A HYBRID MULTI-OBJECTIVE OPTIMISATION FOR ENERGY EFFICIENCY AND BETTER COVERAGE IN UNDERWATER WIRELESS SENSOR NETWORKS

SALMAH BINTI FATTAH

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
UNIVERSITI MALAYA
KUALA LUMPUR

2022

A HYBRID MULTI-OBJECTIVE OPTIMISATION FOR ENERGY EFFICIENCY AND BETTER COVERAGE IN UNDERWATER WIRELESS SENSOR NETWORKS

SALMAH BINTI FATTAH

THESIS SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
UNIVERSITI MALAYA
KUALA LUMPUR

2022

UNIVERSITI MALAYA

ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: SALMAH BINTI FATTAH

Registration/Matric No.: 17028285/1

Name of Degree: DOCTOR OF PHILOSOPHY

Title of Project Paper/Research Report/Dissertation/Thesis ("this Work"): A HYBRID MULTI-OBJECTIVE OPTIMISATION FOR ENERGY EFFICIENCY AND BETTER COVERAGE IN UNDERWATER WIRELESS SENSOR NETWORKS

Field of Study: SENSOR NETWORKS (COMPUTER SCIENCE)

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this Work;
(2) This work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every rights in the copyright to this Work to the University of Malaya ("UM"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature Date: 8/6/2022

Subscribed and solemnly declared before, Witness's Signature Date: 8 June 2022

Name:
Designation:

A HYBRID MULTI-OBJECTIVE OPTIMISATION FOR ENERGY EFFICIENCY AND BETTER COVERAGE IN UNDERWATER WIRELESS SENSOR NETWORKS

ABSTRACT

Underwater wireless sensor networks (UWSNs), which benefit ocean surveillance applications, marine monitoring and underwater target detection, have advanced substantially in recent years. However, existing deployment solutions do not satisfy the deployment of mobile underwater sensor nodes as a stochastic system. Internal and external environmental problems concern maximum coverage in the deployment region while minimising energy consumption. To fill this gap, this research proposes and implements a multi-objective optimisation solution to balance conflicts concerning node deployment objectives. First, this research analyses the existing mobile underwater node deployment algorithms to identify the significant problems in existing solutions. Next, it establishes the research problems by implementing various existing algorithms using comparative analysis. Based on that analysis, this research suggests a hybrid algorithm: the Multi-Objective Optimisation Genetic Algorithm based on Adaptive Multi-Parent Crossover and Fuzzy Dominance (MOGA-AMPazy). The method adapts the original Non-Dominated Sorting Genetic Algorithm II (NSGA-II) by introducing a hybridisation of adaptive multi-parent crossover genetic algorithm and fuzzy dominance-based decomposition techniques. The algorithm introduces the fuzzy Pareto dominance concept to compare two solutions and uses the scalar decomposition method when one solution cannot dominate the other in terms of the fuzzy dominance level. The solution also proposes adaptive multi-parent crossover (AMP) to balance exploration and exploitation with new offspring, changing the number of parents involved in the crossover based on the execution of

the new generation. The solution is further improved by introducing prospect theory to guarantee convergence through risk evaluation. The results obtained are then analysed to assess the proposed solution's performance in obtaining each deployment objective's optimal value. Finally, the proposed algorithm's effectiveness regarding node coverage, energy consumption, Pareto-optimal value, and algorithm execution time is validated using three Pareto-optimal metrics: including inverted generation distance (IGD), hypervolume, and diversity. Furthermore, this research utilises five commonly used two-objective ZDT test instances as benchmark tests, namely ZDT-1, ZDT-2, ZDT-3, ZDT-4, and ZDT-6. These tests use specific problem characteristics to impose the underlying proposed solution as well as three other systems. Pareto-optimal values obtained indicate that the proposed solution has almost complete coverage involving the actual Pareto front. Furthermore, all analysis and evaluation attributes indicate that the MOGA-AMPazy deployment algorithm can handle the multi-objective underwater sensor deployment problem better than other solutions. Thus, MOGA-AMPazy provides an efficient and comprehensive deployment solution for mobile sensor nodes in UWSNs. This study makes several noteworthy contributions to the body of knowledge concerning UWSNs, and it provides an excellent multi-objective representation to decision-makers or mission planners to monitor the region of interest (RoI).

Keywords: Mobile sensor nodes, Multi-objective optimisation, Ocean surveillance, Sensor deployment, Underwater wireless sensor networks.

HIBRID PENGOPTIMUMAN BERBILANG-OBJEKTIF UNTUK KECEKAPAN TENAGA DAN LIPUTAN DALAM RANGKAIAN SENSOR BAWAH AIR TANPA DAWAI

ABSTRAK

Rangkaian sensor tanpa dawai bawah laut (UWSNs), yang memanfaatkan aplikasi pengawasan lautan, pemantauan marin dan pengesanan sasaran bawah air, telah maju dengan ketara dalam beberapa tahun kebelakangan ini. Walau bagaimanapun, penyelesaian penyebaran sedia ada tidak memenuhi penggunaan nod sensor bawah air mudah alih sebagai sistem stokastik. Masalah persekitaran dalaman dan luaran melibatkan liputan maksimum di kawasan penempatan sambil meminimumkan penggunaan tenaga. Untuk mengisi jurang ini, penyelidikan ini mencadangkan dan melaksanakan penyelesaian pengoptimuman berbilang objektif untuk mengimbangi konflik mengenai objektif penyebaran nod. Pertama, penyelidikan ini menganalisis algoritma penyebaran nod bawah air mudah alih sedia ada untuk mengenal pasti masalah penting dalam penyelesaian sedia ada. Seterusnya, ia mewujudkan masalah kajian dengan melaksanakan pelbagai algoritma sedia ada menggunakan analisis perbandingan. Berdasarkan analisis itu, penyelidikan ini mencadangkan algoritma hibrid: Algoritma Genetik Pengoptimuman Pelbagai Objektif berdasarkan Penyesuaian Lintas Berbilang-Induk dan Kedominan Kabur (MOGA-AMPazy). Kaedah ini menyesuaikan Pengisihan Tak-Terdominan Algoritma Genetik II (NSGA-II) asal dengan memperkenalkan penghibridan Algoritma Genetik Pengoptimuman Pelbagai Objektif berdasarkan Penyesuaian Lintas Berbilang-Induk dan Kedominan Kabur. Algoritma ini memperkenalkan konsep penguasaan Pareto kabur untuk membandingkan dua penyelesaian dan menggunakan kaedah penguraian skalar apabila satu penyelesaian tidak dapat menguasai yang lain dari segi tahap penguasaan kabur.

Penyelesaian itu juga mencadangkan Penyesuaian Lintas Berbilang-Induk (AMP) untuk mengimbangi penerokaan dan eksploitasi dengan anak baharu, menukar bilangan induk yang terlibat dalam lintas berdasarkan pelaksanaan generasi baharu. Penyelesaian itu dipertingkatkan lagi dengan memperkenalkan teori prospek untuk menjamin penumpuan melalui penilaian risiko. Keputusan yang diperoleh kemudian dianalisis untuk menilai prestasi penyelesaian yang dicadangkan dalam mendapatkan nilai optimum setiap objektif penyebaran. Akhir sekali, keberkesanan algoritma yang dicadangkan berkenaan liputan nod, penggunaan tenaga, nilai Pareto-optimum dan masa pelaksanaan algoritma disahkan menggunakan tiga metrik Pareto-optimum: termasuk Penjanaan Jarak Tersongsang (IGD), Hiperisipadu dan Kepelbagaian. Selain itu, penyelidikan ini menggunakan lima contoh ujian ZDT dua objektif yang biasa digunakan sebagai ujian penanda aras, iaitu ZDT-1, ZDT-2, ZDT-3, ZDT-4 dan ZDT-6. Ujian ini menggunakan ciri masalah khusus untuk mengenakan penyelesaian cadangan asas serta tiga sistem lain. Nilai Pareto-optimum yang diperoleh menunjukkan bahawa penyelesaian yang dicadangkan mempunyai liputan hampir lengkap yang melibatkan bahagian hadapan Pareto sebenar. Tambahan pula, semua sifat analisis dan penilaian menunjukkan bahawa algoritma penyebaran MOGA-AMPazy boleh menangani masalah penyebaran sensor bawah air berbilang objektif dengan lebih baik daripada penyelesaian lain. Oleh itu, MOGA-AMPazy menyediakan penyelesaian penyebaran yang cekap dan komprehensif untuk nod sensor mudah alih dalam UWSN. Kajian ini memberikan beberapa sumbangan penting kepada badan pengetahuan mengenai UWSN, dan ia menyediakan perwakilan berbilang objektif yang sangat baik kepada pembuat keputusan atau perancang misi untuk memantau kawasan sekepentingan (RoI).

Katakunci: Nod sensor bergerak, Pengotimuman berbilang-objektif, Pengawasan lautan, Penyebaran sensor, Rangkaian sensor bawah air tanpa dawai.

ACKNOWLEDGEMENTS

First of all, I am thankful to Allah Almighty for endowing me the strength, wisdom, and endless blessings to study. I would like to sincerely express my deepest gratitude to my supervisor, Professor Dr. Abdullah Gani, and co-supervisor, Associate Prof. Dr. Mohd Yamani Idna Bin Idris, Dr. Ismail Bin Ahmedy for their invaluable guidance, supervision, and encouragement to me throughout this research. Their continuous guidance and support assisted me conducting a valuable piece of research reported in this thesis. They provided me the opportunity to broaden my professional experience and prepare me for future challenges. Throughout the course of this research, I was fortunate enough to meet wonderful people and receive a great deal of support from them while pursuing my PhD at University of Malaya. I would also like to express my sincerest gratitude and special appreciation to my husband, sons, parents and siblings for all of the sacrifices, endless love and support during my PhD journey. Without their moral support, this thesis would never have been completed. Finally, thank you to all the members and support staff at the Center of Mobile Cloud Computing, Wisma RnD and FSKTM for their support and resources.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

2D	Two-Dimensional
3D	Three-Dimensional
AI	Artificial intelligence
APF	Artificial Potential Field
ASVs	Autonomous Surface Vehicles
AUVs	Autonomous Underwater Vehicles
CG	Computational Geometry
DABVF	Distributed node deployment algorithm
DT	Delaunay Triangulation
FDDA	Fully Distributed Deployment Algorithm
IoT	Internet of Things
IP	Internet Protocol
M-UW	Mobile Underwater
M2M	Machine-to-Machine
PSO	Particle swarm optimization
ROVs	Remotely Operative Underwater Vehicles
SOFAR	Sound Fixing and Ranging
VD	Voronoi Diagram

VFRBEC	Virtual forces redeployment algorithm based on energy consumption

CHAPTER 1: INTRODUCTION

This chapter presents an overview of the research carried out in this thesis, and it is organised as follows: Section 1.1 discusses the background of mobile underwater wireless sensor node (UWSN) deployment, purposes and perspectives. Next, Section 1.2 describes the motivation for this research work. Section 1.3 states the problem concerning node coverage and energy consumption. The research objectives are presented in Section 1.4, and Section 1.5 provides an overview of the research methodology chosen to achieve the defined objectives. Finally, Section 1.6 outlines the structure of the thesis.

1.1 Background

This section presents an overview of UWSNs, mobile underwater (M-UW) sensor node deployment, and multi-objective optimisation to provide fundamental knowledge on the research domain. A UWSN is a network of intelligent sensors and instruments intended to interact collaboratively via wireless connections to monitor tasks across a particular area (Akyildiz, Pompili, & Melodia, 2005). UWSN architecture is categorised according to type, sensor node network capabilities, and the spatial coverage of the application.

The static sensor node is usually placed on the seafloor in a two-dimensional (2D) region. It uses multi-hop communication among several clusters to connect with a sink node for data transfer (Felemban, Shaikh, Qureshi, Sheikh, & Qaisar, 2015). A static three-dimensional (3D) design, on the other hand, uses inflatable buoys to attach the system to the bottom and deploys sensors at various depths by changing cable length. The sensor nodes in the mobile architecture are free to move around, thereby allowing for dynamic changes in network structure. The mobile node needs two transceivers to maximise network data collection efficiencies. Remotely operated underwater vehicles (ROVs) , autonomous underwater vehicles (AUVs), and sea gliders are all part of it.

However, (Mohamed, Hamza, & Saroit, 2017) emphasised that, due to changes in mobility characteristics, mobile node features require special attention to ensure complete network coverage. A hybrid architecture, on the other hand, mixes static and mobile sensor nodes to accomplish particular functions. In a hybrid system, mobile nodes may serve as routers or controllers, communicating with static or standard sensors for data sensing. Scientific research institutions as well as public and private-sector industries, including maritime surveillance, have increased interest in UWSNs and robotics.

Communicating via wireless technology, sensor nodes placed across a wide region gather useful data from the sensor field in mobile underwater sensor networks. A sensor must be placed in a contextually suitable position before it can give useful data to the system, and many wireless sensor systems – particularly those used for remote monitoring and surveillance – can be placed in hazardous areas, most often without human intervention. Node deployment is regarded a crucial process in underwater sensor networks. It supports many important functions, such as routing protocol, localisation and network architecture, and it has a major effect on network performance. Sensor nodes in mobile deployments may operate freely in any orientation. In UWSNs, the majority of mobile deployment algorithms have focused on decreasing energy usage while increasing coverage and connectivity. Based on the current knowledge about the target, an underwater mobile sensor may adjust the initial node location; such relocation seeks to achieve an expected end configuration (Vilela, Kashino, Ly, Nejat, & Benhabib, 2016). Reorganisation or redeployment is necessary after adjustments have been made to the networks, often because of sensor failure (e.g. malfunction, power reservation) or target detection.

The research scope of this study encompasses homogeneous mobile sensor nodes. All sensor nodes are expected to have the same sensing, communication, computing and mobility features. Each node's sensing and communication coverage is expected to be ideal, which means that both coverage areas have a circular shape with no irregularity. Error-free data transmission and node position determination is another assumption. It is further assumed that each node possesses local information from neighbouring nodes within the direct communication range. Mobile sensor nodes can adjust their locations, consequently changing network topology. The topological changes are generally because of deteriorated underwater environments and the occurrence of certain events. As a result, the deployment solutions for mobile sensor nodes in the area of interest must address a variety of difficulties and concerns. In this research, a suite of multi-objective evolutionary algorithms is developed for coverage and energy consumption problems. Performance is evaluated using coverage rate, node energy consumption, Pareto optimal metrics, and execution time.

1.2 Research Motivation

This section describes the motivation for mobile sensor node deployment concerning UWSN technologies.

Some regions are particularly vulnerable to border encroachments and traditional threats such as kidnappings, illegal immigration and smuggling. Furthermore, regional authorities face a number of challenges in monitoring such locations. In many cases, funding has been increased, but has not been used optimally, because not all available assets are suitable to detect intruders at the border. Therefore, this study proposes mobile sensors in UWSNs as an alternative solution to assist authorities in monitoring locations without requiring a physical human presence at all times.

The wireless sensor network (WSN) is one emerging technology that has received particular attention from researchers and professionals alike (Jiang, Xu, & Wu, 2016). Market research by the firm (MarketsandMarkets, 2019) highlights the importance of wireless sensor technology. The WSN industry is expected to reach a market value of

USD 93.86 billion by 2023, up 18.55% from USD 29.06 billion in 2016. Data released by Market Watch (MarketWatch, 2019) also supports the growth rates for the forecast period. Wireless technology adoption and cost reduction are the main factors boosting market value. According to the report, demand is growing every year due to the compatibility and wide application of WSN in various fields. The report also affirmed that UWSN contributes to market growth, especially in monitoring specific areas using sensors and vehicles for data collection.

Furthermore, scientific research institutions – including maritime surveillance and the public and private sectors – have shown increasing interest in UWSNs and the robotics market. UWSN technologies are deployed and operated deep underwater with sensors that communicate via acoustic signals. Overall, these reports indicate that awareness of UWSN technology has increased significantly, sparking interest among researchers and offering better business opportunities in the industrial world. Due to enormous technical developments in UWSN, sensors are smaller, smarter, more adaptable, more energy-efficient, have improved processing power, and can operate underwater in different circumstances. UWSN technology is also connected with the Internet of Things (IoT) , internet protocol (IP)-based systems , and machine-to-machine (M2M) real-time monitoring frameworks (Lazaropoulos, 2016).

Consequently, according to (Ullah, Liu, Su, & Kim, 2019), many undiscovered resources – especially seas, sensors and sensor networks – warrant continuous study and investigation. In particular, UWSN theories and applications require further study and evaluation, given that UWSNs have shown substantial market growth worldwide. However, a wide variety of application needs have imposed critical limitations on network job performance and sensor node capabilities in the monitored area; research works have paid minimal attention to deploying multiple mobile sensor nodes in UWSNs with multiple-objective solutions.

Typically, the nodes in a practical surveillance network are sparsely distributed and use mobility as well as high-level coordination to provide a lower surveillance cost per unit area. For such a network to be feasible, a dynamic control system is needed to coordinate the autonomous vehicles using all available target information. The control and deployment system must balance multiple objectives while maximising network coverage, energy consumption, and hold time when performing this allocation. The techniques proposed in this research provide steps for AI-driven emerging technologies concerning autonomous vehicles for ocean applications. According to (Gartner, 2018), the first trend in strategic technology is that of autonomous objects using artificial intelligence (AI) to drive new hardware and software systems on land, in the air, or underwater.

1.3 Statement of the Problem

The mobile underwater sensor deployment problem comprised several factors. Causes trigger events; some causes may trigger the same event. Considering numerous causes and generalising the causes that trigger such events is essential to provide an accurate solution.

Underwater mobile sensor node deployment involves several challenges. Notably, existing mobile underwater sensor node solutions have enormous power needs to maximise network coverage; arbitrary movement of underwater mobile nodes or vehicles may cause swift and unpredictable changes to network topology. Thus, every time a node moves out of place, the network topology must be re-calculated and re-distributed. The network may also experience redundancy or coverage gaps due to the likelihood of sensor failure, destruction, or unusual events. Therefore, nodes require immense energy to communicate with the deployed sensors. In a centralised approach, every sensor node must send its position information to a control centre. The control centre then performs computations based on global information about the network topology before instructing the nodes about subsequent directions and movement. An increase in the number of mobile sensor nodes may overload the control centre, thus affecting processing time. In addition, UWSNs generally have a long propagation latency and a high bit error rate, which may cause the central controller to make poor node position movement choices in extremely dynamic situations.

Thus, it is crucial to design a decentralised system that makes movement decisions using local information. Also, the limited communication bandwidth of mobile nodes inhibits network performance in the underwater environment, creating a potential bottleneck for network expansion. Mobile node allocation to an underwater wireless sensor network environment always faces multiple conflicting objectives. In this research, the sensor node deployment tasks must satisfy maximum coverage in the deployment region while minimising energy consumption. Taking all objectives into consideration in a single problem poses a significant challenge that requires effective methods.

All of the aforementioned events contribute significantly to the node sensor's energy degradation. In addition, the node sensor's coverage strength will be impacted, hindering its ability to interact with other nodes in the network. These two factors, however, are both intertwined with, and in contradiction to, one another. Greater network coverage leads to increased energy usage and vice versa. The dynamic is reinforced by the continuously changing underwater environment, which makes it difficult to establish the position of the node sensors. Underwater mobile sensor node deployment is a stochastic system because the internal environment (i.e. arbitrary node movement, sensor failure and limited communication bandwidth) is affected by random events in the external environment (i.e. unpredictable underwater characteristics such as high acoustic propagation delay, water pressure, unpredictable underwater activities, and varying depth). The aforementioned problems give rise to the research statement for this thesis, which follows below:

The deployment problem concerning mobile sensor nodes in underwater wireless *sensor networks comprises several random events; hence, uncertain, and unreliable events contribute to a stochastic aspect. Existing deployment solutions lack in addressing the deployment of mobile underwater sensor nodes as a stochastic system, which faces internal and external environment problems that must be addressed for maximum coverage in the deployment region while minimising energy consumption.*

In existing systems, internal factors (mobile node features, i.e. arbitrary movements, unpredictable change to network topology, sensor failure) and external factors (underwater environment, i.e. water pressure, acoustic propagation delay) as a stochastic system are not considered/highlighted to maximize coverage while minimizing energy consumption.

1.4 Statement of Objectives

This research aims to enhance the mobile underwater sensor node deployment model by presenting a hybrid multi-objective formulation. The following set of objectives must be achieved to fulfil the research aim:

1. To analyse several mobile underwater node deployment algorithms by evaluating coverage rate, node energy consumption, and execution time.
2. To develop a hybrid fuzzy dominance-based decomposition technique and adaptive multi-parent crossover Genetic Algorithm for mobile UWSNs to optimise conflicting deployment objectives.
3. To evaluate the proposed solution and to validate and compare its performance with that of other existing techniques.

1.5 Research Questions

This study investigated three research questions:

1. What are the node deployment features to enhance the performance of mobile underwater node deployment?

2. How does the a hybrid fuzzy dominance-based decomposition technique and adaptive multi-parent crossover Genetic Algorithm perform for mobile UWSNs?
3. How can the hybrid multi-objective optimisation algorithm improve the existing solution in an underwater wireless sensor network (UWSN) environment?

1.6 Proposed Methodology

This research study is divided into four main phases (depicted in Figure 1.1) to accomplish the research questions, and objectives listed in Section 1.4. These steps are outlined as follows.

In the first stage, analysis for current mobile underwater sensor node approaches were conducted to understand sensor node deployment problems by evaluating coverage rate, node energy consumption, and execution time. The investigation reveals that uniform mobile node deployment is required for a cooperative approach. Furthermore, unpredictable underwater environments and central controller utilisation significantly affect deployment tasks for maximising coverage and minimising energy consumption for the sensor nodes to function efficiently. Considering these challenges, the research focuses on using a decentralised system that mitigates all the aforementioned shortcomings to independently create an intelligent movement system using local information for neighbouring nodes.

The second and the third phase focus on designing and implementing a hybrid multi-objective algorithm to balance the conflicts concerning deployment objectives. The fourth phase emphasises the evaluation of the proposed solution using various statistical analyses to establish the outcomes. This phase also tests the proposed algorithm's effectiveness by contrasting it with state-of-the-art solutions.

