formation Building with Anonymous Robots

This section describes the problem of formation building in 3D environments for the case where the mobile robotic sensors are unaware of their positions in the configuration. In the following, we propose a random decentralized motion coordination law for the mobile robotic sensors so that they reach consensus on their positions and form a desired geometric pattern from any initial position in 3D spaces.

**Definition 6.3.1.** A navigation law with the given configuration $C = \{X_1, X_2, \ldots, X_n, Y_1, Y_2, \ldots, Y_n, Z_1, Z_2, \ldots, Z_n\}$ and anonymous robots is said to be globally stabilizing if for any initial conditions $\theta_i(0), \psi_i(0), x_i(0), y_i(0), z_i(0)$ and $v_i(0)$, there exists a permutation $p(i)$ of the index set $1, 2, \ldots, n$ such that for any of them there exists a Cartesian coordinate system and $v_0$ such that the solution of the closed-loop system (6.1) with these initial conditions and the proposed navigation law in this Cartesian coordinate system satisfies:

$$\lim_{t \to \infty} x_i(t) - x_j(t) = X_{p(i)} - X_{p(j)},$$

$$\lim_{t \to \infty} y_i(t) - y_j(t) = Y_{p(i)} - Y_{p(j)},$$

$$\lim_{t \to \infty} z_i(t) - z_j(t) = Z_{p(i)} - Z_{p(j)}. \quad (6.38)$$

Let $0 < \lambda < \frac{R_c}{2}$ be a given constant. Also, let $N > 1$ be a given integer. We consider a undirected graph $g$ as a representation of the set of vertices of the given configuration $C = \{X_1, X_2, \ldots, X_n, Y_1, Y_2, \ldots, Y_n, Z_1, Z_2, \ldots, Z_n\}$. We assume that graph $g$ is connected. The index permutation function $p(i)$ defines the position of the mobile robotic sensors in the configuration. As a result, to determine the position of the mobile robotic sensors in the configuration we need to build the index permutation function. In the following, we propose a random algorithm to build the index permutation function $p(i)$. 
Definition 6.3.2. A vertex $j$ of the graph $g$ is vacant at time $kN$ for robot $i$ if there is not any robot inside the sphere of radius $\lambda$ centered at the following point:

$$
\begin{pmatrix}
  x_i(kN) + \tilde{x}_i(kN) + X_j + kN\tilde{v}_i(kN) \\
  y_i(kN) + \tilde{y}_i(kN) + Y_j \\
  z_i(kN) + \tilde{z}_i(kN) + Z_j
\end{pmatrix}
$$

(6.39)

Let $b_i(k)$ be a boolean variable such that $b_i(k) = 1$ if there exists another robot $j \neq i$ within the sphere of radius $\lambda$ centered at the following point:

$$
\begin{pmatrix}
  x_i(kN) + \tilde{x}_i(kN) + X_i + kN\tilde{v}_i(kN) \\
  y_i(kN) + \tilde{y}_i(kN) + Y_i \\
  z_i(kN) + \tilde{z}_i(kN) + Z_i
\end{pmatrix}
$$

(6.40)

at time $kN$, and $b_i(k) = 0$ otherwise. We introduce the following algorithm for updating permutation $p(i)$ as follows:

$$
p((k+1)N,i) = \begin{cases}
  p(kN,i) & \text{if } b_i(kN) = 0 \\
  j & \text{with probability } \frac{1}{\|S(kN,i)\|}, \forall j \in S(kN,i) \\
  p(kN,i) & \text{if } b_i(kN) = 1 \text{ and } \|S(kN,i)\| > 1.
\end{cases}
$$

(6.41)

Where $S(kN,i)$ is a set consisting of $p(kN,i)$ and those vertices of $g$ that are vacant for robot $i$ at time $kN$ and connected to $p(kN,i)$.

Theorem 6.3.1. Consider the mobile robotic sensors and their constraints described by the equations (6.1), (6.2), (6.3), (6.4) and (6.5). Let $C = \{X_1, X_2, ..., X_n, Y_1, Y_2, ..., Y_n, Z_1, Z_2, ..., Z_n\}$ be a given configuration. Suppose that assumptions 6.2.1 and 6.2.2 hold, and $c_0$ is a constant satisfying (6.12). Also, assume that graph $g$ is connected. Then, for any initial conditions there exists an integer $N_0 > 0$ such that for any $N > N_0$, the decentralized control law (6.10), (6.20), (6.21), (6.22) and (6.23) with probability 1 is globally stabilizing with the given configuration.

Proof: Algorithm (6.41) implies that the mobile robotic sensors move randomly to the neighbouring unoccupied vertices of the given configurations graph; this will continue until all mobile robotic sensors are positioned in different vertices of the graph $g$. These states are considered as absorbing states and they are impossible to leave. As a result, the algorithm (6.41) describes an absorbing Markov chain. It is obvious that, from any initial state with probability 1, one of the absorbing states will be reached.

6.4 Simulation Results

In this section, a simulation study is carried out to evaluate the performance of the proposed consensus-based decentralized navigation laws for formation build-
CHAPTER 6. DECENTRALIZED THREE DIMENSIONAL FORMATION BUILDING

Figure 6.4: Trajectories of six mobile robotic sensors during formation building in 3D environments. For simulation, we used MATLAB R2014b with simu-
Table 6.1: Simulations Parameters

| Parameter | Value | Comment |
|-----------|-------|---------|
| $T_s$ | .01 | Sampling time |
| $R_c$ | 100 | Communication range |
| $\lambda$ | 20 | $0 < \lambda < \frac{R_c}{2}$ |
| $N$ | 10 | The given integer |
| $V_{max}$ | 8 m/s | Maximum linear velocity for the mobile robotic sensors. |
| $V_{min}$ | 2 m/s | Minimum linear velocity for the mobile robotic sensors. |
| $U_{max}$ | 2 | Maximum angular velocity. |

Simulation parameters shown in Table. 6.1 for all configurations. In the first set of simulations, we consider a fleet of six mobile robotic sensors defined by (6.1) with the constraints (6.3) and (6.4) for an edge and a random formation building in a 3D environment. The initial headings and positions of all mobile robotic sensors are generated randomly within a pre-specified area. As shown in Fig. 6.4(a) and Fig. 6.4(b), after the transient state the mobile robotic sensors constitute the desired formation and they preserve this formation as they move along the x-axis. It is obvious that the communication range should be chosen large enough so that the mobile robotic sensors form a connected graph during the simulations.

The next set of simulations show the ability of the proposed formation building strategy to work at different conditions. Fig. 6.5(a) and Fig. 6.5(b) demonstrate the process of formation building for two different configurations with five and four mobile robotic sensors. As shown in Fig. 6.5(a) and Fig. 6.5(b), for both cases the mobile robotic sensors adjust their headings after a reasonable amount of time, and then they converge to the vertices of the given 3D configurations. The presented simulation results verify the effectiveness of the proposed decentralized formation building control law for various initial conditions, the number of mobile robotic sensors and configurations.

The second set of simulations is carried out to show the headings of the mobile robotic sensors in the whole formation building process. By application of the proposed decentralized navigation law, the robots heading angles converge to zero, because eventually all robots move in the same direction with the X-axis, as shown in Fig. 6.6. To illustrate the convergence of the mobile robotic sensors to the given configuration, in the following we introduce the errors term defined by: $e_{y_{ij}}(t) = (y_i(t) - y_j(t)) - (Y_i - Y_j)$ and $e_{z_{ij}} = (z_i(t) - z_j(t)) - (Z_i - Z_j)$. Fig. 6.7(a) and Fig. 6.7(b) show that $e_{y_{ij}}(t) \to 0$ and $e_{z_{ij}}(t) \to 0$, for all $i = 1, 2, \ldots, n$. It means that, eventually all mobile robotic sensors converge to the given configu-
Figure 6.5: Trajectories of mobile robotic sensors during 3D formation building
In the following, we simulate the proposed algorithm for the problem of formation building with anonymous robots. Fig. 6.8(a) and Fig. 6.8(b) show the performance of this algorithm using different initial conditions and configurations. As shown, in both cases the mobile robotic sensors converge to the given configuration.

6.5 Summary

At first part of this chapter, a decentralized control law for formation building in three dimensional environments was proposed. We used nonlinear standard kinematics equations with hard constraints on the robot angular and linear velocity to describe the robot motion in 3D spaces. Then, we presented a random formation building algorithm for the problem of formation building with anonymous robots in 3D environments. This algorithm is an appropriate option for the case where robots do not know a priori its position in the configuration. The proposed control laws are based on the consensus approach that is simply implemented and computationally effective. The proposed control algorithms are decentralized, and the control action of each robot is based on the local information of its neighbouring robots. Based on this algorithms, all mobile robotic sensors eventually converge to the desired geometric configuration with the same direction and the same speed. The performance of the proposed decentralized control laws have been confirmed by extensive simulations. Furthermore, we give a mathematically rigorous analysis of the convergence of the mobile robotic sensors to the given configurations.
Figure 6.7: Convergence of the mobile robotic sensors to the given configuration
Figure 6.8: Formation building with anonymous robots
Chapter 7

Conclusions and Future Work

Contents

7.1 Future Work . . . . . . . . . . . . . . . . . . . . . . . . 88

This report considers the problem of coverage, search and formation building by a network of mobile robotic sensors in three dimensional spaces. Unlike most of the existing algorithms which focus on two dimensional environments, our algorithms are designed for three dimensional environments. In this report, we present some novel decentralized control algorithms for search, coverage and formation building in 3D spaces. The decentralized control of mobile robotic sensor networks is a relatively recent area of research, and this approach is believed more promising due to many inevitable physical constraints such as limited resources and energy, short wireless communication ranges, narrow bandwidths, and large sizes of robots to manage and control. Furthermore, the distributed method has many advantages in realizing cooperative group performances, scalability, and robustness.

In the following, we summarize the important contributions of this report for self deployment of mobile robotic sensors networks for coverage, search and formation building in 3D environments.

- We introduced a distributed motion coordination algorithm for a network of mobile robotic sensors for complete sensing coverage of a bounded three-dimensional space. The algorithm was developed based on some simple consensus laws that only require information about the closest neighbours of each mobile robotic sensor. The proposed control laws drive the network of mobile robotic sensors to form a 3D covering grid in the target region and then they occupy the vertices of the grid. Also, we proposed a distributed control law for coordination of the mobile robotic sensors such that they form a given 3D shape at vertices of a truncated octahedral grid from any initial positions. Simulation results showed that the 3D truncated octahedral grid outperforms other 3D grids in terms of complete sensing coverage time and a minimum number of mobile robotic sensors required to cover completely the given 3D region.
• We presented a set of random distributed laws for search in unknown bounded three dimensional environments. The mobile sensors utilize a truncated octahedral grid for the search process. Furthermore, they exchange information with their neighbouring sensors to minimize the search time.

• We introduced a novel random bio-inspired search algorithm to drive mobile robotic sensors for locating clustered and sparsely located targets in bounded 3D areas. The proposed method combines the bio-inspired Levy flight random search mechanism for determining the length of the walk with vertices of a covering truncated octahedral grid to optimize the search procedure. The proposed approach has the advantage that it does not need centralized control system, also it is scalable.

• We extended our grid-based random search algorithm for the problem of detecting mobile targets moving randomly in a bounded 3D space by a mobile robotic sensor network.

• We presented a decentralized control law for formation building in 3D environments. We used nonlinear standard kinematics equations with hard constraints on the robot angular and linear velocity to describe the robots’ motion in 3D spaces. Then, we proposed a random formation building algorithm for the problem of formation building with anonymous robots in 3D environments. This algorithm is an appropriate option for the case where robots do not know a priori its position in the configuration.

• The proposed algorithms are distributed. As a result, these algorithms can be used to the cases where the communication and the sensing range are limited. Moreover, The proposed control laws are based on the consensus approach that is simply implemented and computationally effective. The control algorithms are decentralized and the control action of each robot is based on the local information of its neighbouring robots.

• The proposed methods can deal with new tasks in various environments, such as different number of targets or mobile sensors. In other words, these algorithms are flexible in various scenarios.

• Finally, the effectiveness of the proposed control algorithms have been confirmed by extensive simulations. Also, we gave mathematically rigorous proof of convergence with probability 1 of the proposed algorithms.

7.1 Future Work
The work in this report opens up some future research problems and the practical realization of the proposed algorithms. The following suggestions are examples of possible future research based on the results presented in this report:
• In the future studies, the problem of navigating mobile robotic sensors with complicated models can be considered. In such cases, it is not sufficient to solve the navigation problem by only considering the kinematic models of the mobile robotic sensors, the dynamics of the robots should also be studied. The model proposed in the current works is simple. Future research could consider a more complex nonlinear version of the model [85,140,142]. In such a model, dynamic equations for mobile robots should be applied. Moreover, we will combine our proposed navigation method with methods of robust nonlinear control and robust state estimation proposed in [104,121,132,141].

• In the future search and coverage research, we will consider more realistic scenarios with robotic sensors motion described by non-holonomic kinematic models with the problem of collision avoidance [70,146,155,184].

• Current works assumed homogeneous sensors as well as spherical detection ranges of those sensors. These assumptions are simplistic. To extend this work, we consider nodes with more realistic features.