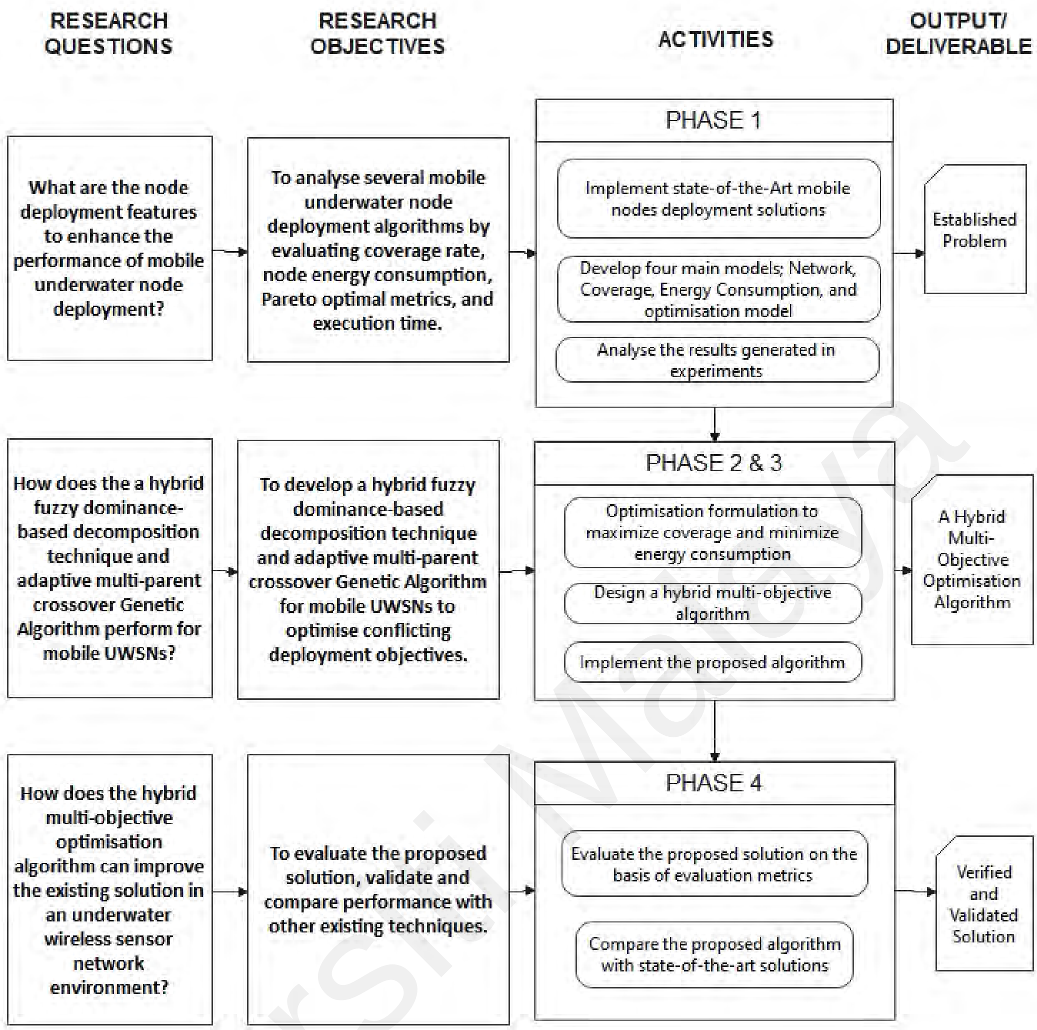

Figure 1.1: Research Questions, Research Objectives, Activities and Output

1.7 Significance of the Study

This research conducted a detailed analysis to improve the original non-dominated sorting genetic algorithm II (NSGA-II) into the proposed solution. The study evaluated the proposed solution with four different parameters (coverage rates, energy consumption, Pareto Optimal value and execution time) to prove its effectiveness in addressing the problems identified early in this research. The results and comparative analysis between the proposed and other existing multi-objective algorithms prove that the MOGA-AMPazy works on the multi-objective sensor nodes deployment problem better than NSGAII, SPEA2, and MOEA/D algorithms. The simulated grid improved in complexity as a further two conflicting objectives were specified and presented uncertainty in the form of a stochastic underwater environment.

A MOGA-AMPazy solution can provide mission planners with non-dominated solutions for mobile deployment of underwater sensor networks for effectively monitoring the region of interest. This study proposes mobile sensors in UWSNs as an alternative solution to assist authorities in monitoring locations without requiring a physical human presence at all times.

1.8 The Layout of the Thesis

The thesis comprises seven chapters; each contains a part of the research conducted to address the research problem. The thesis is presented and organised as described in Figure 1.2.

Chapter 2: Literature Review

Chapter 2 reviews state-of-the-art mobile sensor node deployments in UWSNs, offering a thematic taxonomy to categorise parameters for existing mobile nodes. This chapter also includes an in-depth analysis of current solutions, based on the thematic taxonomy provided, to emphasise similarities and contrasts among existing solutions. Finally, the research identifies potential domain challenges that need urgent consideration for optimal mobile sensor node deployment solutions in UWSNs.

Chapter 3: Problem Analysis

Chapter 3 analyses the problems and shortcomings in mobile node deployment solutions. It also presents an empirical analysis by measuring the performance and capability of existing mobile sensor deployment solutions. Coverage rate, execution time and energy consumption are the performance evaluation metrics used to contrast the proposed solution with existing approaches. The analysis shows that present state-of-the-art solutions fail to provide mobile node deployment because of two conflicting objectives: increasing coverage and reducing node energy consumption.

Chapter 4: A MOGA-AMPazy Method

Chapter 4 presents a hybrid multi-objective optimisation method, MOGA-AMPazy, to solve the underwater mobile sensor node deployment problem. It explains the phases of the proposed solution (MOGA-AMPazy) as well as the algorithms presented in each step. Furthermore, this chapter highlights the unique features of the proposed method that help accomplish performance goals and enhance mobile sensor node deployment in UWSNs.

Chapter 5: Experimental Data and Evaluation of MOGA-AMPazy Method

Chapter 5 explains the experimental setup and data collection approach chosen to evaluate MOGA-AMPazy. It also describes the tools used to evaluate the proposed method, followed by a breakdown of the performance parameters, experimental setup and statistical approaches used to provide value ranges ascertained using sample statistics.

Chapter 6: Results and Discussion on the Performance of MOGA-AMPazy Method

Chapter 6 presents results that validate the performance of MOGA-AMPazy. It also contains a comparative analysis of MOGA-AMPazy against the present state-of-the-art mobile node solutions in UWSNs.

Chapter 7: Conclusion

Chapter 7 concludes the thesis by explaining how it has accomplished its research objectives. It summarises the research contributions and highlights the significance as well as the limitations of the proposed solution. Finally, recommendations are made concerning potential avenues for future research in this domain.

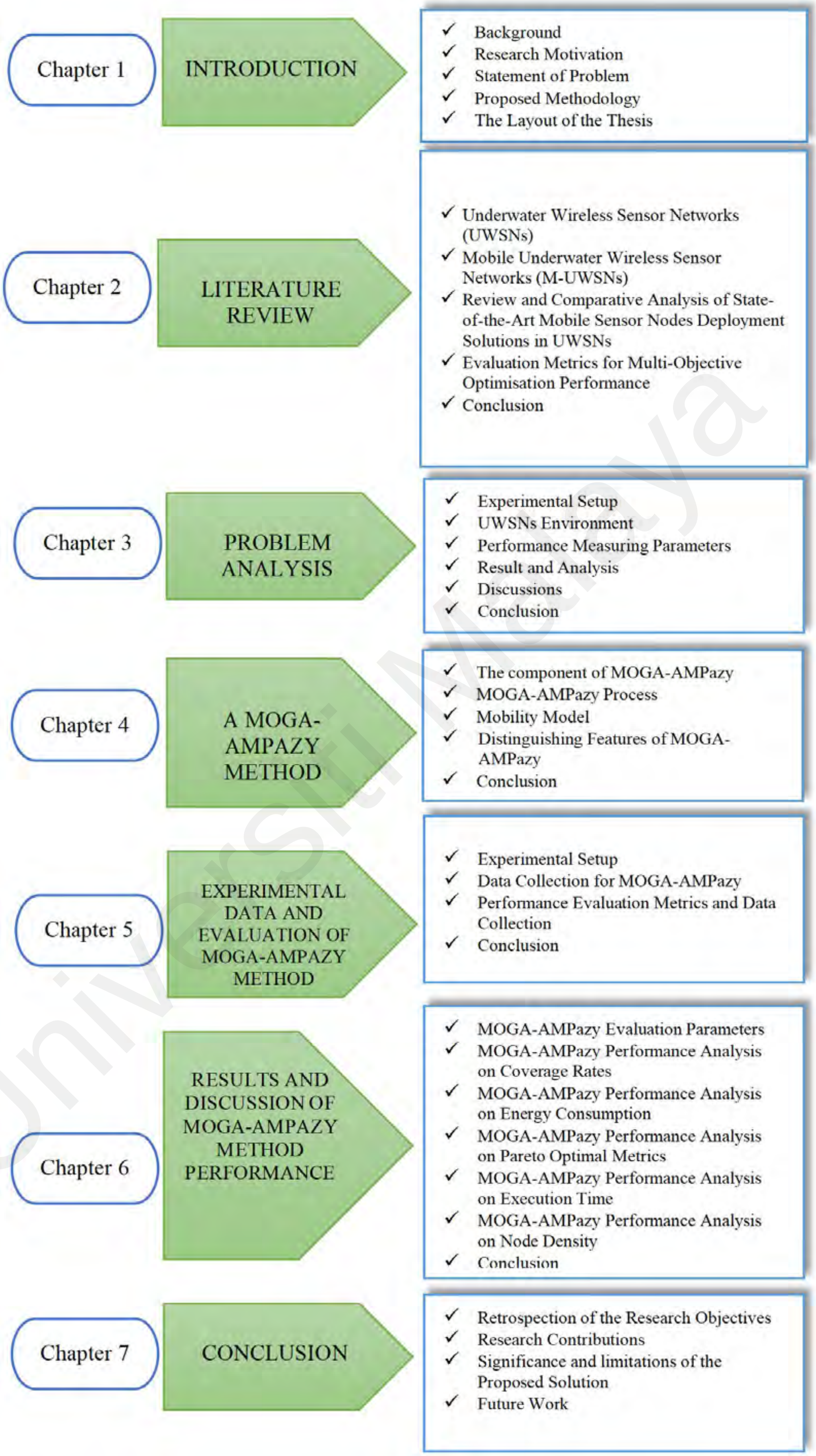

Figure 1.2: Outline of the Thesis

CHAPTER 2: LITERATURE REVIEW

This chapter aims to identify the critical shortcomings concerning state-of-the-art mobile sensor node deployment methods in UWSNs. It also examines the research problems stated in Chapter 1. A thematic taxonomy is proposed to classify the existing mobile underwater sensor node solutions based on parameters; this taxonomy is used to provide an in-depth study of the similarities and contrasts among existing solutions. This study also compares the various objectives emphasised in the current literature. Finally, the chapter covers a number of open-domain research questions addressed in this study. The chapter comprises the following sections: Section 2.1 introduces the fundamental concepts and classification of UWSNs. Section 2.2 explains mobile sensor characteristics in UWSNs, deployment mechanisms of mobile underwater sensor nodes, and the distributed models deployed for mobile UWSNs. Section 2.3 presents a study and comparison of the most recent mobile underwater sensor node deployment methods. Section 2.4 highlights future challenges and directions. Finally, Section 2.5 concludes by summarising the chapter.

2.1 Underwater Wireless Sensor Networks (UWSNs)

A UWSN is an area-specific monitoring network equipped with smart sensors and vehicles designed for cooperative communication via wireless connections (Akyildiz et al., 2005) A surface sink collects data from the sensor nodes; the sink node is equipped with a transceiver that regulates the acoustic signals sent by successive nodes. The transceiver may also transmit and receive long-range onshore radio frequency signalling. The collected data is used locally or in conjunction with another network for a specific purpose.

UWSNs can improve ocean monitoring and prediction activities; this capacity can be greatly improved by using the right technology and hardware to achieve the stated goal. Previous studies have shown that a combination of AUV and underwater sensors facilitates observation or monitoring applications at different depths (Akyildiz et al., 2005). (Xiao, 2010) stated that integrating the sensors and AUV requires a network coordination algorithm to achieve adaptive sampling and self-configuration. Adaptive sampling is a strategy for controlling mobile vehicles and moving them around the covered areas to collect data. Self-configuration is an AUV intervention procedure to detect gaps in the network caused by sensor node failure or channel destruction.

UWSN technology can replace traditional approaches by offering real-time monitoring, an onshore system to control underwater appliances remotely, and advanced devices for data recording. Typically, UWSN applications comprise three categories: scientific, industrial, and military and security. Military applications deploy sensor nodes to detect the movement and location of the enemy. Nodes can be deployed to monitor ports and harbours, conduct border surveillance, identify underwater mines, and detect enemy submarines. Sensor nodes are equipped with seismic activity monitoring capabilities that allow early warning and monitoring during natural disasters. A wide range of applications requires rapid developments in standards and technologies for supporting and enhancing new use cases.

2.1.1 Thematic taxonomy of UWSNs

The authors built the classification based on surveys and trend analysis of credible articles in the last five years. The most frequent topics discussed in the literature review are also considered before devising the thematic taxonomy. Figure 2.1 depicts a thematic taxonomy of UWSNs to better understand their characteristics. Vital attributes are categorised based on Architectural Elements, Communication, Routing Protocol and Standards, Security, and Applications. These attributes are discussed in the following sections:

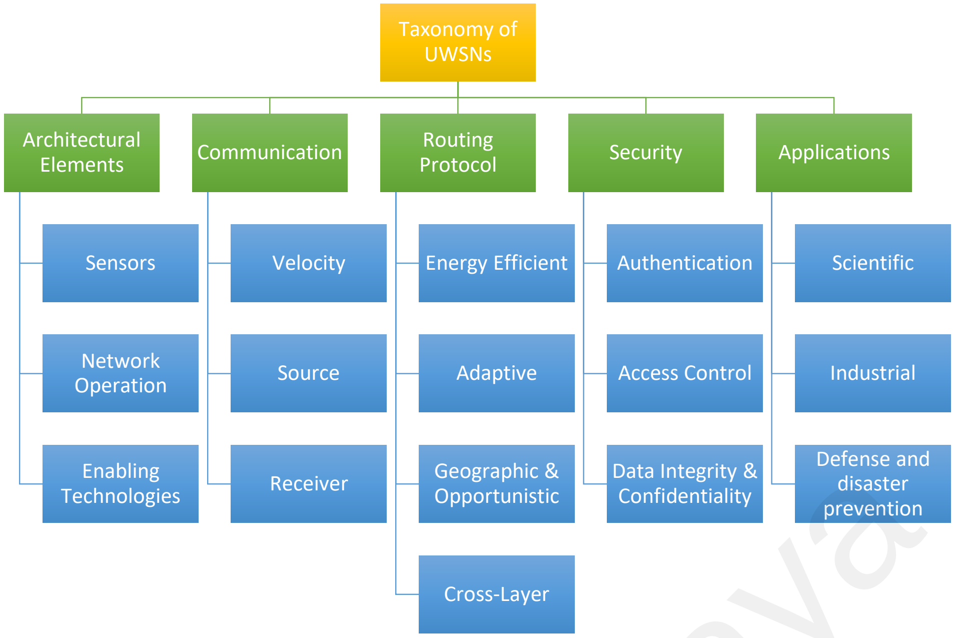

Figure 2.1: A Thematic Taxonomy of UWSN

2.1.1.1 Architectural elements

UWSN architecture is classified based on type, sensor node capability, and the spatial coverage of its applications.

2.1.1.2 Sensors

A hybrid architecture consists of both static and mobile sensors. In a 2D space, the static sensor node is generally mounted on the seabed. It communicates and transmits data to a sink node using multiple hops among several clusters (Felemban et al., 2015). However, a static 3D architecture has a slightly different setup, wherein sensors are deployed based on varying depths using floating buoys to adjust the wire length connected to the anchor on the seabed. In the mobile architecture, the sensor nodes can move freely, enabling dynamic changes to network topology. The mobile node requires two transceivers to maximise network data collection capabilities. A node might use ROVs, AUVs or sea gliders. The third type is a hybrid architecture that combines static and mobile sensor nodes to perform specific tasks (W. Lin, Li, Tan, Chen, & Sun, 2008). Mobile nodes can act as routers or controllers to communicate with static or standard sensors for data sensing in a hybrid system.

2.1.1.3 Underwater Acoustic Communications Characteristics

This section discusses the main principles of underwater acoustic communication. Additionally, all factors that affect the speed of sound and its influence on system performance or network device operation are also addressed.

Sound Velocity: Acoustic waveforms in water are dependent on the sound velocity and the environment. Empirical experimentation indicates that the key factors affecting sound velocity in water are temperature, salinity, and hydrostatic pressure. Figure 2.2 shows how these factors influence sound velocity in the ocean; critical points concerning these factors are discussed below.

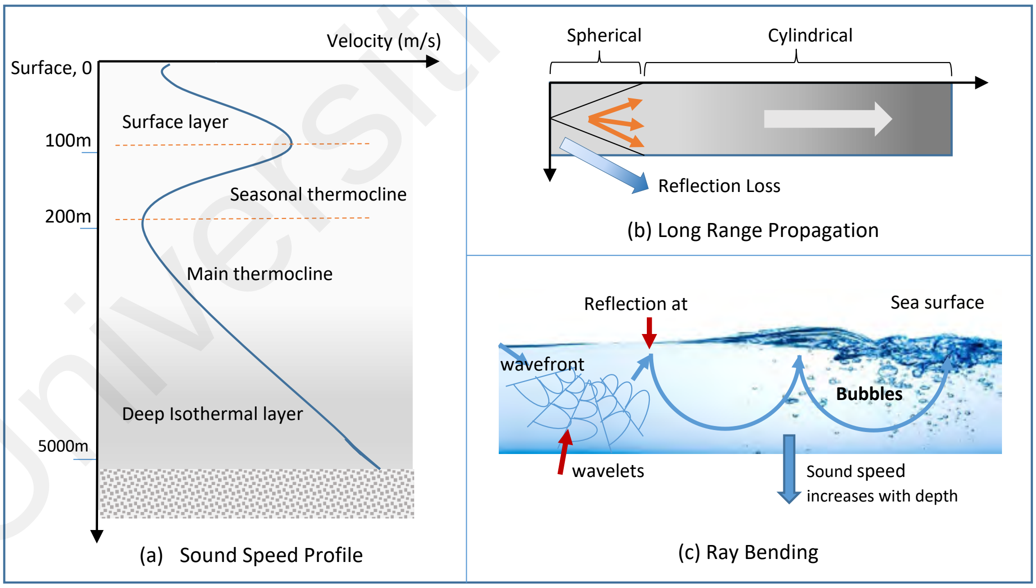

Figure 2.2: Sound Velocity Factors

1. Temperature. Sound velocity and water temperature are closely related: sound velocity rises as water temperature increases, and both increase as one moves closer

to the water surface.

2. Salinity. The second factor affecting sound velocity in water is the salinity ratio. However, salinity has a lesser effect on sound velocity compared to that of temperature. Different concentrations of dissolved salts in pure water have different effects on sound velocity. Ocean salinity is typically around 35 p.s.u; however, this value varies depending on water characteristics and the effects of rock, soil, and atmosphere. Water depth is another factor that affects salinity levels.

3. Hydrostatic Pressure. Hydrostatic pressure also affects sound velocity in water. Hydrostatic pressure increases with depth, increasing sound velocity (Lurton, 2002). The increase in hydrostatic pressure is directly proportional to the increase in depth.

Sound Velocity Profile: The ocean comprises two principal regions based on ocean depth. Each region produces different sound velocity variations known as 'sound velocity profiles'.

1. The 'surface layer' (0–100 m) is subject to environment, wind, and temperature change. Wind circulation can mix up this layer and convert wind power to isothermal (mixed layer). Sound velocity is reduced dramatically if the wind speed is higher than 7 m/s due to the dominance of bubbles found 10 m below the water surface. The temperature changes seasonally in the thermocline region (100–200 m). Temperature decreases as water depth increases. Consequently, in the winter season, the thermocline is weak because the water surface is consistently cool.

2. The 'main thermocline', which exists at depths of 100–200 m, is a region with minimal sound speed. At these depths, water temperature begins to increase. In the deepest zone, known as the 'deep isothermal layer', temperature characteristics depend on water density and salinity. Nevertheless, the impact of hydrostatic pressure

on sound velocity is significantly higher in the deep isothermal layer compared to temperature and salinity.

***Ray Bending*:** Sound waves interchange through the medium at a fixed rate, even with variable speed of sound. Ray bending is categorised as either qualitative or quantitative. In qualitative ray bending, sound speed will increase according to the depth and increasing bubble population; bubble population, in turn, decreases with increasing paths at the sea surface. Reflections occur when acoustic energy concentrates within a layer at the sea surface. It does not spread in all directions because sound speed is minimal where the course of the wave fronts propels towards water depth. This velocity profile is known as the Sound Fixing and Ranging (SOFAR) channel. Conversely, in quantitative ray bending, a sound ray moves horizontally through the points. Sound velocity increases linearly and parallelly with depth.

***Long Range Propagation*:** The reduction of sound in signal-to-noise amplitude due to long-range propagation, especially in the SOFAR channel, is influenced by geometry, attenuation and thermometry. According to the inverse-square law, acoustic wave strength decreases when a wave with spherical symmetry geometrically deviates from the primary source point. In the SOFAR channel, however, the action is different; the rays do not bend spherically but rather spread from a line source having cylindrical symmetry. The inverse-square law concludes that geometric distribution can reduce the strength of acoustic waves with decreasing distance.

There is a significant difference in the acoustic absorption rate between seawater and freshwater within the 5–50 kHz frequency range. The frequency-based difference occurs due to a mechanism associated with viscosity under 100 kHz, where magnesium sulphate is loose. As highlighted previously, water temperature is one of the main factors affecting sound velocity. At 1 km ocean depth, temperature and sound speed increase when sound spreads over distances at 4.6 m/s per degree centigrade.

***Sea Surface*:** The parameters affecting sound velocity in different areas, such as at the boundary, the bottom, and the ocean interface, have various proportions. The composition and density of hard rocks and sediments at the sea bottom are factors that affect the change in sound velocity. Furthermore, the bubble population on the sea surface is another factor that influences the velocity of sound. The presence of bubbles causes a rise in average water density. The velocity of sound reduces the occurrence of bubbles, as evidenced by formulations and experiments.

2.2 Mobile Underwater Wireless Sensor Networks (M-UWSNs)

Each sensor in a UWSN can monitor and detect events occurring in a 3D underwater environment. In most cases, fixed or static sensors are emplaced. However, the natural behaviour of water and coverage area of interest can change dynamically. Furthermore, the application of stationary sensors can make some targets challenging to detect, especially if the target is moving or located in zone not covered by the sensor network. In traditional underwater monitoring applications, sensors are usually mounted on the seafloor or are attached to a pillar or float on the water surface. With the deployment of mobile sensors, the probability of target detection is higher, and the monitoring process is more efficient. Moreover, M-UWs can readjust their positions in response to rapid changes in the network topology. After initial deployment, the mobile nodes can self-deploy in the region of interest. In this study, multiple nodes or multi-robots are used to prevent intruders from entering protected areas using cooperative monitoring and approaching near the preclusion point.