• Another exciting direction is to combine the proposed formation building algorithms with a local obstacle avoidance technique to generate safe and applicable strategies in the real world so that the mobile robots will not collide with each other and to obstacles during formation building. See e.g. [31,40,71,99,102,106,139,147,152,160,164,184].

• In the last chapter, we proposed a decentralized formation building control law for a group of mobile robots with hard constraints on the linear and angular velocities, and we presented their computer simulations. It would be interesting to investigate and verify the effectiveness of the proposed algorithms through the experiments with a group of real mobile robots.

• In the future works, we can evaluate the proposed distributed algorithms based on the factors such as cost, power consumption, etc.

• To use mobile robotic sensors in search, surveillance, monitoring and exploration in future applications, it is definitely necessary to move on real mobile robotic sensors and implement solutions using realistic scenarios. Moreover, we can use the 3D mobile robotic sensor networks for environmental exploration of 2D and 3D environmental fields and surfaces. See, e.g. [50,100,101].

• Further research is needed to optimize the travelling distance of the mobile robotic sensor networks as it can effectively reduce the energy consumption and prolong the network lifetime.

• In chapter 2,3,4,5, we consider the problem of decentralized search and coverage in 3D spaces. It is possible to combine the proposed algorithms with a collision avoidance technique in order to avoid any collision with nearby moving agents. See e.g. [29,158].
• In our proposed algorithms, we didn’t take into account limitations of communication channels. Future research is needed to combine our proposed navigation method with the tools for control and estimation via limited capacity channels developed in [95–98,134].
Bibliography

[1] E. U. Acar, H. Choset, Y. Zhang, and M. Schervish. Path planning for robotic demining: Robust sensor-based coverage of unstructured environments and probabilistic methods. *The International journal of robotics research*, 22(7-8):441–466, 2003.

[2] K. Akkaya and A. Newell. Self-deployment of sensors for maximized coverage in underwater acoustic sensor networks. *Computer Communications*, 32(7):1233–1244, 2009.

[3] F. M. Al-Turjman, H. S. Hassanein, and M. A. Ibnkahla. Efficient deployment of wireless sensor networks targeting environment monitoring applications. *Computer Communications*, 36(2):135–148, 2013.

[4] S. Alam and Z. J. Haas. Coverage and connectivity in three-dimensional networks. In *Proceedings of the 12th annual international conference on Mobile computing and networking*, pages 346–357. ACM, 2006.

[5] S. Alam and Z. J. Haas. Coverage and connectivity in three-dimensional underwater sensor networks. *Wireless Communications and Mobile Computing*, 8(8):995–1009, 2008.

[6] S. Alam and Z. J. Haas. Coverage and connectivity in three-dimensional underwater sensor networks. *Wireless Communications and Mobile Computing*, 8(8):995–1009, 2008.

[7] S. N. Alam and Z. J. Haas. Coverage and connectivity in three-dimensional networks with random node deployment. *Ad Hoc Networks*, 34:157–169, 2015.

[8] M. Anderson and N. Papanikolopoulos. Implicit cooperation strategies for multi-robot search of unknown areas. *Journal of Intelligent and Robotic Systems*, 53(4):381–397, 2008.

[9] G. Antonelli and G. Antonelli. *Underwater robots*. Springer, 2014.

[10] X. Bai, Z. Yun, D. Xuan, T. Lai, and W. Jia. Optimal patterns for four-connectivity and full coverage in wireless sensor networks. *IEEE Transactions on Mobile Computing*, 9(3):435–448, March 2010.
[11] X. Bai, C. Zhang, D. Xuan, and W. Jia. Full-coverage and k-connectivity (k= 14, 6) three dimensional networks. In INFOCOM 2009, IEEE, pages 388–396. IEEE, 2009.

[12] X. Bai, C. Zhang, D. Xuan, J. Teng, and W. Jia. Low-connectivity and full-coverage three dimensional wireless sensor networks. In Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing, pages 145–154. ACM, 2009.

[13] L. Bakule. Decentralized control: An overview. Annual reviews in control, 32(1):87–98, 2008.

[14] A. Baranzadeh. A decentralized control algorithm for target search by a multi-robot team. In Proceedings of Australasian Conference of Robotics and Automation, Sydney, Australia, 2013.

[15] A. Baranzadeh and V. Nazarzehi. A decentralized formation building algorithm with obstacle avoidance for multi-robot systems. In 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO), pages 2513–2518. IEEE, 2015.

[16] A. Baranzadeh and V. Nazarzehi. Distributed formation building with obstacle avoidance for a team of wheeled mobile robots. In Australasian Conference on Robotics and Automation. ACRA2015, 2015.

[17] J. C. Barca and Y. A. Sekercioglu. Swarm robotics reviewed. Robotica, 31(03):345–359, 2013.

[18] S. Barr, B. Liu, and J. Wang. Underwater sensor barriers with auction algorithms. In Proceedings of 18th International Conference on Computer Communications and Networks, pages 1–6. IEEE, 2009.

[19] S. J. Barr, J. Wang, and B. Liu. An efficient method for constructing underwater sensor barriers. Journal of Communications, 6(5):370–383, 2011.

[20] N. Bartolini, G. Bongiovanni, T. La Porta, and S. Silvestri. On the vulnerabilities of the virtual force approach to mobile sensor deployment. IEEE Transactions on Mobile Computing, 13(11):2592–2605, 2014.

[21] J. L. Baxter, E. Burke, J. M. Garibaldi, and M. Norman. Multi-robot search and rescue: A potential field based approach. In Autonomous robots and agents, pages 9–16. Springer, 2007.

[22] F. Belkhouché, S. Vadhva, and M. Vaziri. Modeling and controlling 3d formations and flocking behavior of unmanned air vehicles. In 2011 IEEE International Conference on Information Reuse and Integration (IRI), pages 449–454. IEEE, 2011.

[23] S. Benhamou. How many animals really do the levy walk? Ecology, 88(8):1962–1969, 2007.
[24] M. Bernard, K. Kondak, I. Maza, and A. Ollero. Autonomous transportation and deployment with aerial robots for search and rescue missions. *Journal of Field Robotics*, 28(6):914–931, 2011.

[25] E. Borhaug, A. Pavlov, and K. Pettersen. Cross-track formation control of underactuated autonomous underwater vehicles. In *Group Coordination and Cooperative Control*, pages 35–54. Springer, 2006.

[26] E. Borhaug, A. Pavlov, and K. Y. Pettersen. Straight line path following for formations of underactuated underwater vehicles. In *IEEE Conference on Decision and Control*, pages 2905–2912, 2007.

[27] E. Borhaug, A. Pavlov, and K. Y. Pettersen. Straight line path following for formations of underactuated underwater vehicles. In *46th IEEE Conference on Decision and Control*, pages 2905–2912. IEEE, 2007.

[28] N. Boufares, I. Khoufi, P. Minet, L. Saidane, and Y. Ben Saied. Three dimensional mobile wireless sensor networks redeployment based on virtual forces. In *Wireless Communications and Mobile Computing Conference*, pages 563–568. IEEE, 2015.

[29] A. Breitenmoser and A. Martinoli. On combining multi-robot coverage and reciprocal collision avoidance. In *Distributed Autonomous Robotic Systems*, pages 49–64. Springer, 2016.

[30] T. Bretl and S. Hutchinson. Robust coverage by a mobile robot of a planar workspace. In *2013 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4582–4587. IEEE, 2013.

[31] R. Brockers, Y. Kuwata, S. Weiss, and L. Matthies. Micro air vehicle autonomous obstacle avoidance from stereo-vision. In *SPIE Defense+ Security*, pages 90840O–90840O. International Society for Optics and Photonics, 2014.

[32] F. Bullo, J. Cortés, and S. Martinez. *Distributed control of robotic networks: a mathematical approach to motion coordination algorithms*. Princeton University Press, 2009.

[33] W. Burgard, M. Moors, C. Stachniss, and F. E. Schneider. Coordinated multi-robot exploration. *IEEE Transactions on Robotics*, 21(3):376–386, 2005.

[34] L. Buşoniu and L. Tamás. Handling uncertainty and networked structure in robot control. 2015.

[35] Y. Cai and S. X. Yang. An improved pso-based approach with dynamic parameter tuning for cooperative multi-robot target searching in complex unknown environments. *International Journal of Control*, 86(10):1720–1732, 2013.
[36] M. Cao, D. A. Spielman, and A. S. Morse. A lower bound on convergence of a distributed network consensus algorithm. In 44th IEEE Conference on Decision and Control, and the European Control Conference, pages 2356–2361, 2005.

[37] Y. Cao, W. Yu, W. Ren, and G. Chen. An overview of recent progress in the study of distributed multi-agent coordination. IEEE Transactions on Industrial Informatics, 9(1):427–438, 2013.

[38] D. W. Casbeer, R. W. Beard, T. W. McLain, S.-M. Li, and R. K. Mehra. Forest fire monitoring with multiple small uavs. In American Control Conference, pages 3530–3535. IEEE, 2005.

[39] O. Cayirpunar, V. Gazi, B. Tavli, et al. Experimental study on the effects of communication on cooperative search in complex environments. In In Proceedings of the International Conference on Robot Communication and Coordination, pages 1–6. Woodhead Publishing, 2009.

[40] K. Chang, Y. Xia, and K. Huang. Coordinated formation control design with obstacle avoidance in three-dimensional space. Journal of the Franklin Institute, 352(12):5779–5795, 2015.

[41] A. Chen, S. Kumar, and T. H. Lai. Local barrier coverage in wireless sensor networks. IEEE Transactions on Mobile Computing, 9(4):491–504, 2010.

[42] Y. Q. Chen and Z. Wang. Formation control: a review and a new consideration. In International Conference on Intelligent Robots and Systems, pages 3181–3186. IEEE, 2005.

[43] T. M. Cheng and A. V. Savkin. A distributed self-deployment algorithm for the coverage of mobile wireless sensor networks. IEEE Communication Letters, 13(11):877–879, 2009.

[44] T. M. Cheng and A. V. Savkin. Decentralized control for mobile robotic sensor network self-deployment: barrier and sweep coverage problems. Robotica, 29(02):283–294, 2011.

[45] T. M. Cheng and A. V. Savkin. Self-deployment of mobile robotic sensor networks for multilevel barrier coverage. Robotica, 30(04):661–669, 2012.

[46] T. M. Cheng and A. V. Savkin. Decentralized control of mobile sensor networks for asymptotically optimal blanket coverage between two boundaries. IEEE Transactions on Industrial Informatics, 9(1):365–376, 2013.

[47] T. M. Cheng, A. V. Savkin, and F. Javed. Decentralized control of a group of mobile robots for deployment in sweep coverage. Robotics and Autonomous Systems, 59(7):497–507, 2011.

[48] M. ChinnaDurai, S. MohanKumar, and S. Sharmila. Underwater wireless sensor networks. Compusoft, 4(7):1899, 2015.
[49] J. Clark and R. Fierro. Cooperative hybrid control of robotic sensors for perimeter detection and tracking. In *American Control Conference*, pages 3500–3505. IEEE, 2005.

[50] J. Clark and R. Fierro. Mobile robotic sensors for perimeter detection and tracking. *ISA transactions*, 46(1):3–13, 2007.

[51] J. Cortés and F. Bullo. Coordination and geometric optimization via distributed dynamical systems. *SIAM Journal on Control and Optimization*, 44(5):1543–1574, 2005.

[52] J. Cortes, S. Martinez, T. Karatas, and F. Bullo. Coverage control for mobile sensing networks. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1327–1332. IEEE, 2002.

[53] M. S. Couceiro. An overview of swarm robotics for search and rescue applications. *Handbook of Research on Design, Control, and Modeling of Swarm Robotics*, page 345, 2015.

[54] A. Datta and S. Soundaralakshmi. On-line path planning in an unknown polygonal environment. *Information Sciences*, 164(1):89–111, 2004.

[55] D. V. Dimarogonas and K. J. Kyriakopoulos. Decentralized navigation functions for multiple robotic agents with limited sensing capabilities. *Journal of Intelligent and Robotic Systems*, 48(3):411–433, 2007.

[56] H. Dong, K. Zhang, and L. Zhu. An algorithm of 3d directional sensor network coverage enhancing based on artificial fish-swarm optimization. In *International Workshop on Microwave and Millimeter Wave Circuits and System Technology (MMWCST)*, pages 1–4. IEEE, 2012.

[57] M. F. Ercan, X. Li, and X. Liang. A regular tetrahedron formation strategy for swarm robots in three-dimensional environment. In *Hybrid Artificial Intelligence Systems*, pages 24–31. Springer, 2010.

[58] N. Farmani, L. Sun, and D. Pack. Tracking multiple mobile targets using cooperative unmanned aerial vehicles. In *International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 395–400. IEEE, 2015.

[59] L. Feng, T. Qiu, Z. Sun, F. Xia, and Y. Zhou. A coverage strategy for wireless sensor networks in a three-dimensional environment. *International Journal of Ad Hoc and Ubiquitous Computing*, 15(1):83–94, 2014.

[60] A. Flenner, J. Flenner, J. Bobinchak, D. Mercier, A. Le, K. Estabridis, and G. Hewer. Levy walks for autonomous search. volume 8389, pages 83890Z–83890Z–10, 2012.

[61] G. M. Fricke, F. Asperti-Boursin, J. Hecker, J. Cannon, and M. Moses. From microbiology to microcontrollers: Robot search patterns inspired by
t cell movement. In *Advances in Artificial Life, ECAL*, volume 12, pages 1009–1016, 2013.