M-UWs, which are distinct from conventional wireless sensor networks, include the following features:

- **Complex application environment**

  Underwater environments are complex and hostile, disintegrating and degrading sensor nodes' responsivity, making routine operations more difficult to maintain. Therefore, in order to adapt to this environment, the nodes must be optimised.

- **Limited sensor energy**

  Because sensor nodes are submerged, it is not easy to supply power because they cannot move. As a result, one of the most critical issues to consider when designing sensors is energy usage. Furthermore, node energy consumption should be regarded as a crucial assessment indication when developing an algorithm.

- **Inefficient communication method**

  The underwater sensor network primarily employs underwater acoustic communication. However, several shortcomings – such as the restricted bandwidth of the underwater acoustic channel, high signal absorption, extensive and dynamic time delay fluctuations, strong multipath consequence and high failure rate – necessitate the construction of a much more complicated communication protocol. The communication efficiency of underwater networks is lower than that of terrestrial wireless sensor networks.

- **Changing network topology**

  Changes in network topology, often caused by water movement and aquatic wildlife, may cause acoustic communication connections to become intermittent and deactivate local topology. According to (Ojha, Misra, & Obaidat, 2020) the effects of moving nodes, waves, and the ocean environment cause the topology to be dynamic or irregular. As a result, MUWSNs should be capable of restructuring.

2.2.1 Deployment of Mobile Underwater Wireless Sensor Networks

Changes in network topology are often caused by water movement and underwater creatures. It may cause acoustic communication connections to become intermittent and deactivate local topology. As a result, MUWSNs should be able to restructure.

After the initial deployment, sensors are fixed to surface buoys or anchored to the ocean floor, and their locations are assumed to be permanent. Static deployment is subdivided into random and regular deployments. The next category is self-adjustment. After initial deployment, underwater sensor nodes may automatically change their depths by inflating floating buoys or by being pushed to specific desired locations by mobile sensor nodes. Self-adjustment deployment is further subdivided into uniform coverage and non-uniform coverage deployments. Underwater mobile sensor nodes patrol the controlled area to work with other sensors to carry out monitoring duties. Some algorithms and movement-based algorithms for self-adjustment incorporate the mobility of sensor nodes induced by water current or marine life. The last few years have seen significant growth in UWSN applications, especially in hostile environments where manual deployment is unlikely or hazardous and subject to time constraints, including battlefield observation, disaster-affected area monitoring and deep-sea observation. The distribution of sensors in a particular area would be implemented by randomly dropping sensors on the water surface. However, if the sensors are not properly distributed to the region of interest with depth adjustment, it will amount to a waste of resources (i.e. sensors) because the network will be incapable of accurate event detection. Thus, mobile sensor nodes – which can adjust their placement autonomously in the specific area to improve network coverage, connectivity, scalability, reliability, energy consumption and satisfy application requirements – constitute a significant advancement.

2.2.1.1 Deployment Objectives

When deploying sensor nodes, four objectives should be considered: network coverage, network connection, network lifespan, and energy efficiency.

***Network Coverage*:** Coverage of an area is described as each place within the sensing range of at least one active sensor node. Three distinct kinds of network coverage issues exist: area, point and barrier coverage. The aim of area coverage is to completely cover the area; however, if there are insufficient sensors, complete coverage cannot be obtained, and the objective is then to maximise coverage. In certain instances, complete coverage of a particular region is not necessary. As a result, partial coverage that ensures a certain level of coverage is both adequate and acceptable. In many applications, monitoring the whole region is superfluous and it is sufficient to monitor particular locations. At least one sensor node should be present at each location. In such scenarios, deployment costs are reduced because fewer sensors are needed to cover the whole region.

Point coverage (i.e. monitoring points of interest, or PoIs) involves surveillance of enemy soldiers and bases as well as the capture of possible mobile targets using real-time video footage. Flying mobile sensors may be used to monitor a PoI in such applications. PoIs may be mobile or stationary. A PoI is fixed if the position is always the same. Covering a fixed PoI with previous knowledge of its location is easier than following a movable PoI. If the location of PoI varies, it is said to be mobile. If mobile sensors are employed, they should be placed to cover and monitor this movable PoI as it travels. If static sensors are used, they should be positioned in such a way that at least one sensor node may cover each new location of the PoI. Sensors are used in various critical applications to monitor events occurring inside the region rather than to identify intruders attempting to penetrate the area.

Barrier coverage, which ensures the observation of every movement passing through a barrier of sensors, is a well-known coverage model for such applications. There are two kinds of barrier coverage: complete and partial. A barrier is considered completely covered if at least one sensor node is present at each point along its length. Partial coverage occurs when the number of sensors is inadequate to cover the barrier completely. The deployment method should guarantee that by repositioning the sensor nodes along the barrier, they identify an intruder attempting to penetrate it with a probability that exceeds a specified threshold.

***Network Connectivity*:** Two sensor nodes are deemed linked if they can communicate directly (single-hop connectivity) or indirectly (multi-hop connectivity). It is insufficient to guarantee coverage without taking connection into account while monitoring a particular region. Detectable events must be reported to a sink, which requires a connection between the sensor nodes and the sink to ensure data transmission. Data transmission, therefore, should be given the same priority as coverage. Thus, in order to effectively monitor a particular region, many applications need complete coverage and connection in order to gather and report data. It is not always essential to provide complete connection across the region under consideration. It is, however, necessary to provide intermittent connection via the use of a mobile sink that travels between unconnected nodes and gathers data.

***Network Lifetime*:** Sensor arrangement is a pressing issue for some essential objectives in UWSNs, including network lifetime. Network lifespan refers to the time between network deployment and the point at which the network is deemed inoperable. Balancing energy during initial placement is essential for increasing the lifetime of UWSNs. When a UWSN has been in operation for a long time, the batteries of some sensor nodes (such as those close to the sink) become exhausted, resulting in a broken network connection. Thus, by implementing mobile nodes, lifetime is increased while k-coverage is preserved.

***Energy Efficiency*:**

Wireless sensors are battery-powered, which means they have a finite amount of energy because it is not possible to replace or recharge their batteries. Typically, active sensors detect their surroundings, send sensory data to their next hop neighbours, receive sensory data from their neighbours, and relay that data to the sink. As a result, the active sensor's energy rapidly depletes as monitoring continues, resulting in coverage gaps. The UWSN's lifespan begins when it meets the first coverage gap. Due to the ad hoc deployment of mobile sensors during initialisation, it is paramount that sensors communicate with one another in an energy-efficient manner.

The selection of deployment objective as a parameter is usually chosen as the optimisation objective. However, several parameters need to be optimised simultaneously. Therefore, multi-objective optimisation offers the best solution (Tam, Hung, Binh, et al., 2021).

2.2.1.2 Control Strategy

One of the challenges of multi-mobile-sensor or multi-robot networks is how the sensors/robots communicate to share information. There are two main approaches to controlling a robot team: centralised and decentralised/distributed. In a centralised approach, each sensor node must send its position information to a control centre. Then, the control centre performs computations based on the global information about network topology before directing the nodes toward the specific location. However, the centralised approach has several limitations, which are addressed by the decentralised approach. An increase in mobile sensor nodes can overload the control centre, increasing processing time and rendering the centre incapable of operating over large spaces, which leads to poor decision-making in highly dynamic settings. These constraints may preclude the use of a centralised approach in real-world settings.

However, in the decentralised approach, the control process is distributed between the sensor nodes and is more applicable in non-uniform settings compared to centralised control. Each node has independent local information from its neighbours, which it uses to autonomously generate an intelligent decision about the next movement. Each sensor is expected to make a decision based on its limited communication and sensing capabilities, as well as the limited information gleaned from other sensors. This study aims to increase network coverage and minimise energy consumption while providing for a decentralised deployment of nodes in a dynamic ocean environment for mobile underwater wireless networks.

2.2.1.3 Topology

Unique underwater circumstances present difficulties regarding sensor nodes and topology creation, enabling MUWSNs to change dynamically, affecting network stability. Because complex underwater circumstances have a considerable impact on M-UWS, the architecture of the network varies, as well as communication delays and energy consumption. Generation of topology is the process of placing sensors in the water for creating effective overlay networks. During this process, the initial energy of each sensor node, the communication link and the coverage of entire networks are fixed, which impacts network coverage efficiency and life expectancy. Continuously changing the node depth reduces overlap between neighbouring nodes and improves the monitored area's coverage; nevertheless, these two techniques require all nodes' information. The first important issue is scientifically assigning nodes, creating 3D architecture, and using the fewest undersea sensor nodes possible to cover the region without aperture monitors. A useful and highly efficient method for topology creation should have the following characteristics:

Reducing energy consumption of nodes. Underwater sensor nodes have limited energy to refill. Thus, it is essential to decrease power consumption and extend the coverage of sensor nodes during the generation of topology.

Connection communication delay: Limiting variables, such as the bandwidth of the subsequent acoustic communication, lead to short and lengthy delays in communication. This should minimise the distance or the path among the sensor nodes in order to reduce communication delays by using the network topology produced by the 3D topology method.

Overlay network efficiency: Due to the limited monitored overlay region, the method for generating the 3D topology should utilise the fewest sensor nodes possible.

2.2.2 Design Factors

Several variables influence deployment design and affect how well the application works. They concern:

1. **Sensing model.** The sensing model estimates the sensor's probability of detecting a target or an event. Provided the target or event occurs at position p_j inside the RoI, P_{ij} represents the likelihood of a sensor detecting the target or event. Binary and probabilistic sensing models are two types of sensing models. The model of binary sensing is referred to as the Disk Model. Binary and probabilistic sensing models are two types of sensing models. Binary sensing (also referred to as the 'Disk Model') merely presumes that a sensor's detecting range is fixed at r_s. When an event occurs at a location p_j that is equal to or less than r_s from the position of the sensor, the event is sequentially recognised by s_i and vice versa. Nevertheless, probabilistic sensing models are designed to detect different variables influencing sensor readability. These variables include – in addition to the nature of the detected physical event and poor sensor detectability – ambient circumstances such as noise and obstacles.
2. **Sensing and communication range.** The sensing and communication ranges define the smallest number of sensors required to completely cover the monitored region.
3. **Number of sensor nodes for deployment.** The monitored region decides whether

the number is enough to cover the RoI completely. This RoI is generally considered to have a regular form (e.g. rectangle, disk). In reality, however, complex forms with irregular boundaries are quite common.

4. **Initial topology.** If a centralised deployment method is employed, a mobile robot should gather the starting locations of the nodes that the centralised deployment algorithm would use to calculate the final node positions, and this information should be made widely available. If a distributed deployment method is selected, it should include a neighbourhood discovery phase as well as a distributing phase to enable sensor nodes to rapidly find other attached components.
5. **The existence of obstacles.** No sensor node should be positioned in such a way that it cannot be found. Thus, obstacles must be identified and the deployment algorithm must employ a method to circumvent them. Additionally, if the monitored entity's form is complex with irregular boundaries, modifications to the deployment procedure will be required.

2.2.2.1 Mobility Model

'Node position' refers to a node's specific location. In mobility models, node velocity and direction are the mobility characteristics that capture node movement. Some models also take into account node acceleration. 'Inter-contact time' is defined as the time gap between two successive interactions between identical nodes. The term 'contact duration' refers to the time period during which two nodes are in radio contact. The mobility model is critical in determining the execution quality of the node deployment. After initial deployment, 'mobility' refers to the nodes' capacity to alter their own positions. Inadvertent movement may result in coverage gaps. However, deliberate movement (active mobility) may be used to improve network coverage, connectivity and lifetime.

The mobility model is designed to describe the movements of mobile users and the time difference between positions, speed and acceleration. Mobility models are constructed by specifying several parameters relating to node movement. Fundamental characteristics of mobility models include the beginning position, direction of travel, velocity range, and speed variations over time of mobile nodes. Mobility models are classified into four groups based on their unique mobility characteristics:

Random: Random models are composed of nodes that move randomly; they may be further categorised according to the statistical characteristics of randomness; examples the random waypoint, random direction, and random walk mobility models. Random mobility models allow mobile nodes to wander independently and randomly. More precisely, the destination, speed and direction, regardless of the location of other nodes, are all selected at random.

Temporal dependency: It quantifies the relationship between the present velocity (magnitude and direction) and the prior velocity. Nodes with identical velocities exhibit a high degree of temporal dependence. Mobility models with temporal dependence are likely to have their occurrence patterns affected by their movement histories. The category covers the random mobility Gauss-Markov and smooth models.

Spatial dependency: In certain situations of mobility, the mobile nodes move in a correlated fashion. It is a measure of the degree to which two nodes are reliant on one another in terms of their mobility. Two nodes travelling in the same direction have a high degree of spatial dependence. These mobility models are referred to as spatially dependent mobility models. This category contains models of mobility such as the mobility model for the group of points of reference and other spatially associated models.

Geographical Restrictions: The movement of mobile nodes in this model is limited by streets, highways and obstacles. Two examples of this mobility model are the pathway and obstacle mobility model. In most real-world applications, a node's mobility is affected by its surroundings.

2.2.2.2 Applications

UWSN technology can replace traditional approaches by offering real-time monitoring, an onshore system to remotely control underwater appliances, and advanced data recording devices. Commonly, UWSN applications comprise three categories: scientific, industrial, and military and security.

Scientific. Environmental monitoring and ocean sampling encompass a wide range of applications for UWSNs in the scientific field. The environmental monitoring application monitors chemical and biological pollution deposited on the seabed (Stojanovic, 2003; Akyildiz et al., 2005). Scientific applications of UWSNs also include water-quality assessments that involve human participation using real-time notifications for affected individuals (Merico, 2010). In (Lazaropoulos, 2016), robotic fish were employed to measure oxygen levels in the water and pressure and temperature monitoring (Majid et al., 2016). An ocean sampling application described in (Luo et al., 2017) focused on monitoring a wide coastal region at several sites utilising underwater sensor vehicle technology to investigate ocean phenomena. The collected data is automatically sent to the shore for further analysis. (W. Lin et al., 2008) present a coral reef application that combines sensor network technology, big data, and the Internet of Things (IoT) to study the effects of ocean salinity, temperature, humidity, and pressure on coral bleaching and marine ecosystems. Long-term marine environmental monitoring can also be implemented using a combination of various agents and communication methods. (Lončar et al., 2019) conducted experiments at Biograd Na Moru, Croatia, by combining Autonomous Surface Vehicles (ASVs), highly mobile artificial fish, and artificial mussels to collect data.

Industrial. Numerous industrial applications of UWSNs facilitate commercial activities. Notably, (Saeed, Ali, Rashid, Qaisar, & Felemban, 2014) have designed a prototype for monitoring oil and gas pipelines underwater. In (Abbas, Bakar, Arshad, Tayyab, & Mohamed, 2016; Jawhar, Mohamed, & Agrawal, 2011), the authors also designed an underwater oil and gas pipeline monitoring system that requires control of an actuation component.

Defense and Disaster Prevention Application. Military and defence applications use a combination of underwater sensors to pre-emptively identify potential enemies by monitoring ports and harbours (Antonelli et al., 2016), detect sea mines (Kumar et al., 2004), conducting border protection against incursions by hostile warships or submarines, and performing distributed tactical surveillance (Kemna, Hamilton, Hughes, & LePage, 2011). Furthermore, advanced UWSN technologies, such as the mobile underwater sensor network, are also utilised to provide early warnings about natural catastrophes such as seismic activity on the sea floor (Antonelli et al., 2016). (Jain & Virmani, 2017) designed a model for real-time tsunami prediction and used data about the 2004 Indian Ocean tsunami for evaluation.

2.2.3 Self-organisation Theory

This section introduces the self-organising principle and its properties.

2.2.3.1 Principles of Self-organization

Self-organisation occurs when an organisation has internal organisational instructions/-forces. In other words, a system is said to be self-organising if it evolves without external interference (Von Foerster, 1960). The greater a system's capacity for self-organisation, the greater its capacity for generating and maintaining new functions. Self-organising systems are capable of evolving and improving their organisational behaviour or structure on their own. In essence, a self-organising system is one that lacks a centralised and global control unit that regulates the system's behaviour. On the contrary, each unit in the system population operates independently and is unaware of the system's goals. It operates according to its own set of regulations. The global organisation is created when all peers' contributions are added together, which indicates that each peer may decide its own behaviour without depending on a central server's selection.

2.2.3.2 Properties of Self-Organisation

A self-organising system is one that interacts with its surroundings to generate novel structures and functions. Self-organising systems are distinguished from conventional mechanical systems by several unique characteristics. For instance, instead of centralised control, there is constant adaptability to changing environmental conditions. Thus, local interactions result in global organisation; this is the most conspicuous characteristic of a self-organising system. Local interactions occur concurrently with fundamental physical processes, and any effect from one area to another must pass through all intermediary regions. Throughout the process of self-organisation, all of the system's components are inextricably connected. It is necessary to understand the structure of a regional component in order to comprehend the structure of the components that compose its adjacent regions. Organisational control is spread across a self-organising system such that every component contributes to the ultimate state of the system. Although centralised systems may have certain advantages over distributed controls, even centralised systems must include some aspect of distributed control.

Complex self-organising systems often include many chains of positive and negative feedbacks, allowing changes to be amplified in one direction while being repressed in others. This results in unpredictable behaviours. There are many autonomous and organisation-closed subsystems in a self-organising system. These subsystems communicate more subtly. Additionally, they adapt to the terminal structure and control subsystems at a higher level. As components, the newly created subsystems include the old subsystems. Each self-organising system comprises a tiered architecture of subsystems at various levels. Systems interact with counterparts at the same level. Thus, hierarchy is a key characteristic of self-organising systems.

2.2.3.3 Self-Organisation Algorithms

Intelligent computing is a highly effective method for resolving increasingly difficult issues. It is capable of automatically adjusting settings during calculation in order to achieve optimal results. Evolutionary computing – which includes, for example, genetic algorithms – seeks optimal solutions by mimicking the dynamics of biological evolution found in nature. Swarm intelligence algorithms are novel evolutionary algorithms that have significant similarities to evolutionary methods and genetic algorithms. (Hackwood & Beni, 1992) originally suggested swarm intelligence in the context of cellular automata. Swarm intelligence refers to a collection of non-intelligent creatures that collaborate to solve issues via dispersed methods. Entities communicate through changes in the surrounding environment, either directly or indirectly. Through collaboration, these non-intelligent creatures demonstrate intelligence. A distinguishing characteristic of swarm intelligence is that, despite their basic individual behaviours, when all individuals cooperate together, they form a system capable of displaying very sophisticated behaviours. Swarm intelligence solves dispersed issues without relying on centralised control or global models.

Genetic algorithms are variations of stochastic search algorithms that mimic organism development (e.g. survival of the fittest, genetic mutations). (Holland, 1975) pioneered the use of genetic algorithms to explain adaptation processes in both natural and artificial systems. Genetic algorithms are mainly characterised by (1) directly operating structural objects, (2) the absence of assumptions regarding derivatives and functional continuity, (3) implied parallelism and better search performance via global optimisation, and (4) using the probabilistic search-optimisation process to determine and guide an optimised search area and adaptive adaptability automatically. These properties of genetic algorithms have encouraged their extensive use in areas such as hybrid optimisation, machine learning, intelligent systems, signal processing, and adaptive control. The primal technologies for the future of computer technology include genetic algorithms, adaptive systems, cellular automatic systems, chaos theory and artificial intelligence.

In addition to the well-known algorithms, many novel algorithms have been suggested for studying self-organisation. These newcomers include the fish colony algorithm (Li & Qian, 2003); (Grosenick, Clement, & Fernald, 2007); (Chen, Shatara, & Tan, 2009), the bee colony algorithm, the co-evolutionary algorithm, the memetic algorithm, the hybrid optimisation algorithm, the bio-inspired algorithm, evolutionary programming, evolutionary strategy, and parallel algorithm are all well-known algorithms. To summarise, the self-organising mobile sensor network enhances fault tolerance, sensing, scheduling, monitoring, surveillance, and control of underwater activities.

2.2.4 Prospect Theory

Mathematician Daniel Bernoulli pioneered risk-averse decision-making by incorporating risk aversion theory, utility, the St. Petersburg paradox, and risk premiums. Bernoulli also proposed the utility function as a way to describe how individuals make decisions. Bernoulli believed that individuals prioritise utility above anticipated value and are generally risk-averse. This paradigm served as the foundation for utility theory, which incorporates both descriptive and normative components. In 1944, John von Neumann and Oskar Morgenstern presented expected utility theory (EUT), a revised conventional model based on Bernoulli's principle, which has since been utilised as a risk-averse rational choice model on the premise that individuals act rationally when making risky choices (Andriotti, 2009). However, EUT is unable to represent a decision of human attitudes toward risk (i.e. risk seeking and risk aversion) because it does not account for human intuition or the ideas and preferences that pass through the human mind without much contemplation (Coleman, Pellon, & Zhang, 2007).

(Tversky & Kahneman, 1992) addressed this constraint by developing a choice theory based on the prospect theory (PT) that correctly explains how individuals make decisions. They asserted that PT covers typical human risk-taking attitudes that cannot be represented by EUT, including risk aversion and risk seeking. Risk aversion is frequently assumed while conducting economic research under uncertain circumstances, but not risk request. PT explains why individuals often make illogical choices. Daniel Kahneman and Amos Tversky, two behavioural psychologists, created prospect theory in 1979 to explain human decision-making in risk situations. This model belongs to the category of subjective probability and aids in decision-making based on context and loss aversion in the present condition. Prospect theory is a descriptive method for modelling how humans make risky choices. Prospect theory suggests that decision-making under uncertainty may be seen as a choice between many gambles or possibilities.

The purpose of using prospect theory in this study is to ensure convergence via risk assessment. Success probability and risk evaluation are used to make the choice. Prospect theory is a subfield of subjective probability and assists in making choices based on context and balancing loss aversion based on the present condition. The system uses probability weighting to solve multi-objective problems in a computationally efficient manner and adds dynamism to the proposed method.

2.3 Review and Comparative Analysis of State-of-the-Art Mobile Sensor Nodes Deployment Solutions in UWSNs

This section consists of two major subsections; the first presents a comprehensive review of current state-of-the-art mobile sensor node deployment solutions selected from the literature as well as an independent assessment of selected works. The second subsection comprises a comparative analysis of state-of-the-art mobile sensor node deployment solutions.

2.3.1 State-of-the-Art Mobile Sensor Nodes Deployment Solutions in UWSNs

This sub-section presents a comprehensive review of state-of-the-art mobile sensor node deployment solutions. It critically analyses various techniques used to deploy sensor nodes. Selected works are thoroughly reviewed and reported individually.