[62] G. M. Fricke, S. R. Black, J. P. Hecker, J. L. Cannon, and M. E. Moses. Distinguishing adaptive search from random search in robots and t cells. In *Proceedings of the 2015 on Genetic and Evolutionary Computation Conference*, pages 105–112. ACM, 2015.

[63] D. W. Gage. Command control for many-robot systems. Technical report, DTIC Document, 1992.

[64] M. Ghaemi, Z. Zabihinpour, and Y. Asgari. Computer simulation study of the levy flight process. *Physica A: Statistical Mechanics and its Applications*, 388(8):1509–1514, 2009.

[65] A. Gkikopoulou, G. Nikolakopoulos, and S. Manesis. A survey on underwater wireless sensor networks and applications. In *Mediterranean Conference on Control & Automation (MED)*, pages 1147–1154. IEEE, 2012.

[66] S. Grzonka, G. Grisetti, and W. Burgard. Towards a navigation system for autonomous indoor flying. In *IEEE International Conference on Robotics and Automation*, pages 2878–2883. IEEE, 2009.

[67] Y. Gu, D. Bozdağ, R. W. Brewer, and E. Ekici. Data harvesting with mobile elements in wireless sensor networks. *Computer Networks*, 50(17):3449–3465, 2006.

[68] J. Heidemann, M. Stojanovic, and M. Zorzi. Underwater sensor networks: applications, advances and challenges. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 370(1958):158–175, 2012.

[69] J. M. Hereford and M. A. Siebold. *Bio-inspired search strategies for robot swarms*. INTECH Open Access Publisher, 2010.

[70] M. Hoy, A. S. Matveev, and A. V. Savkin. Algorithms for collision-free navigation of mobile robots in complex cluttered environments: a survey. *Robotica*, 33(03):463–497, 2015.

[71] S. Hrabar. An evaluation of stereo and laser-based range sensing for rotorcraft unmanned aerial vehicle obstacle avoidance. *Journal of Field Robotics*, 29(2):215–239, 2012.

[72] C.-F. Huang, Y.-C. Tseng, and L.-C. Lo. The coverage problem in three-dimensional wireless sensor networks. *Journal of Interconnection Networks*, 8(03):209–227, 2007.

[73] K. Ijaz and U. Manzoor. Using vision and coordination to find unknown target in fixed and random length obstacles. *WSEAS Transactions on Computers*, 5(10):2400–2405, 2006.
[74] Z. H. Ismail, N. Sarman, and M. Dunnigan. Dynamic region boundary-based control scheme for multiple autonomous underwater vehicles. In OCEANS, 2012-Yeosu, pages 1–6. IEEE, 2012.

[75] A. Jadbabaie, J. Lin, and A. S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. IEEE Transactions on Automatic Control, 48(6):988–1001, 2003.

[76] P. Jalalkamali. Distributed Tracking and Information-Driven Control for Mobile Sensor Networks. Dartmouth College, 2013.

[77] A. Jeremic and A. Nehorai. Design of chemical sensor arrays for monitoring disposal sites on the ocean floor. IEEE Journal of Oceanic Engineering, 23(4):334–343, 1998.

[78] B. Jouvencel, O. Parodi, X. Xiang, et al. Coordinated formation control of multiple autonomous underwater vehicles for pipeline inspection. International Journal of Advanced Robotic Systems, 7(1):75–84, 2010.

[79] B. Jouvencel, X. Xiang, L. Zhang, Z. Fang, et al. 3d reconstruction of seabed surface through sonar data of auvs. Indian Journal of Geo-Marine Sciences (IJMS), 41(6):509–515, 2012.

[80] S. Kalantar and U. R. Zimmer. Distributed shape control of homogeneous swarms of autonomous underwater vehicles. Autonomous Robots, 22(1):37–53, 2007.

[81] M. Keeter, D. Moore, R. Muller, E. Nieters, J. Flenner, S. E. Martonosi, A. L. Bertozzi, A. G. Percus, and R. Levy. Cooperative search with autonomous vehicles in a 3d aquatic testbed. In American Control Conference (ACC), 2012, pages 3154–3160. IEEE, 2012.

[82] K. K. Khedo, R. Perseedoss, A. Mungur, et al. A wireless sensor network air pollution monitoring system. International Journal of Wireless and Mobile Networks, 2(2):31–45, 2010.

[83] B. O. Koopman. Search and its optimization. The American Mathematical Monthly, 86(7):527–540, 1979.

[84] N. Kottege and U. R. Zimmer. Underwater acoustic localization for small submersibles. Journal of Field Robotics, 28(1):40–69, 2011.

[85] M. Krstic, P. V. Kokotovic, and I. Kanellakopoulos. Nonlinear and adaptive control design. John Wiley & Sons, Inc., 1995.

[86] K. W. Lin, M.-H. Hsieh, and V. S. Tseng. A novel prediction-based strategy for object tracking in sensor networks by mining seamless temporal movement patterns. Expert Systems with Applications, 37(4):2799–2807, 2010.
[87] K. D. Listmann, M. V. Masalawala, and J. Adamy. Consensus for formation control of nonholonomic mobile robots. In IEEE International Conference on Robotics and Automation, pages 3886–3891. IEEE, 2009.

[88] B. Liu, O. Dousse, P. Nain, and D. Towsley. Dynamic coverage of mobile sensor networks. IEEE Transactions on Parallel and Distributed Systems, 24(2):301–311, 2013.

[89] J. Liu, Z. Wang, M. Zuba, Z. Peng, J.-H. Cui, and S. Zhou. Da-sync: A doppler-assisted time-synchronization scheme for mobile underwater sensor networks. IEEE Transactions on Mobile Computing, 13(3):582–595, March 2014.

[90] N. A. Lynch. Distributed algorithms. Morgan Kaufmann, 1996.

[91] U. Mansoor and H. M. Ammari. Coverage and connectivity in 3d wireless sensor networks. In The Art of Wireless Sensor Networks, pages 273–324. Springer, 2014.

[92] S. Mansour, N. Nasser, L. Karim, and A. Ali. Wireless sensor network-based air quality monitoring system. In International Conference on Computing, Networking and Communications (ICNC), pages 545–550. IEEE, 2014.

[93] A. S. Matveev, M. C. Hoy, and A. V. Savkin. 3d environmental extremum seeking navigation of a nonholonomic mobile robot. Automatica, 50(7):1802–1815, 2014.

[94] A. S. Matveev and A. V. Savkin. Qualitative Theory of Hybrid Dynamical Systems. Birkhauser Boston, 2000.

[95] A. S. Matveev and A. V. Savkin. Optimal state estimation in networked systems with asynchronous communication channels and switched sensors. In Proceedings of the 40th IEEE Conference on Decision and Control, volume 1, pages 825–830. IEEE, 2001.

[96] A. S. Matveev and A. V. Savkin. The problem of state estimation via asynchronous communication channels with irregular transmission times. IEEE Transactions on Automatic Control, 48(4):670–676, 2003.

[97] A. S. Matveev and A. V. Savkin. Comments on” control over noisy channels” and relevant negative results. IEEE Transactions on Automatic Control, 50(12):2105–2110, 2005.

[98] A. S. Matveev and A. V. Savkin. Estimation and control over communication networks. Birkhauser, Boston, 2009.

[99] A. S. Matveev, H. Teimoori, and A. V. Savkin. A method for guidance and control of an autonomous vehicle in problems of border patrolling and obstacle avoidance. Automatica, 47(3):515–524, 2011.
[100] A. S. Matveev, H. Teimoori, and A. V. Savkin. Navigation of a unicycle-like mobile robot for environmental extremum seeking. *Automatica*, 47(1):85–91, 2011.

[101] A. S. Matveev, H. Teimoori, and A. V. Savkin. Method for tracking of environmental level sets by a unicycle-like vehicle. *Automatica*, 48(9):2252–2261, 2012.

[102] A. S. Matveev, C. Wang, and A. V. Savkin. Real-time navigation of mobile robots in problems of border patrolling and avoiding collisions with moving and deforming obstacles. *Robotics and Autonomous systems*, 60(6):769–788, 2012.

[103] C. Miao, G. Dai, X.-m. Zhao, Z. Tang, and Q. Chen. 3d self-deployment algorithm in mobile wireless sensor networks. In *Advances in Wireless Sensor Networks*, pages 27–41. Springer, 2014.

[104] S. R. Moheimani, A. V. Savkin, and I. R. Petersen. Robust filtering, prediction, smoothing, and observability of uncertain systems. *IEEE Transactions on Circuits and Systems I Fundamental Theory and Applications*, 45(4):446–457, 1998.

[105] N. Moshtagh and A. Jadbabaie. Distributed geodesic control laws for flocking of nonholonomic agents. *IEEE Transactions on Automatic Control*, 52(4):681–686, 2007.

[106] F. Munoz, E. Espinoza, I. Gonzalez, L. Garcia Carrillo, S. Salazar, and R. Lozano. A uas obstacle avoidance strategy based on spiral trajectory tracking. In *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 593–600. IEEE, 2015.

[107] V. Nazarzehi and A. Baranzadeh. A decentralized grid-based random search algorithm for locating targets in three dimensional environments by a mobile robotic network. In *Australasian Conference on Robotics and Automation*. ACRA2015, 2015.

[108] V. Nazarzehi and A. Baranzadeh. A distributed bio-inspired algorithm for search of moving targets in three dimensional spaces. In *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 2507–2512. IEEE, 2015.

[109] V. Nazarzehi and A. V. Savkin. Decentralized control of mobile three-dimensional sensor networks for complete coverage self-deployment and forming specific shapes. In *IEEE Conference on Control Applications*, pages 127–132. IEEE, 2015.

[110] V. Nazarzehi and A. V. Savkin. Decentralized three dimensional formation building algorithms for a team of nonholonomic mobile robots”. *Submitted to Journal of Intelligent and Robotic Systems*, 2015.
[111] V. Nazarzehi and A. V. Savkin. Distributed self-deployment of mobile wire-
less 3d robotic sensor networks for complete sensing coverage and forming
specific shapes. Submitted to Robotica, 2016.

[112] V. Nazarzehi, A. V. Savkin, and A. Baranzadeh. Distributed 3d dynamic
search coverage for mobile wireless sensor networks. IEEE Communications
Letters, 19(4):633–636, 2015.

[113] V. Nazarzehi had and A. V. Savkin. Decentralized navigation of nonholo-
nomic robots for 3d formation building. In The 2014 IEEE International
Conference on Robotics and Biomimetics. IEEE, 2014.

[114] S. G. Nurzaman, Y. Matsumoto, Y. Nakamura, S. Koizumi, and H. Ishig-
uro. Biologically inspired adaptive mobile robot search with and without
gradient sensing. In International Conference on Intelligent Robots and
Systems, pages 142–147. IEEE, 2009.

[115] S. G. Nurzaman, Y. Matsumoto, Y. Nakamura, S. Koizumi, and H. Ishig-
uro. Yuragi-based adaptive searching behavior in mobile robot: From
bacterial chemotaxis to levy walk. In IEEE International Conference on
Robotics and Biomimetics, pages 806–811. IEEE, 2009.

[116] R. Olfati-Saber, A. Fax, and R. M. Murray. Consensus and cooperation in
networked multi-agent systems. Proceedings of the IEEE, 95(1):215–233,
2007.

[117] M. Ou, H. Du, and S. Li. Finite-time formation control of multiple non-
holonomic mobile robots. International Journal of Robust and Nonlinear
Control, 24(1):140–165, 2014.

[118] J. Partan, J. Kurose, and B. N. Levine. A survey of practical issues in
underwater networks. ACM SIGMOBILE Mobile Computing and Commu-
nications Review, 11(4):23–33, 2007.

[119] J. Penders, L. Alboul, U. Witkowski, A. Naghsh, J. Saez-Pons, S. Her-
brechtsmeier, and M. El-Habbal. A robot swarm assisting a human fire-
fighter. Advanced Robotics, 25(1-2):93–117, 2011.

[120] Z. Peng, G. Wen, A. Rahmani, and Y. Yu. Distributed consensus-based
formation control for multiple nonholonomic mobile robots with a specified
reference trajectory. International Journal of Systems Science, pages 1–11,
2013.

[121] I. R. Petersen and A. V. Savkin. Robust Kalman filtering for signals and
systems with large uncertainties. Springer Science & Business Media, 1999.

[122] M. Plank and A. James. Optimal foraging: Levy pattern or process?
Journal of The Royal Society Interface, 5(26):1077–1086, 2008.
[123] D. Pompili, T. Melodia, and I. F. Akyildiz. Deployment analysis in underwater acoustic wireless sensor networks. In Proceedings of the 1st ACM international workshop on Underwater networks, pages 48–55. ACM, 2006.

[124] D. Pompili, T. Melodia, and I. F. Akyildiz. Three-dimensional and two-dimensional deployment analysis for underwater acoustic sensor networks. Ad Hoc Networks, 7(4):778–790, 2009.

[125] J. Qi, R. Vazquez, and M. Krstic. Multi-agent deployment in 3-d via pde control. IEEE Transactions on Automatic Control, 60(4):891–906, 2015.

[126] P. Rawat, K. D. Singh, H. Chaouchi, and J. M. Bonnin. Wireless sensor networks: a survey on recent developments and potential synergies. The Journal of supercomputing, 68(1):1–48, 2014.