UWSN deployment algorithms can be categorised into four types: random, planned, incremental, and movement-assisted. The first three are static deployment methods, whereas the fourth is a dynamic approach. Random deployment is used when the RoI is inaccessible or when no prior knowledge is available (e.g. disaster zones and military applications). It is also used during the first phase of movement-assisted deployment methods, in which sensor placements are modified depending on the result of random placement (G. Wang, Cao, & La Porta, 2006). When the RoI is accessible, planned deployment is used, wherein sensor locations are determined simultaneously. Incremental deployment strategies use a one-at-a-time centralised approach to place the sensors (Farmer & Poulos, 2015). Each node determines its location using the information provided by the previously deployed nodes. The advantage of the incremental deployment algorithm is that it can optimise node locations in each step; however, network initialisation time is lengthy because the sensors are deployed iteratively. Random and planned deployment methods suffer from inaccuracy due to limited control over actual sensor locations.

Furthermore, because incremental deployment methods are complex and time-consuming, this research proposes movement-assisted deployment algorithms. These algorithms place sensors by optimising one or more UWSN design objectives under specific application constraints. Typical objectives include maximising coverage, minimising power consumption, and facilitating reliable network connectivity. In movement-assisted deployment algorithms, sensors are first deployed randomly and then moved using knowledge of other node locations.

The main movement-assisted approaches are discussed below.

Computational Geometry (CG): CG is formally defined as ‘the systematic study of algorithms and data structures for geometric objects, with a focus on exact algorithms that are asymptotically fast’ (De Berg, Van Kreveld, Overmars, & Schwarzkopf, 1997). The purpose of this research was to develop efficient deployment methods for static and mobile UWSNs. The Voronoi Diagram (VD) is a flexible geometric structure found in physics, astronomy, robotics, and networking (Deif & Gadallah, 2013). It is closely related to the Delaunay Triangulation, which is used extensively in topographic maps to approximate the Earth’s topography using observed heights at a limited number of sample locations. The Delaunay Triangulation (DT) is a particular case of planar point set triangulation. The DT of a set P is always an angle-optimal triangulation of P, which indicates that it would provide a more accurate representation of a particular terrain compared to other potential triangulations. (X. Wang, Li, Liu, & Ding, 2017) developed the double-coverage method to solve early failures caused by high energy consumption in UWSN sensor node deployment. To begin, the coverage area of the node is split into a grid surrounding the mobile sensor network using AUV. The sensor node is then analysed in order to create the sensing and node mobility models. Additionally, a double-coverage method is suggested that is based on the deployment of mobile nodes in UWSNs. Four mobile sensor nodes placed on the rectangle’s four vertices are required to double its coverage. However, it is found that the model only covers 58% of the sensing region.

Furthermore, a fully distributed deployment algorithm (FDDA) can automatically control all distributed underwater mobile sensors, reposition them, and offer maximum-level underwater strong k-barrier coverage (Shen, Zhang, Zhang, & Shi, 2019). The FDDA method places a sensor in the centre of each regular hexagon with circumradius r, where r is the sensing radius of the sensor; the total sensing range of all sensors covers the rectangle.

Although the suggested design beats Hungarian and HungarianK in terms of building k-barrier coverage duration, the research does not account for energy usage when determining algorithm performance. Meanwhile, (Dang, Shao, & Hao, 2019) presented a target detection coverage method based on 3D-Voronoi partitioning to guarantee the network's reliability. The research expands Voronoi partitioning based on the 2D plane, enabling 3D-Voronoi partitioning of sensor nodes in 3D areas. The 3D space is partitioned into Voronoi polyhedron 'V-body' units, each of which is an irregular multifaceted closed convex body. The findings indicated that the 3D-VPCA algorithm was capable of substantially lowering network energy usage while increasing coverage ratio. Nonetheless, it is necessary to identify the mobility of sensor nodes as one of the performance variables affecting energy efficiency.

Artificial Potential Field (APF): In robotics, Artificial Potential Field (APF) methods were initially presented by (Khatib, 1986). The research proposed was a novel real-time obstacle avoidance strategy for mobile robots based on the idea of 'artificial potential'. A mobile robot is supposed to be travelling through an artificial field of forces (i.e. virtual forces). The desired location (i.e. the objective) may be represented by an attractive pole that exerts virtual attractive forces on the mobile robot. Obstacles are depicted as repellent surfaces that apply virtual repulsive pressures on the mobile robot. Due to its simplicity of operation and sole reliance on neighbour node information, the virtual forces-based node deployment method is more efficient for deploying nodes in unknown maritime space. The virtual force model is extensively used to enhance network coverage in 2D and 3D settings. The oceans comprise stratified and rotating complex fluid movements (Pedlosky, 2013), and due to the movement of nodes induced by ocean currents, network coverage cannot be assured. As a result, the node deployment approach for underwater wireless sensor networks must be redesigned rather than being directly applicable to terrestrial wireless sensor networks. (Mahboubi & Aghdam, 2016) and (Fang, Song, Wu, Sun, & Hu, 2018) addressed these issues by developing a node mobility strategy based on the virtual force model and Voronoi diagram in order to maximise sensor coverage in a given 2D sensing field.

Sensors are considered as virtual particles that are susceptible to repulsive or attractive virtual forces in virtual force-based deployment algorithms. Each sensor is acted upon by virtual forces emanating from the vertices and limits of its Voronoi cell. When the distance between sensors is smaller than a defined threshold, the sensors move in response to the resulting forces. Simulations show the suggested algorithms' effectiveness in improving sensor coverage. The network models, on the other hand, are simplistic and cannot be directly applied to maritime settings because they do not measure the effect of water currents on network coverage.

(C. Liu, Zhao, Qu, Qiu, & Sangaiah, 2019) extended the 2D virtual force model to 3D environments. They proposed the implementation of virtual forces in distributed node deployment algorithm (DABVF) by analysing the force of anchor nodes in the underwater environment. The result of the algorithm shows that the solution can increase the network coverage of an underwater wireless sensor network. It examines the effects of node mobility and analyses node locomotivity and location after offset in order to maximise the efficiency of node mobility during node deployment. The simulation results demonstrate that DABVF is capable of increasing network coverage, decreasing node energy consumption, balancing residual node energy, and optimising node distribution. Nevertheless, the proposed algorithm was not proposed for network topology repair strategies if sensor nodes fail. It also requires more complex coordination among sensors. (Jiang, Wang, & Liu, 2018) previously presented a virtual-force redeployment method based on energy consumption, which modifies node displacement by considering various energy consumption situations in different directions; their research introduces the concept of water flow force. The authors suggest a virtual forces redeployment algorithm based on energy consumption (VFRBEC), in which different energy consumption scenarios in various directions are utilised to alter node displacement. When the sensor node movement and water flow velocity rise, however, the algorithm works poorly and becomes very complicated.

Metaheuristic Algorithm: This thesis focuses on meta-heuristic algorithms as they can solve complex optimisation problems effectively and efficiently. A recent study by (Yakıcı & Karatas, 2021) concluded that the metaheuristic approach could create diverse and optimum solutions with less execution time. Optimisation methods can be categorised in several ways. One perspective suggests trajectory- or population-based techniques. Trajectory-based algorithms follow a single path. Hill-climbing and simulated annealing (SA) are examples of trajectory-based optimisation approaches.

In contrast, population-based algorithms, such as particle swarm optimisation (PSO), employ multiple elements (called 'particles' in PSO) to examine numerous paths simultaneously. Optimisation algorithms can be divided into deterministic or stochastic algorithms. Stochastic algorithms employ randomness whereas deterministic algorithms do not. Stochastic algorithms may provide different solutions every time they are executed, even with the same initial values. Genetic algorithms and PSO are examples of stochastic algorithms. The algorithm search capabilities can be used for classification into local and global search algorithms. Hybrid algorithms are mixed techniques that utilise a combination of these characteristics. In general, 'heuristic' indicates discovery using trial and error, whereas 'meta' means a higher level. A trade-off between diversification (global search) and intensification (local search) is utilised in metaheuristic algorithms.

These algorithms can provide solutions to complex optimisation problems within an acceptable amount of time (Desale, Rasool, Andhale, & Rane, 2015). However, there is no guarantee that the solutions are optimal; at the same time, most algorithms do not guarantee convergence. There are two significant components in any metaheuristic algorithm: exploration and exploitation.

Exploitation (or intensification) indicates a local focus on the search region of the current solutions. Exploration (also called diversification) generates diverse solutions by exploring the entire search space. The former ensures that the discovered solutions are locally optimal, whereas the latter enables the algorithm to escape local optima to increase solution diversity. A proper balance between the two components can significantly affect the performance of the algorithm (Arora & Singh, 2017). The first metaheuristic algorithm was developed based on Evolutionary Strategies (ES) during the 1960s (Box, 1957), (Friedman, 1959). As previously reported in (Harizan & Kuila, 2020), ES can produce better and more effective results than other algorithms, especially for solving coverage and connectivity problems. A more comprehensive description of nature-inspired ES can be found in (Singh, Sharma, & Singh, 2021). The discussion covers the implementation of optimisation algorithms and their performance in solving problems in sensor networks. During the 1980s, simulated annealing (SA) (Kirkpatrick, Gelatt, & Vecchi, 1983) (inspired by the metal annealing process) and Tabu search (TS) were developed. TS was the first metaheuristic algorithm to use memory (Lee & El-Sharkawi, 2008).

Ant colony optimisation (ACO) (Dorigo & Gambardella, 1997) was introduced in 1992. This algorithm was inspired by the social interaction of ants using pheromones. Genetic programming was developed in the same year, which provided the basis for machine learning. In 1995, PSO was developed, and differential evolution (DE) was introduced the following year. The latter is a vector-based evolutionary algorithm and has been shown to outperform GA in many applications. GAs are optimisation algorithms based on natural selection observed in biological evolution. The use of crossover, recombination, mutation, and selection was proposed to study artificial systems in the 1970s. Since then, GAs have been used to solve both discrete and continuous optimisation problems. These algorithms are very efficient at solving multi-objective optimisation problems for which deterministic optimisation methods are not practical. GAs have inspired many modern evolutionary algorithms. All variants of genetic algorithms have the following three essential components (Spector, 1994):

- Encoding, which is a genetic representation of the candidate solutions
- A fitness function (or cost function), which is a mathematical expression of the optimisation objective
- Stochastic genetic operators (crossover and mutation) to change offspring composition

Crossover exchanges a segment of two binary strings, whereas mutation arbitrarily modifies the entries of a binary string. Due to the dynamic nature of underwater circumstances (e.g. ocean currents at various depths), it may be impractical to rely on a central controller to direct AUVs to perform particular tasks. Each AUV must autonomously make intelligent decisions regarding its future moving location in order to safeguard vital assets in practical applications. AUVs effectively change their locations under the direction of intelligent topology control algorithms to create desirable UWSN topologies. (Zou, Gundry, Kusyk, Sahin, & Uyar, 2013) previously described a PSO-based topology control mechanism, dubbed 3D-PSO, for AUVs operating in unknown 3D underwater environments. Each AUV may change its speed and direction of travel in order to create a more equal dispersion by using the 3D-PSO algorithm. The quality of possible solutions is determined using the 3D-PSO fitness function. The force-based fitness function determines the fitness of each candidate solution for an AUV in terms of its distance from the area of protection and its proximity to other AUVs within the communication range. The simulation tests demonstrate that 3D-PSO is capable of rapidly and effectively dispersing AUVs and protecting naval assets during 3D underwater operations. However, the algorithm's performance in difficult and hostile settings has not been evaluated; to date, AUVs have been placed only near the sea surface and close to the coast.

The term 'event' (target) is usually clearly defined in non-uniform coverage sensor deployment; the goal of sensor deployment is to cover events and to align sensor distribution with event distribution. However, the majority of sensor deployment techniques are intended for use in static situations and thus cannot be utilised to change the locations of sensors in dynamic, unpredictable settings to provide the necessary monitoring quality. To resolve these problems, (Du, Xia, & Zheng, 2014) examined the challenges of non-uniform coverage by 3D underwater sensors. Their research was inspired by particle swarm systems and introduced PSSD (particle swarm-inspired underwater sensor deployment), a distributed deployable underwater sensor method. By modelling particle flight and including crowd management, PSSD can direct sensors to locations with a high event density while avoiding congestion. PSSD are designed and simulated using the Lagrange flow model, and their performance is assessed using a theoretical information measure. Nonetheless, the findings of the dynamic simulation experiment indicated that the sensors moved more in order to cover the events, resulting in increased energy usage.

In addition to the above efforts at resolving the dynamic issue, another study proposed determining a sufficient number of active nodes in UWSNs at various times to cover the detection targets (C.-C. Lin, Deng, & Wang, 2015). This proposal, termed multi-population harmony search algorithm (MPHSA), takes into account the scheduling issue for dynamic heterogeneous UWSNs in a 3D environment. Additionally, it offers a sleep scheduling system for sensor operations, in which sensors operate and sleep alternately to avoid excessive power consumption or failure. The suggested method may use the changed locations to create a new sleep schedule dynamically. The algorithm, on the other hand, used a centralised optimisation approach that required a greater level of computing complexity. The objects to be monitored are extremely dynamic and unpredictable in open and complex underwater environments. To optimise monitoring target coverage, it is critical to design a technique for altering sensor locations in response to changing surroundings and targets. In response, (H. Wang, Li, Chang, Chang, & Fan, 2018) developed a distributed hybrid fish swarm optimisation method (DHFSOA) that includes the effects of water flow and the functioning of an artificial fish swarm system. The goal was to increase the event set's coverage effectiveness and prevent blind movements by sensor nodes. By mimicking fish foraging behaviour and congestion management, the DHFSOA endows sensor nodes with an autonomous proclivity to cover events. Additionally, the idea of an 'information pool' is proposed to help nodes extend their vision range and avoid blind motions, thereby decreasing node energy consumption during deployment. By incorporating these behaviours, sensor nodes gain the capacity to independently cover occurrences within a monitoring region.

Additionally, (Zhang et al., 2017) offer a novel optimum deployment coverage technique for improving 3D underwater sensor networks based on an enhanced fruit fly optimisation algorithm (UFOA). This technique achieves optimum global coverage by mimicking the foraging behaviour of fruit flies. It has a faster convergence rate, fewer configuration settings, and a better global search capability. Its deployment strategy seeks to find the maximum value of the objective function that corresponds to the optimisation strategy with the fewest repetitions possible. Despite its ability to escape the local optimum, the algorithm may nevertheless fall into it. Finally, (H. Wang, Li, Zhang, Fan, & Li,

2019) demonstrated that node self-deployment with non-uniform coverage in underwater acoustic sensor networks is challenging due to the difficulties associated with assessing the 3D underwater environment. In the deployment of marine sensors nodes, nodes are placed in a 3D underwater environment for stereoscopic sensing of the monitored area. Sensor nodes mounted on autonomous underwater vehicles (AUVs) navigate to necessary monitoring locations before using their self-organising capability to perform network construction and real-time monitoring in order to collect monitoring data inside the network distribution area. This research uses cubic modelling for 3D underwater environments to address coverage issues for target events in 3D space. The position of the nodes is represented using a coordinate system for a three- dimensional grid in order to suggest an ant colony optimisation-based self-deployment method that maintains maximum coverage and connection. This method was used to find the optimum network configurations in order to maximise network coverage and connection.

Others:Currently, dynamic placement of an underwater objective is critical for optimising scheduling and disaster relief. One of the critical criteria for underwater mobile targets is their precise and rapid positioning. (Chengming, Xinnan, Gaifang, Hai, & Xuewu, 2017) suggested an approach using the probabilistic placement of mobile targets to balance underwater sensor network coverage and positioning performance. With the assistance of node coverage optimisation, a probabilistic location model for mobile targets is developed. Additionally, simulation tests were conducted to assess the suggested algorithm's effectiveness in probabilistic target placement.

In another study, (Lv, Li, & Li, 2017) suggested implementing an algorithm for an enhanced k-coverage sensor network based on probabilistic sensing; maximum weight matching was used to implement a centralised allocation method that may significantly decrease the energy consumed by node distribution. The anchor node, which was placed in predefined water locations, gathered and transmitted data to the sink node through the relay node. In order to achieve complete coverage of the water area, the network nodes were required to meet the sensing probability of the grid points in the water resource monitoring area to the greatest degree feasible throughout the sensor nodes' deployment and coverage procedures. Moreover, the scientists used an iterative greedy approach to guarantee that sensor nodes were located within the coverage areas of the water resource region. In a simulation, the coverage rates were found to be greater than the event probability and the energy-efficient coverage methods. However, when the algorithms were applied to a large-scale water region, the communication cost rose as network size increased. As a result, this algorithm's performance for underwater localisation is substantially reduced, falling short of the necessary technical criteria.

Additionally, new mobile underwater acoustic wireless sensor networks are being deployed to increase network cluster connection and coverage. (Raj Priyadarshini & Sivakumar, 2020) proposed random sensor cluster deployment; they also developed an energy prediction method based on the Markov chain Monte Carlo (MCMC) process, which maximises coverage and connection during data transmission by analysing the sample value of a given parameter related to the water surface. Nonetheless, the proposed system's throughput improves once five hundred data or more transfers from sensor nodes have occurred.

All of the above-discussed approaches and their limitations demand a reassessment of the deployment design to support mobile sensor node arrangement in UWSNs. Specifically, an advanced deployment solution for mobile nodes using metaheuristic optimisation will bring promising and improved performance in maximising coverage rates and minimising sensor node energy consumption. UWSN applications in industries may face more challenges from emerging technologies, Internet of Things, target detection, underwater vehicles, and harsh underwater environments. However, sophisticated advanced mobile sensor node deployment solutions using metaheuristic optimisation should be adequate to handle such challenges.

2.3.2 Comparative Analysis of State-of-the-Art Mobile Sensor Nodes Deployment Solutions in UWSNs

This sub-section critically analyses mobile sensor nodes deployment solutions based on the parameters extracted from literature as presented in Section 2.3.1. Table 2.1 presents an abstract view of a comparison of state-of-the-art mobile sensor nodes deployment solutions in UWSNs.

Table 2.1: Comparison of State-of-the-art Mobile Sensor Nodes Deployment Solutions in UWSNs

Paper Reference	Deployment Techniques				Deployment Objectives				Control Strategy		Design Factors					Mobility Model				Applications
	CG	APF	Metaheuristic	Other	Coverage	Connectivity	Energy	Lifetime	Centralized	Decentralized	Sensing Range	Communication Range	Number of Sensors	Topology	Obstacle	Random Model	Temporal Dependency	Spatial Dependency	Geographical Restrictions	
MCMC				√	√	√		√		√	√	√	√	√	√	√				Environmental monitoring
Double-Coverage	√				√		√	√		√	√	√	√			√				Target monitoring
FDDA	√				√	√				√	√		√							Intrusion detection
3D-VPCA	√				√	√	√	√		√	√	√	√	√	√					Target detection
3D-PSO			√		√	√	√			√		√		√						Maritime Surveillance
UFOA			√		√	√			√		√	√	√	√				√		Environmental monitoring
Probabilistic positioning				√	√		√		√		√	√								Target monitoring
Improved k-coverage				√	√	√	√	√	√		√		√	√				√		Water resource monitoring
DABVF		√					√		√		√	√	√						√	Environmental monitoring
PSSD			√		√	√				√	√	√								Target/event monitoring
MPHSA			√		√	√	√		√		√		√							Target monitoring
DHFSOA			√		√	√	√		√		√	√		√						Event monitoring

CG= Computational Geometry
APF= Artificial Potential Field

Developers of applications prefer sensor deployments that adhere to the overall design goals. As a result, most node deployment methods proposed in the literature have focused on increasing coverage area, minimising energy consumption, establishing a stable network connection, and prolonging network lifespan; all four of these deployment objectives are considered by state-of-the-art mobile sensor node deployment solutions, including those proposed by Dang et al. (2019) and Lv et al. (2017). Alternatively, Shen et al. (2019), Zhang et al. (2017) and Du et al. (2018) chose combinations of coverage and connectivity objectives concerning sensor deployment. The remaining state-of-the-art methods investigated combinations of three such deployment objectives for delivering successful solutions.

The control strategy, which defines how sensor nodes are deployed, is an essential element to control the movement of mobile sensor nodes. Proposed control strategy solutions comprise architectures that are either centralised or decentralised. Centralised monitoring uses a central controlling server to perform monitoring control tasks. Alternatively, in decentralised monitoring, this task is distributed among several or all mobile sensor nodes. Existing mobile sensor node deployment solutions, such as in (Zhang et al., 2017), (Chengming et al., 2017), (Lv et al., 2017), (C. Liu et al., 2019), (C.-C. Lin et al., 2015), and (H. Wang et al., 2018), were based on a centralised controller. However, (Raj Priyadarshini & Sivakumar, 2020), (X. Wang et al., 2017), (Shen et al., 2019), (Dang et al., 2019), (Zou et al., 2013) and (Du et al., 2014) selected decentralised architecture for their proposed deployment solutions. Centralised architectures are inherently prone to failure. Decentralised architecture-based solutions, on the other hand, are more fault-tolerant. Despite the corruption of an agent on a physical server, network monitoring continues in a decentralised architecture. Over time, extensive literature has been developed on dynamic control by eliminating the position estimation. For example, (W. Wang et al., 2022) utilised the reinforcement learning method to learn the position and provided faster and more accurate results.

Sensor deployment solution effectiveness is influenced by several factors including sensing range, communication range, number of sensors, network topology, and obstacles. State-of-the-art mobile sensor node deployment solutions, including those of (Raj Priyadarshini & Sivakumar, 2020) and (Dang et al., 2019), considered all the stated factors and obstacles. However, (X. Wang et al., 2017) and (C. Liu et al., 2019) selected only sensing range, communication range and the number of sensors to measure solution effectiveness. The rest of the state-of-the-art techniques considered groups of two or three design factors as elaborated above.

Furthermore, the efficacy of mobile sensor deployment solutions in UWSNs depends on the mobility model of the sensor nodes. There are four distinct types of mobility: random, temporal dependence, spatial dependency, and geographical constraint. However, none of the extant deployment solutions have applied the temporal dependency mobility model in their algorithms. Mobile sensor node deployment solutions such as those by (Raj Priyadarshini & Sivakumar, 2020) and (X. Wang et al., 2017), provided random model solutions to control the movement of sensor nodes. In contrast, (Zhang et al., 2017) and (Lv et al., 2017) implemented the spatial dependency mobility model. The rest of the node deployment solutions considered geographical restrictions.

The growth of UWSNs contributes to several applications, such as environmental monitoring, target or event monitoring, intrusion detection, marine surveillance, and water resource monitoring. However, most current deployment techniques have been used for target or event monitoring applications in the underwater environment. Many current studies concentrate on energy effectiveness and coverage in the network of underwater wireless sensors and have suggested various systems, algorithms, techniques and designs. Despite these efforts, a comprehensive and universally applicable solution has not yet been devised. In response, this study establishes a state-of-the-art categorisation of UWSNs on a variety of parameters, including sensor type, deployment strategy, sensing model, coverage, and energy efficiency. The standard ACO has evolutionary constraints, requiring many iterations to generate the optimal solution.