[127] W. Ren, R. W. Beard, and E. M. Atkins. A survey of consensus problems in multi-agent coordination. In Proceedings of the 2005 American Control Conference., pages 1859–1864. IEEE, 2005.

[128] W. Ren, R. W. Beard, et al. Consensus seeking in multiagent systems under dynamically changing interaction topologies. IEEE Transactions on automatic control, 50(5):655–661, 2005.

[129] A. M. Reynolds and C. J. Rhodes. The lévy flight paradigm: random search patterns and mechanisms. Ecology, 90(4):877–887, 2009.

[130] G. Roussos, D. V. Dimarogonas, and K. J. Kyriakopoulos. 3d navigation and collision avoidance for nonholonomic aircraft-like vehicles. International Journal of Adaptive Control and Signal Processing, 24(10):900–920, 2010.

[131] A. Sangwan and R. P. Singh. Survey on coverage problems in wireless sensor networks. Wireless Personal Communications, 80(4):1475–1500, 2015.

[132] A. Savkin, I. R. Peterson, and V. Ugronovskii. Robust control design using h-infinity methods, 2000.

[133] A. V. Savkin. Coordinated collective motion of groups of autonomous mobile robots: analysis of Vicsek’s model. IEEE Transactions on Automatic Control, 49(6):981–983, 2004.

[134] A. V. Savkin. Analysis and synthesis of networked control systems: Topological entropy, observability, robustness and optimal control. Automatica, 42(1):51–62, 2006.

[135] A. V. Savkin, T. M. Cheng, Z. Li, F. Javed, A. S. Matveev, and H. Nguyen. Decentralized Coverage Control Problems For Mobile Robotic Sensor and Actuator Networks. John Wiley & Sons, 2015.
[136] A. V. Savkin and R. J. Evans. *Hybrid dynamical systems: controller and sensor switching problems*. Birkhauser Boston, 2002.

[137] A. V. Savkin and F. Javed. A method for decentralized self-deployment of a mobile sensor network with given regular geometric patterns. In *International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pages 371–376. IEEE, 2011.

[138] A. V. Savkin, F. Javed, and A. S. Matveev. Optimal distributed blanket coverage self-deployment of mobile wireless sensor networks. *IEEE Communications Letters*, 16(6):949–951, 2012.

[139] A. V. Savkin, A. S. Matveev, M. Hoy, and C. Wang. *Safe Robot Navigation Among Moving and Steady Obstacles*. Elsevier, 2015.

[140] A. V. Savkin and I. R. Petersen. Minimax optimal control of uncertain systems with structured uncertainty. *International Journal of Robust and Nonlinear Control*, 5(2):119–137, 1995.

[141] A. V. Savkin and I. R. Petersen. Model validation for robust control of uncertain systems with an integral quadratic constraint. *Automatica*, 32(4):603–606, 1996.

[142] A. V. Savkin and I. R. Petersen. Robust state estimation and model validation for discrete-time uncertain systems with a deterministic description of noise and uncertainty. *Automatica*, 34(2):271–274, 1998.

[143] A. V. Savkin, I. R. Petersen, E. Skafidas, and R. J. Evans. Hybrid dynamical systems: robust control synthesis problems. *Systems & control letters*, 29(2):81–90, 1996.

[144] A. V. Savkin and H. Teimoori. Bearings-only guidance of a unicycle-like vehicle following a moving target with a smaller minimum turning radius. *IEEE Transactions on Automatic Control*, 55(10):2390–2395, 2010.

[145] A. V. Savkin and H. Teimoori. Decentralized navigation of groups of wheeled mobile robots with limited communication. *IEEE Transactions on Robotics*, 26(6):1099–1104, 2010.

[146] A. V. Savkin and C. Wang. A simple biologically inspired algorithm for collision-free navigation of a unicycle-like robot in dynamic environments with moving obstacles. *Robotica*, 31(06):993–1001, 2013.

[147] A. V. Savkin and C. Wang. Seeking a path through the crowd: Robot navigation in unknown dynamic environments with moving obstacles based on an integrated environment representation. *Robotics and Autonomous Systems*, 62(10):1568–1580, 2014.
[148] A. V. Savkin, C. Wang, A. Baranzadeh, Z. Xi, and H. T. Nguyen. A method for decentralized formation building for unicycle-like mobile robots. In 9th Asian Control Conference (ASCC), pages 1–5. IEEE, 2013.

[149] A. V. Savkin, C. Wang, A. Baranzadeh, Z. Xi, and H. T. Nguyen. Distributed formation building algorithms for groups of wheeled mobile robots. Robotics and Autonomous Systems, 75:463–474, 2016.

[150] L. Scardovi, N. E. Leonard, and R. Sepulchre. Stabilization of collective motion in three dimensions: A consensus approach. In 46th IEEE Conference on Decision and Control, pages 2931–2936. IEEE, 2007.

[151] M. Senouci, A. Mellouk, K. Asnoune, and F. Bouhidel. Movement-assisted sensor deployment algorithms: A survey and taxonomy. Communications Surveys Tutorials, IEEE, 17(4):2493–2510, 2015.

[152] D. H. Shim and S. Sastry. An evasive maneuvering algorithm for uavs in see-and-avoid situations. In American Control Conference, volume 3886, page 3891, 2007.

[153] E. Skafidas, R. J. Evans, A. V. Savkin, and I. R. Petersen. Stability results for switched controller systems. Automatica, 35(4):553–564, 1999.

[154] Y. Song, B. Wang, Z. Shi, K. Pattipati, and S. Gupta. Distributed algorithms for energy-efficient even self-deployment in mobile sensor networks. IEEE Transactions on Mobile Computing, 13(5):1035–1047, May 2014.

[155] T. J. Stastny, G. A. Garcia, and S. S. Keshmiri. Collision and obstacle avoidance in unmanned aerial systems using morphing potential field navigation and nonlinear model predictive control. Journal of Dynamic Systems, Measurement, and Control, 137(1):014503, 2015.

[156] D. W. Stephens and J. R. Krebs. Foraging theory. Princeton University Press, 1986.

[157] T. Stevens and T. H. Chung. Autonomous search and counter-targeting using levy search models. In IEEE International Conference on Robotics and Automation (ICRA), pages 3953–3960. IEEE, 2013.

[158] D. M. Stipanović, C. J. Tomlin, and C. Valicka. Collision free coverage control with multiple agents. In Robot Motion and Control, pages 259–272. Springer, 2012.

[159] T. Stirling, S. Wischmann, and D. Floreano. Energy-efficient indoor search by swarms of simulated flying robots without global information. Swarm Intelligence, 4(2):117–143, 2010.