2.4 Evaluation Metrics for Multi-Objective Optimisation Performance

Currently, various multi-objective metaheuristics have been proposed. Most often, the performance is evaluated by conducting experiments and the results obtained are compared using test problems.

(Deb, 1999) introduced a test problem to compare several methods in the multi-objective optimization Evolutionary algorithm. Originally (Deb, 1999) had suggested a more systematic way of developing test problems for multi-objective optimization. After that, (Zitzler, Deb, & Thiele, 1999) improved the method by proposing six test problems. In 2002, (Deb, Pratap, Agarwal, & Meyarivan, 2002) selected five test problems named ZDT1, ZDT2, ZDT3, ZDT4 and ZDT6. All test problems are used for two-objective functions and are categorized in unconstrained test problems. According to (Deb, 1999), two objectives are sufficient to look at essential aspects in multi-objective optimization. (Deb et al., 2002) has also listed the characteristics of a two-objective optimisation problem: (1) multi-objective optimisation can obtain solutions for diverse Pareto-optimal fronts; (2) it is capable of converging to the Pareto-optimal front globally; and (3) it seeks to solve problems having convex, non-convex and Pareto-optimal fronts.

2.5 Conclusion

This chapter discussed essential features and techniques concerning mobile sensor node deployment in UWSNs. It followed the examination of recent works concerning the environment of UWSNs to identify potential problems. This chapter also discussed the inadequacies and challenges of existing mobile sensor node deployment techniques in the specified domain. All observations from this chapter have been summarised in Figure 2.3.

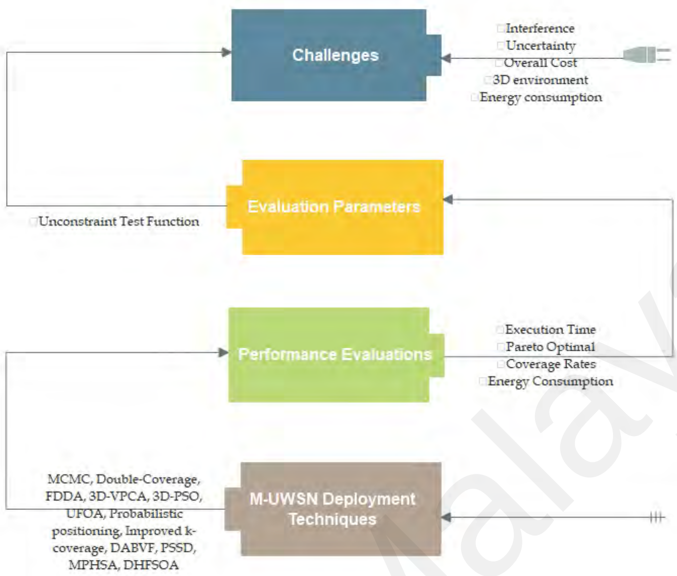

Figure 2.3: Finding from the Literature Review

In Section 2.2.3, the self-organising theory and mechanism have been elaborated. This research focuses on applying the self-organising theory to structure the functionality of underwater mobile nodes without centralised control. In a self-organising mechanism, all sensor nodes interact and contribute to the final arrangement in the deployment process. The self-organising theory adaptation in this research can overcome the shortcomings of the previous centralised control system, enabling sensor nodes to adapt to topological changes caused by the dynamic underwater environment.

In addition, this research also applies prospect theory, as explained in Section 2.2.4. The objective of using prospect theory is to guarantee convergence using risk evaluation. The decision is calculated using success probabilities and risk evaluation. This model fits within the subjective probability field and helps make decisions depending on the context and balance loss aversion based on the current state. Probability weighing allows solving multi-objective problems with reasonable computational requirements and adds a level of dynamism to the proposed technique.

Section 2.3 tabulates various recently used mobile sensor node deployments in UWSNs with different deployment approach categories based on the parameters extracted from literature (see Table 2.1). After review, four distinct metaheuristic algorithms were judged to be best suited for further analysis: particle swarm optimisation (PSO), fish swarm optimisation (FSO), ant colony optimisation (ACO), and fruit fly optimisation algorithm (UFOA). Further investigation of these algorithms has been presented in Chapter 3. Additional experiments have been conducted using the mobile underwater sensor node dataset to demonstrate shortcomings concerning convergence time, coverage rates, and energy consumption.

Furthermore, a taxonomy has also been developed to sort the state-of-the-art approaches that pose research challenges in deploying mobile sensor nodes. It is concluded that mobile sensor node deployment is applied in various UWSN applications, and close attention must be paid to the presented challenges, especially concerning coverage rates, the energy consumption of sensor nodes and convergence time to facilitate deployment tasks, which will be a core component of the future UWSN computing landscape.

Next, Chapter 3 describes the experimental setting for algorithms regarding mobile sensor node deployment, performance measures, and an evaluation of the results to illustrate the shortcomings of current methods. This study investigated four existing methods for deploying underwater mobile sensor nodes.

CHAPTER 3: PROBLEM ANALYSIS

This chapter aims to establish the problem highlighted in Chapter 1 and conduct a comprehensive investigation to show mobile deployment algorithms' impact on coverage rate, deployment time and energy consumption using the conventional method. This research applied different algorithms to select the best techniques for handling mobile sensor node deployment in UWSNs. The rest of the chapter is organised as follows: Section 3.1 discusses the applications and algorithms used to analyse the problem. Section 3.2 describes the performance measuring parameters. Section 3.3 discusses the experimental parameters used in the experiment. Section 3.4 presents results and discussions. Discussions based on empirical data analysis are summarised in Section 3.5. Finally, Section 3.6 iterates the findings of the analysis.

3.1 Experimental Setup

This section presents comprehensive information about the mobile sensor node deployment in UWSNs and includes the empirical study conducted to establish the problem. The discussion includes the experimental setup for the UWSNs environment and develops the compute-intensive tasks to perform the analysis. Unmanned vehicles employed in 3D spaces, including underwater and airborne theatres, are less expensive than comparable human-operated vehicles, making them ideal for a wide variety of applications. Unmanned vehicles operate in harsh environments, such as excessive depths for underwater investigation, harmful environments for maintenance missions, or radiation assessments in contaminated areas. Unmanned underwater vehicles (UUVs) are underwater vehicles that may navigate in any direction within a 3D underwater environment. They are usually remotely operated by humans. Unfortunately, it would be impossible to employ groups of UUVs in cooperative work.

Furthermore, the fast-changing and unpredictable underwater environment precludes the use of a central control device to manage the direction of movements and speeds of autonomous vehicles using dynamical algorithms. The term 'autonomous underwater vehicle' (AUVs) refers to a vehicle that can intelligently choose its next moving location without human assistance by using the local knowledge and onboard intelligence of its neighbours. A major feature of AUVs is their self-deployment capabilities. This means that AUVs may change their locations in response to changing environmental circumstances. As a result, this study focuses on moveable sensor nodes operated by AUVs. However, AUVs frequently operate with less accurate information compared to other vehicles and are equipped with limited-range communication devices. Excessive and inadequate information about the neighbourhood may hinder execution to consistent and stable space coverage.

In the experiment, the coverage rates, development time, and energy consumption of mobile sensor node deployments have been evaluated using four different types of algorithms, namely particle swarm optimisation (PSO), fish swarm optimisation (FSO), ant colony optimisation (ACO), and fruit fly optimisation algorithm (UFOA) on a UWSN platform. The following subsection provides details of the experimental setup.

3.2 UWSNs Environment

An underwater environment platform was used to carry out the experiments; UWSN platform specifications are provided in Table 3.1. The study investigates the effect of four different metaheuristic algorithms on UWSNs.

The experiment specification is set up to prepare the AUVs or mobile nodes operating in uncertain/harsh environments. It is in line with the research problem statement, which mentioned that existing deployment solutions fail to address the deployment of mobile underwater sensor networks as stochastic systems. The specification parameters include

Table 3.1: Specification of the UWSNs platform

Component/Parameter	Specifications
Deployment area	1000m x 1000m x 1000m
Environment map	300*500 pixels
Number of nodes	120
Sensor nodes model	AUV
Sensing range of node	20m
Communication range of node	100m
Nodes placement	Uniform
Maximum energy of node	2000 (mA-h)
Underwater acoustic model	Parabolic Equation
Water Density	1010 g/m^3
Water temperature	20 ˚C
Water salinity	30 PSU
Packet size	40 bytes
Data rate	128 bps

the underwater acoustic model, water salinity, temperature and density, which consider complex range-dependent environments (Y.-T. Lin, 2019). The simulation parameter are derived from (Y.-T. Lin, 2019) and (Priyadarshini & Sivakumar, 2021).

3.2.1 Algorithms

PSO, FSO, ACO and UFOA were the four techniques selected for experiments. These algorithms were selected based on literature review findings.

3.2.1.1 Particle Swarm Optimisation (PSO)

Each AUV operates in a topology control mechanism using 3D-PSO as its stand-alone software agent to modify AUV velocity and motion orientation. AUVs may operate freely in 3D Cartesian space in this context to defend a high-value military property (e.g. port entrance, ship's deck). Each AUV is equipped with a sensor capable of monitoring a small region around it. Once an underwater vehicle discovers an unfamiliar item (e.g. an underwater mine) in its detection zone, it transmits information to a data gathering station, such as a surface ship. AUVs' underwater surveillance must react to the continuously changing underwater environment and maintain connection to provide sufficient coverage and protection of the naval asset (or blocking the harbour access). Each particle in the swarm represents a possible solution for the speed and direction of the next movement. Once the maximum number of generations has been achieved, an AUV will then relocate (t) to an ideal position if its fitness function is better than the present location of the AUV. The 3D-PSO fitness function is used to assess the quality of possible solutions. The force-based fitness function assesses the fitness of each candidate solution for an AUV, taking into account the distance from the protective region and its neighbours within the communication range. Neighbours closest to a node will have stronger virtual forces than those farther away within the communication range.

3.2.1.2 Fish Swarm Optimisation (FSO)

The artificial fish swarm algorithm (AFSA) is an intelligent heuristic search algorithm for global optimisation. By simulating fishes' preying and survival activities, the AFSA can solve combinatorial optimisation problems, such as optimal ordering, grouping, or screening of discrete events with a faster execution speed than previous methods.

The fish swarm algorithm and the underwater mobile sensor networks are intrinsically related. The sensor node in the sensor network is equivalent to the artificial fish in the

AFSA. Events are analogous to food, and the method through which a node detects an event is analogous to an artificial fish looking for food. As a result, the AFSA is often utilised in underwater mobile sensor networks. By mimicking fishes' seeking and congestion management behaviour, the DHFSOA endows sensor nodes with an autonomous proclivity to cover events. Additionally, the idea of a 'information pool' is proposed to help nodes extend their coverage range and avoid making blind motions, thus decreasing node energy consumption during deployment.

3.2.1.3 Ant Colony Algorithm (ACO)

As a new swarm intelligence optimisation algorithm, ACO has the advantages of positive feedback, distributed and parallel processing, and self-organisation. It has fewer adjustable parameters, which is suitable for node deployment in large-scale complex UASN environments. ACO algorithms have demonstrated good performance when dealing with the travelling salesperson problem (TSP); the grid point of each deployed candidate sensor node was analogous to the city points in the TSP. All covered events were analogous to the task termination of post persons. The primary design consideration for ACO was to ensure that every sensor node in the deployment could be connected to the sink node to ensure network connectivity. The ants' movement radius was limited below the communication radius, and it was ensured that the destination grid point for every ant was directly connected to the sink node (or through other deployed sensor nodes). Thus, continuous ant movement caused the sensor nodes in the deployment to increase. In the end, the set of sensor nodes in the deployment could cover all monitoring points; the first iteration of ants was completed.

3.2.1.4 Fruit Fly Optimisation Algorithm (UFOA)

(Pan, 2011) developed a fruit fly optimisation method in Taiwan based on drosophila foraging behaviour. The fruit fly can smell a food supply from forty kilometres away; when it flies near the food source, it may utilise its acute eyesight to identify the location where its friends congregate and quickly go there. Each fruit fly in an iteration of a UFOA matches with a solution defined by the particular location of each sensor in the optimal solutions. The PSO method may be limited to a local optimal solution, thus reducing the accuracy of the global optimisation result. (Abidin, Arshad, & Ngah, 2011) reviewed the literature and discovered that the fruit fly algorithm similarly confronts the local optimum issue; nevertheless, they found that the fruit fly method is capable of escaping the local optimum.

3.3 Performance Measuring Parameters

Underwater mobile sensor node deployment performance was measured using coverage rates, execution time and energy consumption, which are discussed in greater detail below.

3.3.1 Coverage Rates

Coverage rate defines the amount of space the sensor nodes occupy, considering the total deployment space. Underwater sensor network coverage rate directly affects monitoring accuracy and comprehensiveness. The balance between these two factors is usually a trade-off. The spatial overlap rate needs to be turned down to improve the coverage rate; consequently, the accuracy will be lower. Thus, the balance between the two factors depends on the preferred deployment aspect. The event coverage scheme needs enhanced accuracy; concurrently, a full-coverage deployment scheme needs a high coverage rate.

3.3.2 Execution Time

Total deployment time is defined as the time elapsed from the first placement of AUVs inside an unknown space until an acceptable coverage rate is achieved. Total deployment time is an essential performance measure for assessing node self-positioning methods in time-critical circumstances in both civilian and military applications. A mobile sensor node or AUV needs some time at each step to run the algorithm and conduct the required calculations to identify its movement direction and speed. It then travels in many stages to the place specified by the suggested method.

3.3.3 Energy Consumption

Each sensor node's energy consumption is defined as the sum of sensor node maintenance energy used to keep the nodes active, sensor node transmitting energy, sensor node receiving energy, and the number of sensor nodes from which the sensor node receives data and transmits it to the sink node via multi-hop communication. Because movement is a power-intensive activity, decreasing travel distance for each AUV increases the UWSN's lifetime. After completing its task, the node calculates the distance of movement before updating the network's overall energy usage. Thus, the energy consumption concerning network formation can be calculated as node energy consumption for unit distance movement using sensor nodes' mobility.

3.4 Result and Analysis

This subsection illustrates the results acquired from coverage rates, execution time, and energy consumption using traditional mobile sensor node deployment in UWSNs.

3.4.1 Coverage Rates

The experiment was carried out in order to assess the coverage rates of current methods based on the Monte Carlo algorithm in the 3D node deployment space. Figure 3.1 shows a comparison of the coverage rates for each algorithm.

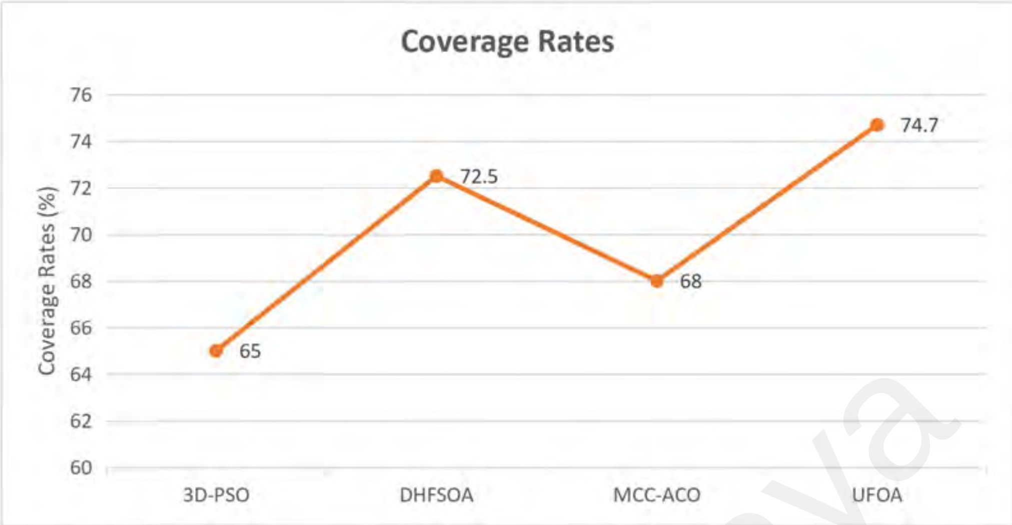

Figure 3.1: Coverage Rates of Existing Algorithms

From Figure 3.1, it is summarised that the UFOA outmatches all other algorithms in terms of coverage rates because it has a faster execution rate, fewer configuration settings and better global search capability (Zhang et al., 2017). However, MCC-ACO, DHFSOA, and 3D-PSO provide lesser coverage rates with smaller datasets. As illustrated above, DHFSOA and UFOA can enhance the underwater 3D network's effective coverage. MCC-ACO performs better than the 3D-PSO algorithm concerning effective coverage rate.

3.4.2 Execution Time

In time-critical circumstances in civilian and military applications, execution time is a key performance measure for assessing node self-organising methods. execution time is measured based on the period of time elapsed from the initial position of mobile sensor nodes or AUVs within the region of interest until the nodes' space is located. Figure 3.2 shows a comparison of the execution time for each algorithm.

Eventually, the sensor node locates an area near the optimum location for a precise search, enhancing the algorithm's local search performance and result accuracy. A high

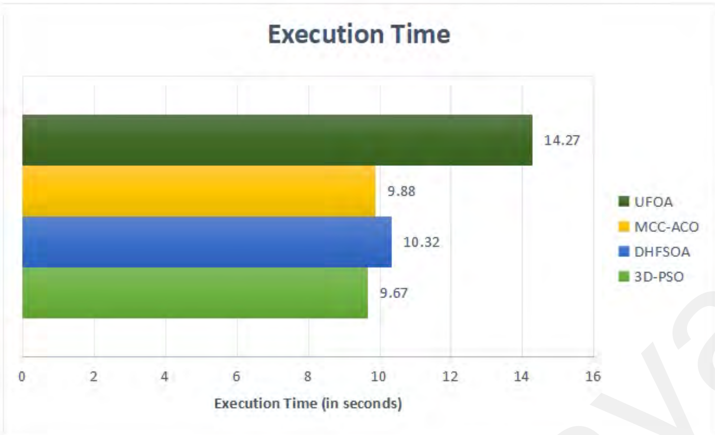

Figure 3.2: Execution Time of Existing Algorithms

execution time for mobile sensor nodes to optimally position themselves means travelling long distances to stabilise. According to Figure 3.2, it is evident that 3D-PSO takes the least time compared to other algorithms. It implemented a forced-based fitness function that determines the fitness of each candidate solution for the mobile sensor nodes in terms of its distance from the monitoring area and its proximity to other nodes within the communication range. Lower execution time values imply that sensor nodes or AUVs travelled a shorter distance to reach a particular topology.

3.4.3 Energy Consumption

Generally, UWSN sensor nodes are limited by energy stored on the onboard battery. Thus, reducing energy consumption during communication is critical for extending the network lifetime of UWSNs. Figure 3.3 shows a comparison of the energy consumption for each algorithm.

The amount of energy required to move a mobile sensor node a unit distance is frequently

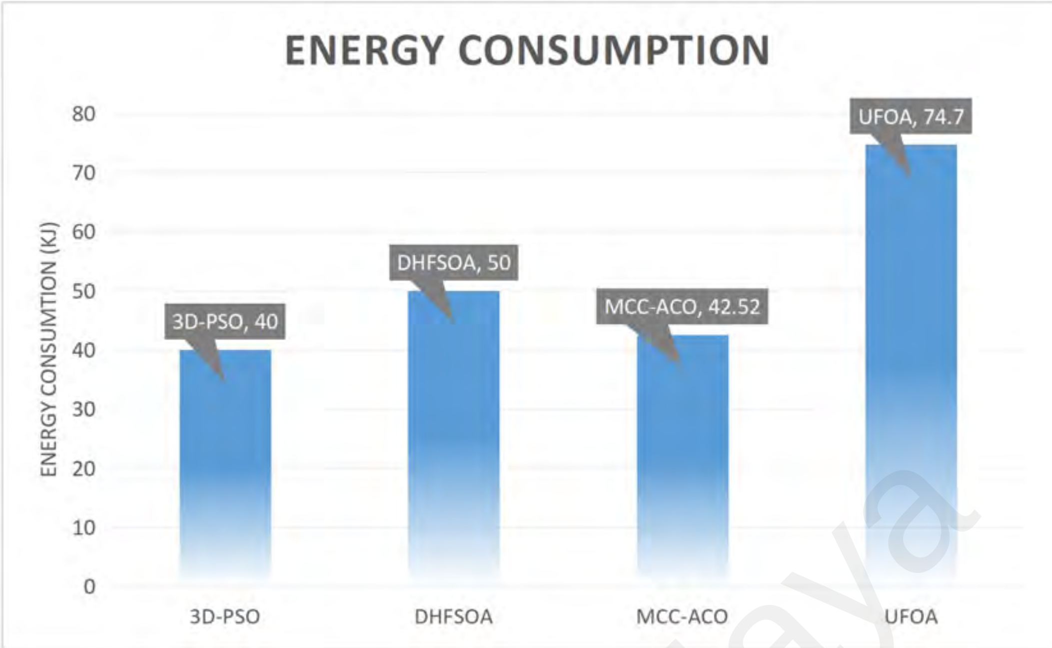

Figure 3.3: Energy Consumption (kJ) of Existing Algorithms

determined. The process of moving a mobile sensor node consumes a portion of its energy. Thus, distributing mobile sensor nodes must be done with as little energy as possible. UFOA has the highest energy consumption; the 3D-PSO technique consumes substantially less energy than the UFOA, DHFSOA, and MCC-ACO approaches. When the algorithm applied UFOA, it encountered challenges due to difficulties associated with 3D underwater environment. Despite its ability to escape from the local optimum, the algorithm may fall into it. When it is 'stuck' at the local optimum, the convergence rate will become slow and, at the same time, will result in higher energy consumption. UFOA also has complex computation for the optimisation process, contributing to high energy consumption, whereas 3D-PSO has less computational complexity in optimisation and thus consumes less energy.

The deployment performance metrics results for selected algorithms are illustrated in Figures 3.1, 3.2 and 3.3. Figure 3.1 indicates that the best coverage rates are obtained from UFOA. However, because UFOA consumes higher energy while moving from the initial position to the optimal location, it would not be an ideal fit for mobile sensor node performance (refer to Figure 3.4).