[160] S. Subramanian, T. George, and A. Thondiyath. Real-time obstacle avoidance for an underactuated flat-fish type autonomous underwater vehicle in 3d space. International Journal of Robotics and Automation, 29(4), 2014.
[161] S. Susca, F. Bullo, and S. Martínez. Monitoring environmental boundaries with a robotic sensor network. *IEEE Transactions on Control Systems Technology*, 16(2):288–296, 2008.

[162] D. Sutanto, P. Levi, C. Moslinger, and M. Read. Collective-adaptive lévy flight for underwater multi-robot exploration. In *IEEE International Conference on Mechatronics and Automation*, pages 456–462. IEEE, 2013.

[163] D. K. Sutanto, S. Kernbach, P. Levi, V. Nepomnyashchikh, et al. Multi-robot searching algorithm using lévy flight and artificial potential field. In *IEEE International Workshop on Safety Security and Rescue Robotics (SSRR)*, pages 1–6. IEEE, 2010.

[164] H. Teimoori and A. V. Savkin. A biologically inspired method for robot navigation in a cluttered environment. *Robotica*, 28(5):637–648, 2010.

[165] H. Teimoori and A. V. Savkin. Equiangular navigation and guidance of a wheeled mobile robot based on range-only measurements. *Robotics and Autonomous Systems*, 58(2):203–215, 2010.

[166] W. Thomson. On the division of space with minimum partitional area. *Acta Mathematica*, 11(1-4):121–134, 1887.

[167] J. N. Tsitsiklis, D. P. Bertsekas, and M. Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. In *American Control Conference*, pages 484–489, 1984.

[168] M. Turpin, N. Michael, and V. Kumar. Decentralized formation control with variable shapes for aerial robots. In *2012 IEEE International Conference on Robotics and Automation (ICRA)*, pages 23–30. IEEE, 2012.

[169] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet. Novel type of phase transition in a system of self-driven particles. *Physical review letters*, 75(6):1226, 1995.

[170] G. Viswanathan, V. Afanasyev, S. V. Buldyrev, S. Havlin, M. Da Luz, E. Raposo, and H. E. Stanley. Lévy flights in random searches. *Physica A: Statistical Mechanics and its Applications*, 282(1):1–12, 2000.

[171] G. Viswanathan, F. Bartumeus, S. V. Buldyrev, J. Catalan, U. Fulco, S. Havlin, M. Da Luz, M. Lyra, E. Raposo, and H. E. Stanley. Lévy flight random searches in biological phenomena. *Physica A: Statistical Mechanics and Its Applications*, 314(1):208–213, 2002.

[172] G. Viswanathan, E. Raposo, and M. Da Luz. Lévy flights and superdiffusion in the context of biological encounters and random searches. *Physics of Life Reviews*, 5(3):133–150, 2008.
[173] Q. Wang, H. Gao, F. Alsaadi, and T. Hayat. An overview of consensus problems in constrained multi-agent coordination. *Systems Science & Control Engineering: An Open Access Journal*, 2(1):275–284, 2014.

[174] Y. Wang, Y. Liu, and Z. Guo. Three-dimensional ocean sensor networks: a survey. *Journal of Ocean University of China*, 11(4):436–450, 2012.

[175] Y. Wang, R. Tan, G. Xing, X. Tan, J. Wang, and R. Zhou. Spatiotemporal aquatic field reconstruction using robotic sensor swarm. In *Real-Time Systems Symposium*, pages 205–214. IEEE, 2012.

[176] Y. Wang, X. Wang, B. Xie, D. Wang, and D. P. Agrawal. Intrusion detection in homogeneous and heterogeneous wireless sensor networks. *IEEE Transactions on Mobile Computing*, 7(6):698–711, 2008.

[177] Y. Wang, Y. Zhang, J. Liu, and R. Bhandari. Coverage, connectivity, and deployment in wireless sensor networks. In *Recent Development in Wireless Sensor and Ad-hoc Networks*, pages 25–44. Springer, 2015.

[178] M. K. Watfa. *Coverage issues in wireless sensor networks*. PhD thesis, University of Oklahoma, 2006.

[179] M. K. Watfa and S. Commuri. The 3-dimensional wireless sensor network coverage problem. In *Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control, 2006. ICNSC’06.*, pages 856–861. IEEE, 2006.

[180] F.-J. Wu, Y.-F. Kao, and Y.-C. Tseng. From wireless sensor networks towards cyber physical systems. *Pervasive and Mobile Computing*, 7(4):397–413, 2011.

[181] M. Yamashita, H. Umemoto, I. Suzuki, and T. Kameda. Searching for mobile intruders in a polygonal region by a group of mobile searchers. *Algorithmica*, 31(2):208–236, 2001.

[182] B. Yang, Y. Ding, Y. Jin, and K. Hao. Self-organized swarm robot for target search and trapping inspired by bacterial chemotaxis. *Robotics and Autonomous Systems*, 72:83–92, 2015.

[183] P. Yang, R. A. Freeman, and K. M. Lynch. Multi-agent coordination by decentralized estimation and control. *Automatic Control, IEEE Transactions on*, 53(11):2480–2496, 2008.

[184] X. Yang, L. M. Alvarez, and T. Bruggemann. A 3d collision avoidance strategy for uavs in a non-cooperative environment. *Journal of Intelligent & Robotic Systems*, 70(1-4):315–327, 2013.

[185] X.-S. Yang and S. Deb. Cuckoo search via lévy flights. In *World Congress on Nature & Biologically Inspired Computing*, pages 210–214. IEEE, 2009.
[186] W. Yintao and Y. Weisheng. Path following for multiple underactuated autonomous underwater vehicles with formation constraints. In 31st Chinese Control Conference (CCC), pages 6004–6009. IEEE, 2012.

[187] M. Younis and K. Akkaya. Strategies and techniques for node placement in wireless sensor networks: A survey. Ad Hoc Networks, 6(4):621–655, 2008.

[188] G. Zeng and X. Li. An artificial physics-based 3d swarm control strategy. In 2011 International Conference on Electric Information and Control Engineering (ICEICE), pages 148–151. IEEE, 2011.

[189] H. Zhang and J. Hou. Is deterministic deployment worse than random deployment for wireless sensor networks? In 25th IEEE International Conference on Computer Communications, pages 1–13, April 2006.

[190] S. Zhang, J. Yu, A. Zhang, L. Yang, and Y. Shu. Marine vehicle sensor network architecture and protocol designs for ocean observation. Sensors, 12(1):373–390, 2012.

[191] C. Zhu, C. Zheng, L. Shu, and G. Han. A survey on coverage and connectivity issues in wireless sensor networks. Journal of Network and Computer Applications, 35(2):619–632, 2012.

[192] Y. Zou and K. Chakrabarty. Sensor deployment and target localization based on virtual forces. In IEEE INFOCOM, volume 2, pages 1293–1303. IEEE, 2003.