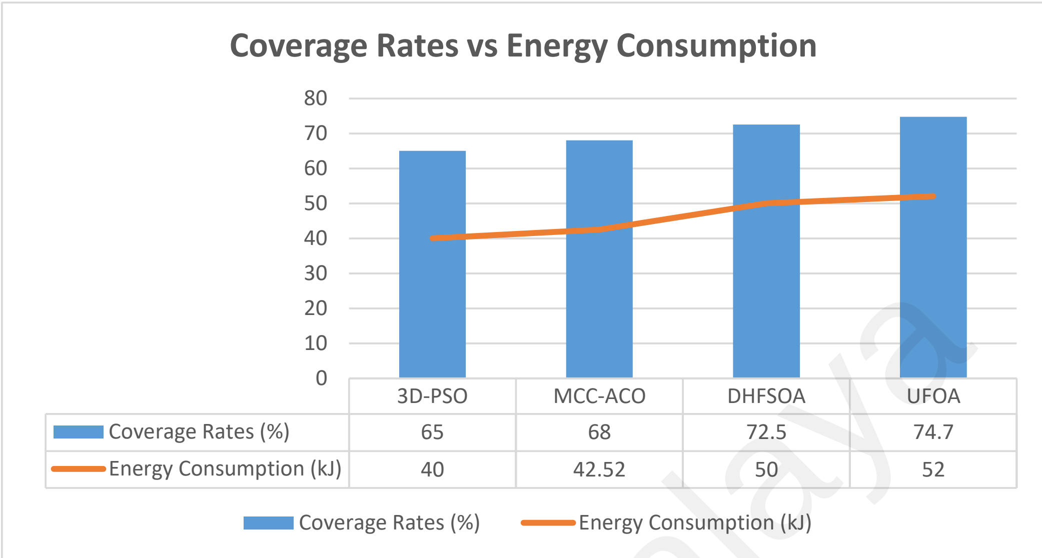

Figure 3.4: Relationship between Coverage rate and Sensor Node Energy Consumption

Figure 3.4 shows the different network coverage rates and node energy consumption. Maximising coverage entails optimising the space monitored by the mobile sensor nodes concerning the total deployment space. Minimising energy consumption or maximising network lifetime requires sensor nodes to be placed near the target. It is evident that the two objectives are conflicting. A higher number of network nodes are associated with a higher net energy consumption but lesser per-node consumption, thereby increasing the network lifetime.

3.5 Discussions

The empirical results highlight the following:

- Maximising coverage reduces the energy consumption of mobile sensor node deployment using existing algorithms.

- Performing mobile deployment without considering coverage rates and energy consumption of nodes may cause conflicting objectives.
- The inefficient movement and velocity of mobile sensor nodes require a longer execution time, significantly degrading existing system performance.
- The existing traditional approach has its performance metrics. However, the significantly higher execution time and energy consumption for deployment are challenges that need to be addressed.

This section clearly shows the severe drawbacks of ignoring the conflicting objectives of maximising coverage rates and energy consumption for deployment. The nodes affect the execution time of mobile sensor node deployment in UWSNs.

3.6 Conclusion

This chapter presents the experiments conducted as part of this study to analyse the performance of mobile sensor node deployment tasks. This study evaluated performance impact based on three parameters: coverage rates, execution time, and energy consumption.

The analysis results concluded that mobile sensor node deployment based on the existing traditional approach could significantly degrade deployment performance. In some cases, a longer execution time results in higher coverage rates and higher energy consumption. As a consequence, each solution is evaluated against a set of different (and often contradicting) objectives. The multi-objective method is, therefore, the best solution to address these issues.

The next chapter proposes mobile sensor node deployment using multi-objective optimisation because single optimal solutions are less reliable. Adopting the genetic algorithm based on adaptive multi-parent crossover and fuzzy dominance solutions offers a considerably less execution time, thus balancing the conflicting objectives between coverage rate and energy consumption.

CHAPTER 4: A MOGA-AMPAZY METHOD

This chapter proposes underwater mobile sensor node deployment using a multi-objective algorithm based on fuzzy dominance and a multi-parent genetic algorithm. The proposed solution is expected to improve the execution time of the deployment process, maximise coverage rate, and minimise energy consumption. This research has demonstrated several components of the proposed solution and its functional capabilities. The proposed solution is a multi-objective genetic algorithm based on adaptive multi-parent crossover and fuzzy dominance (MOGA-AMPazy), which deals with the problems highlighted in Chapters 1 and 3. The chapter is organised into five sections. Section 4.1 elaborates the significant components of the proposed solution for mobile underwater sensor node deployment comprising the network model, coverage model, energy consumption model, differential fuzzy dominance function, adaptive multi-parent crossover, prospect theory model, and multi-objective optimisation model. Section 4.2 illustrates the MOGA-AMPazy process as a decomposition structure and flow diagram. Section 4.3 describes the mobility model. Section 4.4 highlights the distinctive features of the proposed solution. Finally, Section 4.5 summarises the chapter.

4.1 The Component of MOGA-AMPazy

This study proposes a solution model that operates according to the workflow illustrated in Figure 4.1. It consists of four main parts: model development, objective formalisation, solution design, and evaluation.

The first part of Figure 4.1 concentrates on constructing four main models, namely net- work model, coverage model, energy consumption model, and multi-objective optimisation model. The network model focuses on the establishment of the environment for underwater mobile sensor nodes. Each node executes the algorithm without using

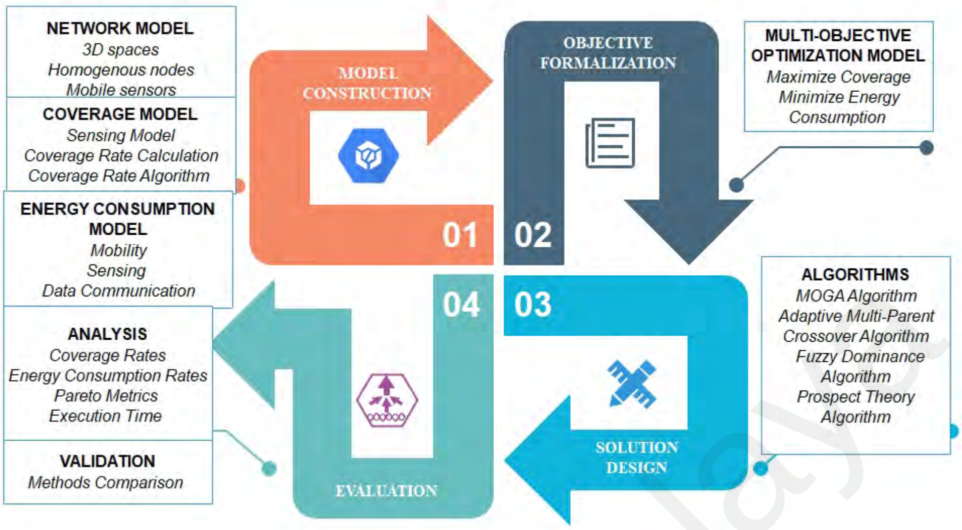

Figure 4.1: Workflow of the Study

the central node. This study assumes that all mobile sensor nodes are identical or homogeneous, especially concerning computational ability. Coverage is the next important model after network. It covers the sensing model, coverage ratio calculation, and coverage algorithm implementation. The next model concerns energy consumption, which this study emphasises as it regards mobility, sensing, and energy used for data communication.

The next phase comprises the objective optimisation formulation, which is the focus of this study. The model involves conflicting optimisation problems (multi-objective optimisation problems (MOOPs)) that simultaneously minimise or maximise objective functions. Figure 4.1 depicts the model diagram in terms of input, algorithm, and output. In the optimisation process, the main contribution of this study is the production of a hybrid solution involving a multi-objective genetic algorithm (MOGA), adaptive multi-parent crossover, fuzzy dominance, and prospect theory. According to (Chakraborty, 2013), a hybrid algorithm combines more than one algorithm for efficiently solving the problem.This research improved the original Non-Dominated Sorting Genetic algorithm (NSGA-II) by combining the existing algorithm with fuzzy dominance-based decomposition and adaptive multi-parent crossover algorithms. The proposed solution can balance conflicts concerning node deployment objectives in a stochastic system.

The final workflow evaluates the proposed solution. At the end of this phase, the analysis includes coverage rates, energy consumption rates, Pareto metrics, and execution time using specific tools. The solution is also compared with the existing methods' results to determine the proposed solution's effectiveness.

4.1.1 Network Model

The network model is composed of three layers: a base layer, a middle layer, and an upper layer. The bottom layer depicts the underwater monitoring region as a 3D cube. Mobile sensor nodes with acoustic communication capabilities are randomly placed for the purpose of monitoring underwater occurrences. In the middle layer, the centre of the cube is connected with a sea surface sink with acoustic and radio waves communication. The sea surface sink communicates with the mobile sensor nodes in order to gather data from a distant point in the top layer. The primary goal of the 3D mobile sensor nodes is to gather event data and transmit it to the sea surface sink. The acoustic signal has a restricted transmission range; therefore, the gathered data is sent through multiple hops to the sink. Nonetheless, if the sink is within communication range of the mobile sensor nodes, the packets can be transmitted through short-distance single-hop transmission.

The sea surface sink gathers data from the mobile nodes and sends the data to the end user situated at an offshore base station remotely. The proposed network model has the following properties: first, the surface sink and mobile sensor nodes (MSNs) are able to determine their position using the method described by (Beniwal, Singh, & Sangwan, 2016) for UWSNs. Second, MSNs in a UWSN are homogeneous in terms of data processing capacity, buffer size, baseline energy, and communication range. Third, due to water currents, the surface sink and MSNs move arbitrarily in longitudinal and transverse directions. Fourth, the connection between MSNs is asymmetrical, and they can interact with adjacent MSNs in a two-way fashion only when they are within communication range. Additionally, the relative distance between two MSNs may be determined from the signal intensity values received at the receiver. Fifth, both the surface sink and the offshore base station are well equipped in terms of energy and data processing. Furthermore, as compared to acoustic data transmission in UWSNs, the communication latency between the surface sink and offshore base station and the local user is considered to be minimal.

4.1.2 Coverage Model

The sensing model determines UWSN coverage. Two different types of sensing models are commonly used for simulating sensor performance: binary and probability models.

4.1.2.1 Sensing Model

The probability model calculates the chance of a target occurring in a given region; the value varies between 0 and 1. However, there is no such region in the binary model; a target is either discovered or undetected. Because each sensor will have a specific region where it may fail to identify the target, this research focuses only on the probability model. If there is a sensor node S at initial location (x_i, y_i, z_i) that moves to final position (x_p, y_p, z_p), this study assumes that the sensing range of S is a circular area with a radius R_s, and centred at (x_i, y_i, z_i). R_s is called the 'sensing radius' of node S. Distance between target and sensor (in 3D area) can be calculated as below in equation 4.1:

$$d_s = \sqrt{\left(x_i - x_p\right)^2 + \left(y_i - y_p\right)^2 + \left(z_i - z_p\right)^2} \tag{4.1}$$

In the probability model, there are two critical distances for a sensor. The first one is

R_s, which is the same as in the binary model. If the distance between the target and the sensor is less than R_s, the target can be detected by the sensor with a probability of 1. The second critical distance is R_u, which stands for the uncertain range. If the distance between the target and the sensor is in the range between R_s and R_s+R_u (refer to Figure 4.2), the probability that the target will be detected by the sensor is related to the distance between them.

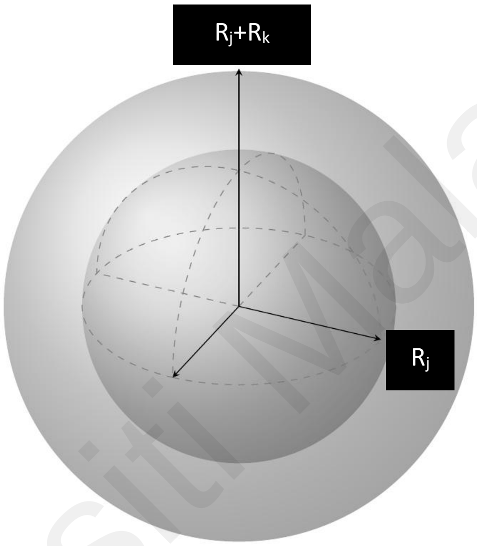

Figure 4.2: Sensing Range (m) of Sensor Nodes

If the distance between the target and sensor is farther than R_s+R_u, the target will not be detected by the sensor. The mathematical expression of probability model is as follows (equation 4.2):

$$P_m = \begin{cases} 0, & \text{if } (R_s + R_u) < d_s \\ e^{-\lambda\alpha\beta}, & \text{if } (R_s < d_s \leq R_s + R_u) \\ 1, & \text{if } d_s \leq d_{ij} \leq R_s \end{cases} \tag{4.2}$$

When the distance between the target and the sensor (d_s) is closer than R_s, both the binary model and probability model give a detection probability of 1. When the distance is longer than R_s and shorter than R_s+R_u, the binary model gives a 0 probability of being detected, but the probability model will have a gradually decreasing probability. When the distance is longer than R_s+R_u, both of the sensing models will give a detection probability of 0. The probability model is more realistic than the binary model and the binary model is the simplified version of probability model when R_u is 1.

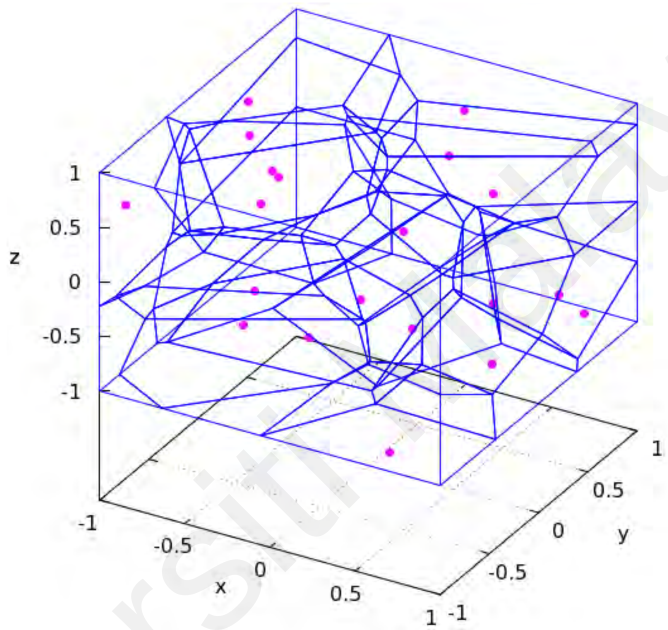

Figure 4.3: Voronoi Diagram

Each sensor can generate a local Voronoi diagram based on the information it receives from its neighbouring sensors. Each sensor adjusts its sensing range according to its distance to the vertices of its Voronoi subarea which provide different coverage. The small red dots in Figure 4.3 represent nodes s_i, and each polyhedron having a different shape indicates that the cube is divided into Voronoi polyhedral units of different sizes. In addition, each Voronoi body unit contains only one node (e.g. red dots as shown in Figure 4.3). As a result, the number of nodes s_i is the same as the number of Voronoi body units after division. The sensor's sensing radius is equal to the distance from the sensor to the vertex of the farthest subarea (d_i).

$$R_s = Min\left[R_{smax}, Max(d_i)\right] \tag{4.3}$$

In this way, according to equation 4.3, each sensor will cover the entire effective Voronoi subarea it belongs to. Therefore, the entire sensing field will be covered.

4.1.2.2 Coverage Calculation

The definition of coverage ratio (R_{Cov}) is the ratio of area that can be covered by sensors cooperatively (A_{Cov}) over the entire sensing field A_{Sum} as stated in equation 4.4:

$$R_{Cov} = \frac{A_{Cov}}{A_{Sum}} \tag{4.4}$$

The maximum coverage ratio is 1. When R_{Cov} is 1, the space or area is completely covered by the sensors. An area is considered to be covered if it can be covered by at least one sensor node or by the joint detection of several sensors. Assuming that there are N sensor nodes in the entire sensing field, the joint detection probability for a certain area can be calculated as below (equation 4.5):

$$P_{joint} = 1 - \prod_{s=1}^{s=N}(1 - P_m) \tag{4.5}$$

In the probability model, P_{joint} will be any value from 0 to 1. The colour bar on the right side as shown in Figure 4.4 indicates the probability. A threshold (P_t) is required for judging whether an area is covered. If P_{joint} is bigger than P_t, the area is covered; otherwise, the area is not covered. The value of P_t depends on the application requirement. Due to the random positions of sensors, the areas they cover will be irregular. According to Figure 4.4, yellow nodes represent the grid points in the monitoring area, whereas the green nodes are all the connected neighbours of the purple node.

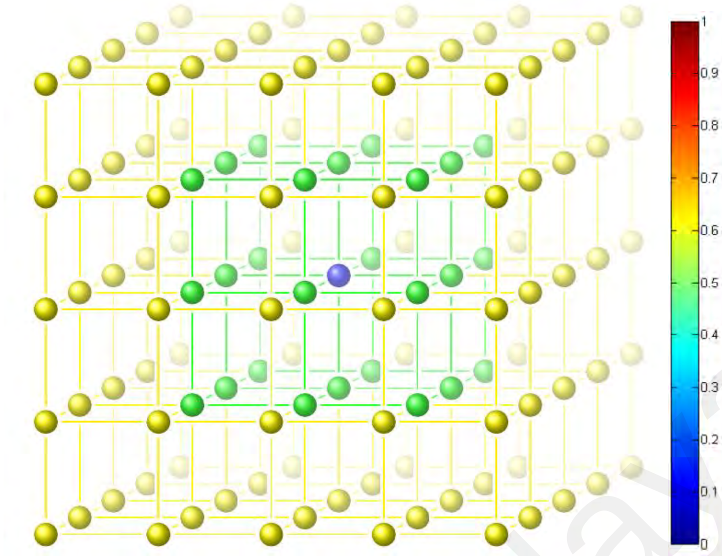

Figure 4.4: Three-Dimensional Coverage Ratio for Probability Sensing Model

4.1.2.3 Coverage Rate Algorithm

In the deployment and coverage processes of the sensor nodes, the network nodes should satisfy the sensing probability $P_{joint} \geq P_t$ of the grid points in the monitoring area as much as possible so that full coverage can be achieved. The algorithm procedure for node deployment to ensure the monitoring area coverage based on sensing probability is as follows:

Algorithm 1: Coverage Ratio

1: *Start*
2: *Initialize:*
  2.1 Number of sensor nodes, sensor-no;
  2.2 Number of covered grid points, Cov_no;
  2.3 Expected network coverage rate, Exp_R_Cov;
  2.4 Maximum total number of sensor nodes, max_sensor_no;
3: Calculate all grid points of the area to be monitored
4: Place the sensor nodes on the grid points that satisfy max_sensor_no
5: The number of sensor nodes is sensor_no++
6: $\forall g \in G$, if coverage of grid point = k, then grid point g has already met the k-coverage requirement; Flag(gv) = 1, Cov_no++
7: Calculate the network coverage rate R_Cov
8: The number of sensor nodes is sensor_no++
9: *End*

When the network complies with a coverage requirement for the k-coverage, or when the sensor nodes deployed already approach the maximum value of the setup, the algorithm stops working. Every iteration of an algorithm would complete the deployment of the sensor node in the monitoring region. Each iterative procedure deploys a single sensor node at a specified grid point.

4.1.3 Energy Consumption Model

The MOGA-AMPazy algorithm considers the energy consumed for mobility, sensing and data communication processes.

4.1.3.1 Energy Consumption in Mobility

The main objective of the proposed algorithm is to improve the coverage area by relocating the underwater mobile sensor nodes. During the relocation process, each sensor node will consume energy as it moves or turns. When the weight of sensors and speed are constant, this study considers the power consumed in relocating to be constant as well. In the proposed solution, because each sensor travels once for each round of the simulation, the moving distance $D_{mov(i)}$ of the sensor s_i is equal to the distance between the initial (x_i, y_i, z_i) and the final (x_p,y_p,z_p) positions. Additionally, energy will be consumed every time a sensor node turns. The energy consumed for turning (E_{turn}) is linearly related to the angle by which the sensor node turns. A sensor can turn either clockwise or counterclockwise, and it will always choose the smaller angle to turn to its destination direction. D_{movSum} and T_{sum} can be calculated by equations (4.6) and (4.7):

$$D_{movSum} = \sum_{i=1}^{N} D_{mov(i)} = \sum_{i=1}^{N} \sqrt{\left(x_i - x_p\right)^2 + \left(y_i + y_p\right)^2 + \left(z_i - z_p\right)^2} \tag{4.6}$$

$$T_{sum} = \sum_{i=1}^{N} \Delta T_i \tag{4.7}$$

D_{movSum} and T_{sum} are the total distance travelled and the total angle turned by the sensor respectively. Therefore, the energy consumed in moving process of sensor s_i is linear with respect to distance it travelled when acceleration is negligible (Qu & Georgakopoulos, 2012) as follows equations (4.8) and (4.9):

$$E_{mov} = K_{mov} \times D_{movSum} \tag{4.8}$$

$$E_{turn} = K_{turn} \times T_{sum} \tag{4.9}$$

where K_{mov} and K_{turn} are coefficients representing the energy consumption rate. Omni-directional small robots built by (Reshko, Mason, & Nourbakhsh, 2002) can be used in mobile sensors. In their research, each robot uses three wheels, which are uniformly fixed along the edge of a round-shaped platform and pointing to three different angles 120 degrees apart. The locations of the wheels form a regular rectangle. In this way, the robot can move straight or turn omni-directionally. (Mei, Lu, Hu, & Lee, 2004) provided the energy model for this type of omni-directional mobile robot. According to their findings, the energy for a robot platform to move 1 meter is 9.34 Joules, when travelling at a constant speed of 0.08m/s. The energy for this three-wheel robot to turn 90 degrees is 2.35 Joules. We assume the sensors in the algorithm move and turn at a constant speed. Therefore, K_{mov} and K_{turn} can be calculated with the parameters provided by equations 4.10 and 4.11:

$$K_{mov} = 9.34 \tag{4.10}$$

$$K_{turn} = 0.0261 \tag{4.11}$$

The total energy for moving the sensors will be the sum of the two types of energy listed above:

$$E_{Mobility} = E_{mov} + E_{turn} \tag{4.12}$$

4.1.3.2 Energy Consumption in Sensing

Adjustable-range sensors are used because when they shorten their sensing radius, the power consumed will be lower, thereby yielding energy savings. Different energy models are used for analysing the relationship between a sensor's sensing radius and its energy consumption, and they depend on the characteristics of the sensing device. Typical models are the linear, quadratic, cubical and quadruplicate models. The most common are the linear and quadratic models by (Cardei, Wu, Lu, & Pervaiz, 2005) and (Dhawan, Vu, Zelikovsky, Li, & Prasad, 2006). This study applies the quadratic model where in this model, sensor (s_i) energy consumption depends quadratically on the sensing radius as stated in equation 4.13.

$$E_{Sensing} = K_{sense} \times R_s^2 (R_s \leq R_{max}) \tag{4.13}$$

where $E_{Sensing}$ is the power consumed in sensing, and K_{sense} is a constant for power consumption rate.

4.1.3.3 Energy Consumption in Data Communication

During the data communication process, the sensor nodes spend much more energy while transmitting and receiving data. At this point in time, the remaining node enters an idle or sleep state. For each cycle of the data transmission process, the energy level of sensor nodes changes due to the amount of energy spent in the cycle of data transmission. The energy consumption is described as the energy used by the nodes to send data. It involves several parameters, such as data transmission distance, data transmission time and minimum power packets that can be received. The energy consumption is expressed as follows (equation 4.14):

$$E_{DC} = D_t \times P_{min} \times T(d) \tag{4.14}$$

where d is the data transmission distance, D_t is the data transmission time, and P_{min} is the minimum power packets that can be received. Transmission distance $T(d)$ is measured based on the underwater acoustic signal attenuation model. Therefore, the total energy consumption of each mobile sensor node (E) is calculated by equation 4.15:

$$E = E_{Mobility} + E_{Sensing} + E_{DC} \tag{4.15}$$

4.1.4 Differential Fuzzy Dominance (DFD) Function

In order to identify the location of high event detection nodes, the DFD function must be combined with the proposed method, which is designed to utilise the Pareto front from a multi-objective representation to choose the optimal solution for high event detection in deployed node areas. The best individual solution (i.e. that with the lowest ranking value) must be identified from among from the ranked population and conditionally added to the alpha set ($\alpha - set$). Each solution may be allocated a single, smooth dominance to represent the degree of domination of the solution after establishing its membership role. A solution with a lower fuzzy dominance value is a better approach. As a result, sorting alternatives according to their fuzzy dominance values may assist in rapidly locating solutions around the Pareto front. In this research, the best K solutions are found and selected for each solution generation round using a fuzzy dominance sorting method.

Algorithm 2: Differential Fuzzy Dominance (DFD)

1: *Start*
2: *Initialize:*
  2.1 Workflow W;
  2.2 Set of instances, I;
  2.3 Rank, R;
  2.4 Solution set, S;
3: Determine the scaling order of tasks, $R \leftarrow UpRank(T)$
4: Determines the tradeoff solutions of tasks and instances
5: Store all possible solutions of assigning tasks, T to n instances
6: Determine the types of assigned instances
7: Create the possible solutions
  7.1 Check whether the type of instance is determined
  7.2 If the type of instance (I_s) has not been decided,
    7.2.1 Generate solution by generated binding I_s to an arbitrary type
  7.3 Otherwise, generate solution when task is assigned to instance
8: *End*

4.1.5 Adaptive Multi-Parent Crossover (AMP)

The main goal of AMP is to create a balance between exploration and exploitation by producing new offspring; if the offspring are too dispersed from their parents, they may be excessively varied, necessitating a longer convergence period. However, if the offspring have a smaller distribution than their parents, early convergence may occur. As a result, this study presents an AMP integrated in GA, in which the number of parents participating in the crossover varies according to the new generation's performance. The number of parents is expanded or reduced depending on the current generation's relative performance to the previous generation. This strategy guarantees that offspring are neither too varied nor too similar to their parents. Adaptive GAs have manipulated control parameters in a variety of ways, including modifying the mutation probability over time, shifting the mutation probability between two fitness-based values, and controlling genetic operators. This research changes the number of parents depending on fitness value. These adaptive approaches have produced additional exploration of the problem space. AMP generates offspring in the direction of a better part of the search space as well as some inferior offspring to increase to more various points.

Algorithm 3: Adaptive Multi-Parent Crossover

1: *Start*
2: *Initialize:* population size, S
3: S randomly generated individual $\rightarrow I$
4: Calculate a fitness value for each individual, $x \in I$
5: Selection operation until all neighbors are explored
6: Non-dominated solutions of Pareto Approximation set, NP
7: Genetic Algorithm, $z \in NP$
8: Repeat step 4 until running time is not reached
9: Return Pareto Approximation set, NP
10: *End*

4.1.6 Prospect Theory Model

Daniel Kahneman and Amos Tversky developed prospect theory in 1979 with the goal of modelling human decision-making in the face of risk. This approach is applicable to the area of subjective probability and enables decision-making based on context and loss aversion in the present condition. Prospect theory enables the MOGA-AMPazy solution to dynamically achieve a balance between exploration and exploitation. This study can solve multi-objective problems with reasonable computational requirements and have dynamism parallelly based on a weighting of probabilities, and proposes the use of prospect theory to make decisions based on success probabilities and risk evaluation.

Prospect theory is a descriptive approach for modelling how human beings make decisions involving risk. Prospect theory suggests that risky decision-making can be viewed as making a selection between many gambles or possibilities. Prospect theory makes an outstanding contribution as a descriptive decision-making model that divides decision-making into editing and judging processes. Editing is the first stage, where individuals collect and process information using 'frameworks' and 'reference points'. In this study, the reference point r_0 is defined to determine the individual's return and loss in the portfolio, r_0 represents the zero return or zero loss that the individual thinks; Evaluation is the second stage; the value and weight functions concerning subjective probability judge information. This study also depends on the subjective probability weight function $\pi(p)$ and the value function to judge information.

In prospect theory, the results are allocated to gains and losses rather than to the actual assets; that is, the value function reflects relative gains or losses. The value function was defined as a power function by (Tversky & Kahneman, 1992), as indicated by equation (4.16):

$$v(r) = \begin{cases} r^{a}, r \geq 0 \\ \\ -\gamma(-r^{\beta}), r < 0 \end{cases} \tag{4.16}$$

where α and $\beta > 0$ measures the curvature of the value function for gains and losses, respectively, and γ is the coefficient of loss aversion. Thus, the value function for gains (losses) is increasingly concave (convex) for smaller values of $\alpha(\beta) > 0$, and loss aversion is more pronounced for larger values of $\gamma > 1$.

Prospect theory assumes that individuals do not weight outcomes by their probability, but by some distortion of probabilities. This distortion of probability is captured by prospect theory's probability weighting function. For the probability, there are two obvious reference points, certainty and impossibility, or a 100% chance and a 0% chance. The weighting function can be parameterised in the following form (equation 4.17) according to the probability weighting function originally proposed by (Tversky & Kahneman, 1992):

$$\pi(p) = \frac{p^y}{(p^y + (1-p)^y)^{\frac{1}{y}}} \tag{4.17}$$

where p is the weighting probability of the distribution of gains or losses and > 0 measures its degree of curvature. Therefore, the prospect-theory utility functions $\pi(p)$ and $v(r)$ can be represented by equation 4.18:

$$PT_U = \sum_{s=1}^{S} \pi(p_s)v(r_s) \tag{4.18}$$

The prospect-theory weight function $\pi(p)$ is used to measure the size of the event's desirability to the result. $\pi(p)$ is an increasing function, $\pi(0) = 0, \pi(1) = 1$, when the probability p is very small, that is $\pi(p) \geq p$. The prospect-theory value function $v(r)$ represents the behavioural value of the result obtained or lost. For a given reference point r_0, because the function reflects the attitudes of different investors toward the gains and losses, the parameters are different.

Algorithm 4: Prospect Theory

Start
Initialize:
    2.1 PT constants;
    2.2 Reference points;
    2.3 Environment constants;
Compute gain and losses of alternatives
Compute expected prospect values of alternatives
Do action
    If risky,
        Distort probability
Ranking alternatives
End

4.1.7 Multi-Objective Optimization Model

The primary goal of a single-objective optimisation problem is to identify a global optimum, which signifies the optimal objective function value. However, the objective function problem is essentially a multi-objective optimisation (MOO) problem. In this study, the difficulty of establishing the objective function is considered within the MOO algorithm's solution. The solution, in general, is not unique due to conflicting objectives. There is a collection of optimum locations (known as the Pareto front) corresponding to a particular trade-off between the values of the objective functions. A selection must be made concerning the Pareto-optimal point for the 'best trade-off'.

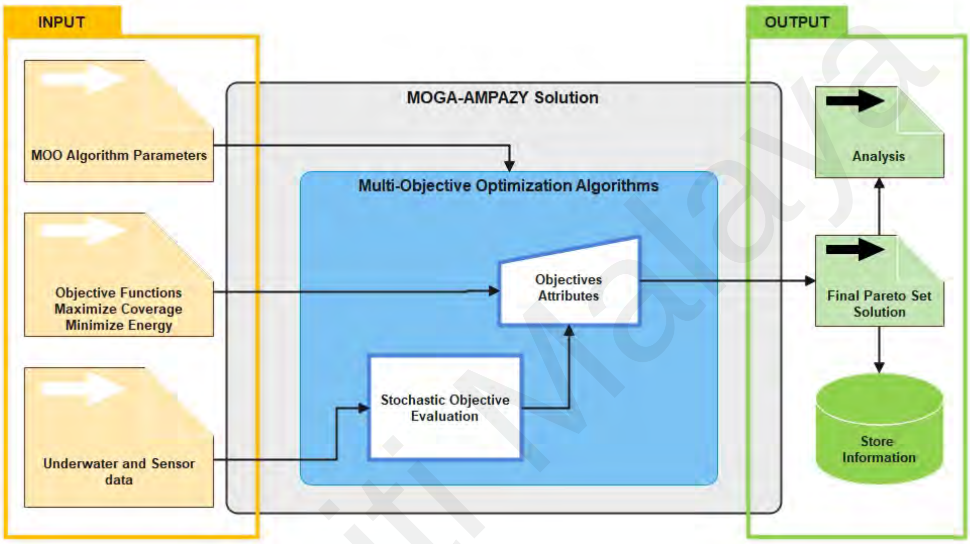

Figure 4.5: Multi-Objective Optimization Model

This study primarily uses genetic algorithms (GA), which are well adapted to multi-objective optimisation problems because they are based on biological processes that are innately multi-objective. GA uses three approaches: criteria selection, aggregation selection and Pareto selection, based on selection techniques and fitness functions. In this research, Pareto selection is characterised by the Non-dominated Sorting Genetic Algorithm II (NSGA-II). The primary concept behind this research is to establish a MOGA-based computational model for the underwater mobile sensor node optimisation principle that produces an adequate trade-off in a surveillance monitoring application.

The original NSGA-II has been modified by introducing a hybrid fuzzy dominance-based decomposition technique into an adaptive multi-parent crossover genetic algorithm. The algorithm introduces a fuzzy Pareto dominance concept to compare two solutions and uses the scalar decomposition method only when one of the solutions fails to dominate the other in terms of a fuzzy dominance level.

A mobile sensor node deployment solution allocates available nodes in an underwater surveillance network to target surveillance. Given a known target, initial node position and velocity must be chosen and surveillance regions assigned such that a given set of objectives is optimised. Considering that multiple criteria are present in most real-world problems, multi-objective optimisation (MO) problems are universal. As the name indicates, multi-objective optimisation problems involve multiple objectives (often conflicting) that must be optimised simultaneously. This results in a group of alternative solutions that must be considered equivalent in the absence of information concerning the relevance of the others.

As discussed in Chapter 3, the central controller in the surveillance network is responsible for high-level planning rather than direct control of mobile sensor nodes. As presented in Section 3.4, the mobile sensor node deployment problem can be formulated as a MOO problem. GAs help find approximate solutions for commercial and military applications that require quick, adaptive, automated and self-learning techniques (Barolli, Koyama, & Shiratori, 2003);(Hu & Yang, 2004). This study applies a topology control technique based on prospect theory, fuzzy dominance, adaptive multi-parent crossover, and genetic algorithms to efficiently deploy mobile nodes. Simulation outcomes of this study demonstrate that the MOGA-AMPazy solution converges to a stable distribution with sufficient coverage and is resilient to node losses. After many nodes are lost or disconnected, it may dynamically rearrange the network architecture. The findings further demonstrate that the suggested method achieves uniform spatial coverage despite ambiguity in identifying the precise positions of neighbours (e.g. due to worsened underwater conditions). The aim of the proposed optimisation method is to deploy nodes in 3D positions (that is, to identify sensor node coordinates) to meet the anticipated objectives.

4.1.7.1 Algorithm Objective and Fitness Function Design

This thesis has formulated the sensor node deployment task as an unconstrained multi-objective optimisation (MOO) problem. The aim is to find a sensor node deployment arrangement in the 3D space that satisfies the required objectives. There are two main objectives of the MOGA-AMPazy solution. The first objective of the algorithm is to maximise the coverage ratio, as defined in Section 4.1.2 and equation (4.4). The second objective is to minimise sensors' energy use in mobility, sensing and data communication, as defined in Section 4.1.3. In MOGA-AMPazy, the fitness function is designed according to the proposed objectives. In optimisation problems having multiple conflicting objectives (such as minimising energy while maximising coverage ratio), the optimised fitness function will provide the Pareto-optimal front, which constitutes the best compromise between the objectives. In this research, because final sensor positions need to be optimised, sensor node coordinates are used as input variables for the GA.

The method is implemented to produce the fitness function's minimal value. Thus, maximising the coverage ratio involves minimising the objective function using the uncovered area ratio (equation 4.19):

$$R_{unCov} = 1 - R_{Cov} \tag{4.19}$$

where R_{Cov} is obtained from equation (4.4). The MOGA-AMPazy solution aims to minimise both the average travelled distance and the uncovered area ratio. It should be noted that there is a trade-off between the two objectives. Once the N sensors are deployed, the sensors remain at their positions before the algorithm is executed. The larger the coverage needed, the longer the movement of the sensors will be. Conversely, such long movements of the sensors do not guarantee larger coverage. Because a multi-objective genetic algorithm can achieve the best trade-off between coverage and travelled distance, this study defines the fitness function of coverage as:

$$Minimize\ F_1 = \sum_{i=1}^{N} R_{unCov(i)} \tag{4.20}$$

Energy consumption must be minimised to maximise sensor lifetime. Thus, the second objective is to decrease the amount of energy used by sensors. The method proposed by this research focuses on sensors' energy consumption during mobility, sensing and data transmission. The method does not extend to the amount of computation sensor nodes require, so there is little communication overhead produced. This algorithm requires only sensor movement after deployment. Thus, the energy consumed during sensor movement is a short-term consideration; however, it far exceeds long-term sensing energy consumption. The total energy consumed by the sensors is obtained from equation (4.22). In an underwater wireless sensor network, relocation optimisation is required if any of the sensors fail. Therefore, this study defines the energy consumption fitness function as:

$$Minimize\ F_2 = E_{total} = \sum_{i=1}^{N} E_i \tag{4.21}$$

4.1.7.2 Algorithm Description

The aim of the proposed solution is to maximise the coverage rate by reducing the gaps and maximising the connectivity among the sensor nodes. If the mobile sensor nodes are deployed randomly over the region of interest, r is the sensing range of the sensors. The proposed genetic algorithm starts with an initial random population (the distribution of the initial nodes). Then, the objective function in each iteration the constraints' satisfaction rate in each iteration, and the new solution (population) is improved after each iteration of the algorithm. This improvement is carried out through the operators (crossover and mutation). A stopping criterion is used to stop the execution of the algorithm. Indeed, the algorithm starts by initialising the length and the width of the area, the number of mobile sensor nodes, the number of targets, the chromosomes, and the random locations of the chromosomes.

First, the coverage ratio of mobile sensor nodes is calculated. Then, although it did not reach the required fitness or the maximum number of iterations, this study evaluates the fitness of the chromosomes, ranks the chromosomes and computes the fitness, does a crossover and mutation on chromosomes, updates the value of the chromosomes, and evolves the next generation of chromosomes.

The proposed autonomous 3D algorithm runs for every mobile sensor node operating in a UWSN. It provides speed and direction guidance towards a uniform spatial distribution while maximising network coverage and minimising energy consumption. The mobile sensor nodes running a 3D GA can adapt to changing conditions such as addition, loss or malfunction. The goal is to find chromosomes with speed and movement directions that minimise the fitness function. Each node transmits its location and number of neighbours to its neighbouring nodes. The degree for a node N_i is defined as the total number of nearby neighbours within its communication range (denoted as D_i). Each node calculates its next best location and speed based on this local information from its nearby neighbours. The fitness function is an important element that considers nodes' current velocity vector and neighbours' relative positions to adjust movement speed and direction, resulting in a better subsequent position, where possible. The new value of the velocity vector depends on the number of neighbouring nodes within its communication range as well as their distance. A smaller fitness value indicates a better position for a node. The mobile sensor nodes will move to a given new location only if it is characterised by better fitness; otherwise, the sensor maintains its current location.

A chromosome has genetic information for solving the problem in a 3D space. Each chromosome has a fixed gene size, determined by the number of sensors in the network. Each gene has an $x, y, andz$. The mobile sensor nodes can move freely in 3D Cartesian space. A chromosome is defined as representing a speed and direction for each of the three dimensions defined in Cartesian space, as shown in Figure 4.6.

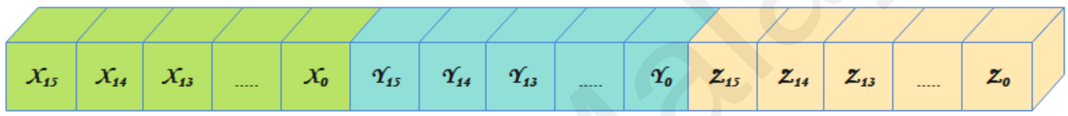

Figure 4.6: Chromosome in Three-dimensional Genetic Algorithm

According to Figure 4.6, the left-most bit X_{15} denotes direction along the X axis, with 1 representing the positive direction and 0 the negative direction. The remaining 15 lower-order bits ($X_{14} \ldots X_0$) represent a scale factor (n) in the range of [0,1]. The different genes of the chromosome represent a binary digit that resembles the value of the position on the X, Y and Z axes. The size of the chromosome population is chosen based on two factors: the area of the region of interest (RoI) and the initial configuration of the network.

After the initialisation, each chromosome's fitness (i.e. the goodness of the solution) is evaluated using the fitness function. The fitness or the formulation of the objective depends on the characteristics of the problem. The fitness function is used to choose the best-fitting chromosomes to reproduce the algorithm's next generation of solutions. The fitness function calculates the maximum number of the covered targets by each mobile node. Overlapping redundancy is prevented by the fitness function among the coverage regions of the deployed mobile nodes.

Reproduction is the next process, which comprises four steps: selection, crossover, mutation and solution acceptance. Fitness is used to rank the chromosomes and perform parent selection according to the participation degree of each chromosome in the fitness function to produce new solutions. However, this also means that fewer fitness members will have a chance to be selected. The selection is performed on two chromosomes to reproduce two new chromosomes. After selecting the chromosomes, a crossover operation is performed between a pair of parent chromosomes by selecting a random point and exchanging genes after this point. This research chooses two random crossing points. The offspring inherits elements positioned between the two crossover points of the first parent. These elements occupy the same positions and appear in the same order in the offspring.

The selection and crossover procedures may result in identical chromosomes, causing the algorithm to cease generating new individuals. It may obstruct average fitness progress by restricting individuals within a local optimum. This issue is resolved by performing a mutation operation on a randomly chosen gene and altering its value. Mutation expands the search space in order to prevent premature convergence or the extinction of variety while introducing innovation into the population. The mutation takes place by reverting the two genes' locations. Often, each gene is represented by a bit; mutations are carried out by randomly flipping a bit in the chromosome. Two new chromosomes are formed as a result of crossover and mutation. In the end, they will be recognised as a new population if they are better than their parents. The halting criteria specifies an upper bound on the number of iterations. Algorithm 5 summarises the algorithm underlying the proposed solution.

Algorithm 5: MOGA-AMPazy Solution

Input: Population size (N_{pop}), Recombination probability (P_c), Mutation probability (P_m), The number of maximum generations (G_{max})

Output: The optimum solutions found in P

1: *Start*

2: *Initialize:* Randomly generate an initial population P of Npop individuals, and create an empty elite sets E

2.1 Compute the Euclidean distances between any two weight vectors;

2.2 Find the T closest weight vectors to each weight vector;

2.3 Generate an initial population by uniformly randomly sampling from the search space;

2.4 Set gen=0 for all i=1, . . . , N;

3: *Evaluation:* For each individual in the population, compute all objective function values F_1 and F_2

4: *Fitness Assignment:* Assign each individual a fitness value F_{ij}

5: *Update elitist:*

5.1 Add the non-dominated individuals in E;

5.2 Considering all individuals in E, remove the dominated ones in E;

5.3 If the number of non-dominated individuals in E is larger than Emax, randomly discard excess individuals;

5.4 Rank population by Fuzzy-Pareto-Dominance(FPD) ordering of fitness vectors of the individuals in the population;

6: *Selection* of sub-problems for search:

6.1 Select $N_{pop} - N_{ps}$ individuals from the population to form a new population using the binary tournament selection and random select Nps individuals from E to form a new population, where Nps = $N_{pop} - N_{ps}$ and p, is a selection proportion. If N_{ps} is greater than the number NE of individuals in E, let $N_{ps} = NE$;

6.2 Calculation of fuzzy dominance level;

6.3 Update of solutions - Add best pn of population individuals, according to FPD ordering ranking values, to the habitat $0 \leq p \leq 1$;

6.4 Select $(1 - p)n$ pairs from population by roulette-wheel selection, using the negated logarithms of the ranking values of the ranked population for selection (lower ranking value counts better), and put these pairs into the mating pool;

7: *Recombination:* Perform the uniform crossover operation with a recombination probability p_c

8: *Mutation:* Apply the simply mutation operator to each gene in the individuals with a mutation probability p_m

9: *Stopping criteria:*

9.1 If satisfied, stop the algorithm;

9.2 If not, perform adaptive multi-parent crossover;

10: gen = gen + 1

11: Go to Step 3

12: *End*

4.2 MOGA-AMPazy Process

The proposed solution has been illustrated using a decomposition diagram (Figure 4.7) and flow diagram (Figure 4.8) to illustrate the flow of steps required to execute the MOGA-AMPazy solution.

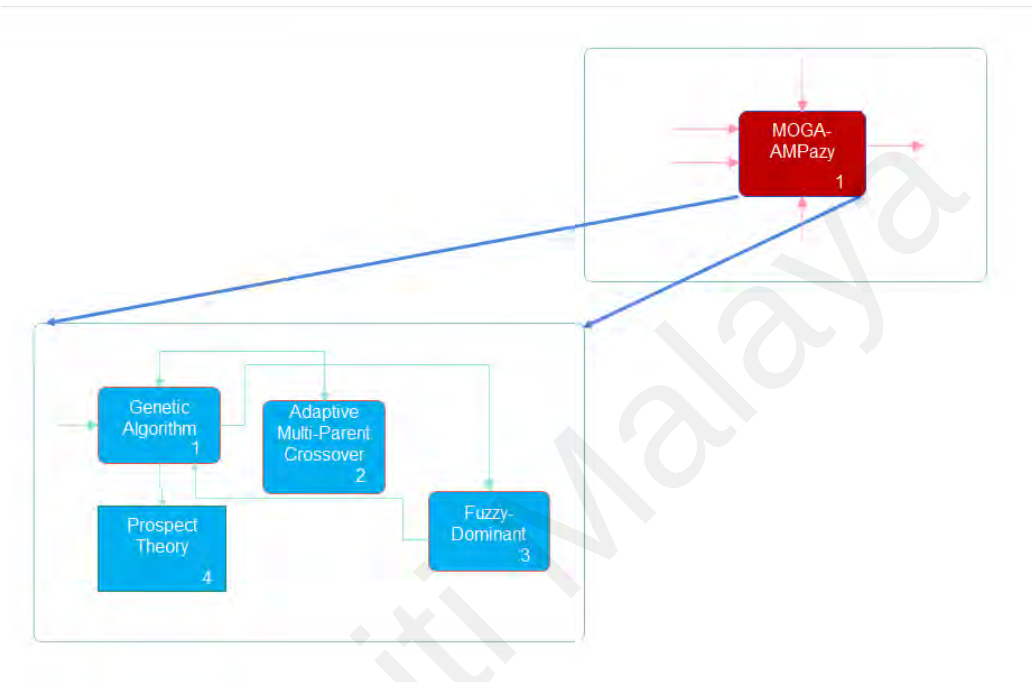

Figure 4.7: Decomposition Structure of the Proposed MOGA-AMPazy Solution

The process starts with random generation of Q solution initial population, followed by evaluation objective function process. Then, the fuzzy dominance process is implemented by combining solution Q with parent population P from the previous iteration. Later, the proposed algorithm sorting the combined population $(P + Q)$ and executes the selection of solution Q. If the set criteria of system termination are achieved, the system execution will be ended. Otherwise, adaptive multi-parent algorithm is applied to provide a balance between exploration and exploitation by providing new offspring. The adaptive multi-parent algorithm manipulates control parameters by changing the mutation probability over time, changing mutation probability between two values based on the fitness value, or adjusting genetic operators over time. The number of parents varies based on the fitness value. The adaptive multi-parent algorithm is executed until the solution search is no longer overlapping.

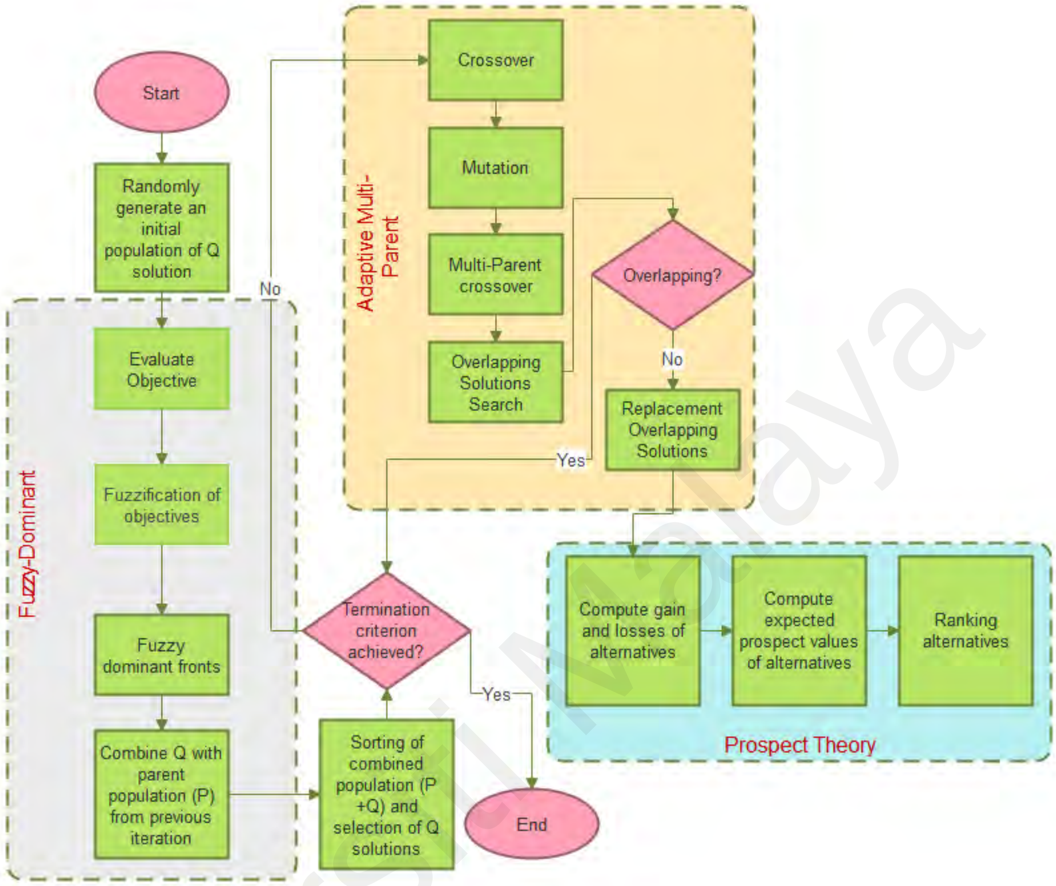

Figure 4.8: Flow Diagram of the Proposed MOGA-AMPazy Solution

4.3 Mobility Model

The mobility model is essential in node deployment because it describes how sensor nodes physically move in an underwater environment. This study has applied the Meandering Current mobility model developed by (Caruso, Paparella, Vieira, Erol, & Gerla, 2008), considering the connectivity and coverage of mobile sensor nodes in the underwater environment. According to (Caruso et al., 2008), sensor speeds are similar and strongly correlate with those of neighbouring sensor nodes. The placement of all sensor nodes becomes uniformly distributed over time; the process is represented as:

$$J_i, P_i, T_i, 0 \leq i < n \tag{4.22}$$

where n is the number of placement rounds, T_i is the time for each round, J_i is the number of nodes, and P_i is the distribution node used to place the nodes. The node distribution for this study is normal. Following the node deployment process, the network is connected. The network of mobile sensor nodes is represented by $G = V(t), E(t)$ where $V(t)$ contains sensor nodes moving in a cube-shaped area at a time (t), $E(t)$ represents communication between sensor nodes.

Based on the Meandering Current mobility model, some of the sensors in the middle of the area will move to a specific coordinate with a speed of S_m up to the maximum value of the bounding box $B^G_{max}(t)$. The rest of the nodes will only be at a particular position $B^G_{max}(t)$. The width of the bounding box is ($B^G_{max}(t)$- $B^G_{max}(t)$), and if a sensor moves more slowly or slightly faster than another sensor node, the sensor will lose connectivity due to limited communication range.

4.4 Distinguishing Features of MOGA-AMPazy

The literature review revealed that many researchers have proposed mobile sensor node deployment solutions, including multi-objective node deployment solutions to address efficient mobile sensor node deployment in UWSNs. However, the proposed solutions analysed in the literature review are inadequate to solve multi-objective node deployment in UWSNs. In this section, we highlight how MOGA-AMPazy addresses the shortcomings of existing monitoring solutions. The following four subsections describe the distinguishing features of MOGA-AMPazy that make it a more efficient and scalable solution for performing mobile sensor node deployment that solves conflicting objectives.

4.4.1 Scalability

Scalability is an important consideration when making a judgement about a mobile sensor node deployment solution. Water is a highly dynamic medium that presents a challenging environment for the propagation of acoustic signals due to its high spatial and temporal variability. Because these characteristics significantly impact the deployment performance of mobile sensor nodes, good mobile sensor node deployment system should accommodate changes in an underwater environment without degrading performance.

4.4.2 3D Environment

In the proposed solution, the existing approach has been modified to deploy mobile sensor nodes in a 3D environment, which is more realistic and accurate for locating sensor nodes. The 3D deployment algorithms measure the exact orientation and distance between neighbouring nodes and use the information to deploy nodes. The solution uses network connectivity information to estimate node locations in the 3D space.

4.4.3 Prospect Theory with Fuzziness

Fuzzy logic systems are good at generalisation; they have distinctive characteristics that facilitate handling non-statistical uncertainties and fuzziness. However, under certain conditions, the design and development of large or highly complex systems are overcomplicated for human operators. Prospect theory remains one of the most effective (and most explored) ways to deal with uncertainties, especially stochastic uncertainty. The fusion of prospect theory with fuzzy logic controllers has been shown to be a powerful tool for real-world applications such as finance and weather forecasting. Both paradigms can work in collaboration and complement each other. As a result, a probabilistic fuzzy logic system can handle an extensive range of uncertainties.

4.4.4 Decision-Making

The present work combines seven components into a hybrid method that is able to obtain optimal decision-making whilst being resistant to stochastic and non-stochastic uncertainties, randomness, and fuzziness. Experiments were carried out to evaluate the performance of the algorithms and compare the results of the proposed solution with the existing methods.

4.5 Conclusion

This chapter presented the proposed multi-objective genetic algorithm based on adaptive multi-parent crossover and fuzzy dominance (MOGA-AMPazy) in the form of pseudo-code that helps elucidate the understanding of solutions.

The solution comprised seven major components that are sufficiently flexible to receive the data for processing. Improved and integrated operations in the proposed algorithms enable coverage maximisation and energy consumption minimisation with a shorter execution time for mobile sensor node deployment in UWSNs. The process flow of the implemented solution has been described; execution time, coverage rate, energy consumption and the Pareto metrics have been identified for performance evaluation. Finally, the distinctive features of the proposed solution have been discussed to determine competitiveness.

The next chapter presents the implementation details of the MOGA-AMPazy solution and associated algorithms. Likewise, various evaluation methods were used to verify the effectiveness of the implemented solution and algorithms. In Chapter 5, the data collected from the implemented solution and algorithms will be presented and used for critical analysis.

CHAPTER 5: EXPERIMENTAL DATA AND EVALUATION OF MOGA-AMPAZY METHOD

This chapter discusses the experimental setup and data collection methods used to effectively evaluate the proposed MOGA-AMPazy method in the underwater environment. The data is collected to measure coverage rates, energy consumption, convergence and diversity, and execution time. A detailed discussion of the results is in chapter 6. This chapter is structured into four main sections. Section 5.1 explains the experimental setup and describes the procedure. Section 5.2 reports the process of data collection for the proposed solution. Section 5.3 details the performance evaluation metrics and the data collected for evaluating the proposed algorithms. Finally, Section 5.4 presents the conclusion for the chapter.

5.1 Experimental Setup

This section presents the details of the experimental setup of this study. This research considers several simulations to evaluate output by checking the validity of the proposed algorithm. The study conducted the simulations on a computer with a 2.1 GHz-CoreTM i7 processor, 64-bit Windows 10 operating system, 1 Mb cache, and 8 GB RAM using MATLAB R2018b. The object-oriented environment in MATLAB can keep private meta-data and combine activities with agents. The system environment consists of a sensor, communicator, node, and network classes. The sensor contains a string identification, a value gathered from the environment, and essential functions to examine new data. This class can emulate a given sensor's behaviour, consisting of the power specifications and configuration, to provide data.

Furthermore, the system has a communicator class providing a basic implementation of a communication modem. The communicator consists of sending and receiving queues, power requirements, and success rates that serve as an intermediary for the type of device and medium used for transfer. The third class is a node that operates as a container of communicators and sensors. Nodes store information concerning their own position, list of neighbours and their location, identification value, and 3D coordinates. The algorithm is executed within each node, including all basic support functions for the decision-making process. Nodes provide information concerning sent messages and power consumed for external classes and scripts. A network class of this system acts as a container for all other containers. The network class contains nodes, supports a periodically varying connectivity matrix and list of positions, and stores simulation meta-data. It also holds all sent and received messages, power consumption, distance travelled, and algorithm execution time per node.

A network stores the locations of nodes and their communication paths based on location and range information. The system provides an ID, location, and neighbours' list as nodes are formed. The nodes produce a reference to the version of the MATLAB code that runs in an environmental implementation. Its implementation strengths are flexibility, ease of continuing functionality, enhanced maintenance, and efficiency achieved by scaling down total references. A series of scripts and helper machine functions are set up whereby non-damaging changes simplify calculations. The scripts and operations are divided into three categories: initialisers, modifiers, and run-time. Initialisers are vital to initialise a data field (node positions, message queues) and determine constants for the simulation. Simultaneously, modifiers are implemented as functions to produce temporary partial copies of matrices and return the result. Modifiers are helper functions and scripts that handle storage systems. Finally, the run-time operates to run the simulation and has algorithmic helpers for the process of generating messages, measuring how to respond, and organising all run-time tasks. During the experiment, the nodes move randomly to resemble real-world dynamics. Figure 5.1 shows the overall implementation of the events.

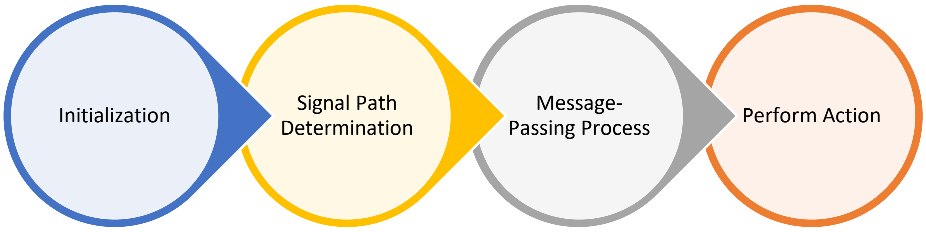

Figure 5.1: Flow Diagram of the Proposed MOGA-AMPazy Solution

5.2 Data Collection for MOGA-AMPazy Solution

The developed solution has been validated by comparing the output obtained against the implemented experimental results. As discussed in Chapter 1, experimental data were collected to analyse the following four parameters: convergence metrics, coverage rates, energy consumption rates, and execution time. At the same time, the collected data can also be used for analysing other parameters. In this section, a set of experiments was conducted to demonstrate the simulator's performance and its model. Each simulation was run 30 times, and the average results are estimated to remove random errors.

The experiments consist of two tests conducted with different nodes, as shown in Table 5.1. The setting guarantees at least acoustic neighbours for each node. One node cannot have more than six acoustic neighbours.

Table 5.1: Tests Parameters

Tests	Field size (m^3)	Number of Nodes
A	500 x 500 x 500	15
B	1000 x 1000 x 1000	200

Underwater test instances of sensor nodes specifications are provided in Table 5.2. It consists of several sensors deployed in RoI, their sensing and communication range, initial energy, transmission power, topology type, data rate, frequency and the distance between sensor nodes and sink.

To establish that the suggested solution is utilized its best performance, multi-objective

Table 5.2: UWSN Test Instance

Parameters	Value
Number of Sensor nodes	200
Sensing Range	300 m
Communication Range	200 m
Initial Energy	15 Joule
Initial energy of sink	12 kJ
Maximum hop distance	85 m
High transmission power	0.99 W
Low transmission power	0.93 W
Packet receiving power	0.25 W
Packet length	1500 bits
Topology	Random
Data rate	16 kbps
Frequency	40 KHz
Maximum distance between sensor nodes and sink	0.81 km

genetic algorithm parameters, comprising the number of generations, population size, crossover rate, and mutation rate, have been set accordingly. The specifications with maximum and minimum values of the experiment are indicated in Table 5.3.

Table 5.3: Parameters of the Proposed Multi-Objective Solution

Parameters	Minimum Value	Maximum Value
Number of generations	100	250
Population size	120	200
Crossover rate	0.1	1
Mutation rate	0.1	0.5

For each mobile sensor node (MSN) engaging in UWSNs, the suggested 3D travel autonomously presents direction and velocity guidance for uniform spatial distribution while optimising network coverage and minimising energy consumption. MSNs offer a 3D multi-objective genetic algorithm that reacts to conditions such as addition, loss or malfunction. The objective is to identify a group of chromosomes with directions and velocity to decrease the fitness function. Each mobile node transmits its location and number of neighbours to neighbouring nodes. Based on local knowledge from its close neighbours, each node decides its next best position and velocity. An essential aspect of the proposed solution is the fitness function, which deals with the node's present velocity vector and neighbours' relative positions to adjust velocity and plan direction, resulting in a better subsequent location.

5.3 Performance Evaluation Metrics and Data Collection

This section evaluates MOGA-AMPazy capabilities based on the collected results and highlights its features. Data collection is presented for the following metrics: i) Coverage rate, ii) Energy consumption rate, iii) Pareto optimal metrics, and iv) Execution time.

5.3.1 Coverage Rates

Network coverage rate indicates the underwater sensor network's coverage strength providing for the monitoring area or targets. This study uses the probabilistic sensing model to determine the network coverage rate. The sensing range of the node is a spherical region with the node as the centre and radius R_s, and $R_s + R_u$ is the communication range shown in Chapter 4 (Figure 4.2).

5.3.1.1 Data Collected for Coverage Rates

This section used the data collected from the experiment to analyse and compare the proposed solution's coverage rate against the other existing algorithms. Table 5.4 compares the sensor nodes' network coverage rates and data transmission for the proposed and existing systems (NSGA-II, SPEA2, and MOEA/D). The first column in the table represents the 30 cycles of data traces for analysis. The first row lists the algorithms. The proposed solution presents a higher percentage of the sensor nodes' network coverage ratio than existing algorithms for different sensor nodes' data transmission – which had 94.46%, 91.25%, and 85.87% coverage, respectively, whereas the proposed MOGA-AMPazy algorithm yielded a much higher coverage rate of 98.75%. The result clearly indicates that the proposed algorithm provides superior network coverage compared to the existing algorithms.

Table 5.4: Comparison of Coverage Rates between Existing and Proposed Algorithms

Algorithm / Data Trace	NSGAII	SPEA 2	MOEA/D	MOGA-AMPazy (Proposed Algorithm)
1	94.12	91.21	85.92	98.42
2	94.02	91.20	85.96	98.96
3	94.05	91.20	85.93	98.10
4	94.06	91.22	85.91	98.90
5	94.73	91.07	85.73	98.73
6	94.82	91.91	85.82	98.82
7	94.91	91.25	85.91	98.91
8	94.11	91.24	85.55	98.88
9	94.13	91.27	85.81	98.87
10	94.14	91.29	85.72	98.72
11	94.15	91.21	85.97	98.76
12	94.18	91.22	85.98	98.59
13	94.18	91.17	85.93	98.66
14	94.20	91.19	85.97	98.78
15	94.23	91.21	85.77	98.79
16	94.32	91.23	85.76	98.82
17	94.38	91.25	85.77	98.83
18	94.42	91.24	85.89	98.45
19	94.46	91.24	85.87	98.50
20	94.47	91.31	85.91	98.78
21	94.55	91.33	85.90	98.88
22	94.63	91.27	85.90	98.69
23	94.68	91.22	85.91	98.78
24	94.73	91.20	85.92	98.89
25	94.78	91.22	85.92	98.79
26	94.82	91.23	85.93	98.85
27	94.84	91.03	85.85	98.87
28	94.91	91.23	85.92	98.89
29	94.94	91.21	85.94	98.86
30	94.96	91.21	91.28	98.87
Mean	94.46	91.25	85.87	98.75

5.3.1.2 Data Collected for Energy Consumption Rates

This study determines the sensor nodes' total energy consumption to evaluate the proposed solution's performance. Total energy consumption is the sum of energy consumed by all sensor nodes used by the networks. As can be observed from Table 5.5, the proposed MOGA-AMPazy solution is more energy-efficient than existing solutions and reduces the total energy utilised in the network. Every sensor determines its power based on the information of its neighbours. This study reveals that the three existing algorithms have similar average energy consumption rates. The average energy consumption of the MOGA-AMPazy algorithm is lower than that of the other three algorithms. Local information enable the proposed solution to comply with the dynamic variation of networks to a certain extent.

Table 5.5: Comparison of Energy Consumption between Existing and Proposed Algorithms

Algorithm / Data Trace	NSGAII	SPEA 2	MOEA/D	MOGA-AMPazy (Proposed Algorithm)
1	41.00	46.00	43.10	40.91
2	41.02	46.02	43.14	40.95
3	41.00	46.00	43.12	40.93
4	41.02	46.02	43.18	40.99
5	41.06	46.01	43.11	40.93
6	41.02	46.06	43.19	40.96
7	41.05	46.02	43.11	40.92
8	41.04	46.09	43.15	40.98
9	41.04	46.05	43.16	40.97
10	41.09	46.03	43.17	40.91
11	41.03	46.08	43.11	40.90
12	41.04	46.08	43.15	40.91
13	41.06	46.05	43.16	40.96
14	41.07	46.03	43.11	40.98
15	41.01	46.06	43.10	40.78
16	41.04	46.03	43.12	40.94
17	41.04	46.01	43.14	40.93
18	41.05	46.04	43.11	40.97
19	41.03	46.03	43.16	40.94
20	41.01	46.09	43.10	40.93
21	41.02	46.03	43.12	40.91
22	41.02	46.04	43.16	40.99
23	41.04	46.05	43.10	40.96
24	41.09	46.06	43.15	40.94
25	41.08	46.09	43.12	40.77
26	41.07	46.04	43.15	40.97
27	41.08	46.07	43.16	40.94
28	41.02	46.04	43.10	40.92
29	41.03	46.01	43.14	40.90
30	41.03	46.05	43.11	40.94
Mean	41.04	46.04	43.13	40.93

5.3.2 Pareto Optimal Metrics

In this context, it is worthwhile to generate solutions that are closer to the Pareto front and varied in the non-dominated front. A convergence consideration is suggested to constrain the initial scaling factor and thereby enhance the algorithm's search performance. Convergence measures the algorithm's capacity to reach the global Pareto front, and diversity resolves the distribution along the Pareto front. The MOGA-AMPazy solution determines three performance metrics: inverted generation distance, hypervolume, and diversity. Both convergence and diversification are analysed.

5.3.2.1 Inverted Generation Distance (IGD)

Inverted generation distance (IGD) is a metric for evaluating the approximations' quality compared to the Pareto front reached by the MOGA-AMPazy algorithm. IGD is interpreted as follows (refer to equation 5.1):

$$IGD(P_A, PF_{true}) = \frac{\sum_{v \in PF_{true}} d(v, P_A)}{|PF_{true}|} \tag{5.1}$$

PF_{true} is a batch of evenly distributed points in the objective space. P_A is the non-dominated solution set gathered by an algorithm, where $d(v, P_A)$ is the minimum Euclidean distance between v and P_A points. Algorithms with smaller IGD values are preferable (Y. Liu, Gong, Sun, & Jin, 2017).

5.3.2.2 Hypervolume

This hypervolume metric considers the volume (in the objective space) enclosed by members of non-dominated solutions sets gathered by multi-objective optimisation. According to (Zambrano-Vega, Nebro, García-Nieto, & Aldana-Montes, 2017), all objectives should be minimised. A hypervolume can be measured as follows (refer to equation 5.2):

$$HV = volume(\bigcup_{i=1}^{|P_A|} v_i) \tag{5.2}$$

The greater the value of the HV, the better the algorithm is.

5.3.2.3 Diversity Metric

The computation of diversity represents a matrix of spread to resolve the uniform distribution of the non-dominated solution. It has been suggested and implemented in (Deb et al., 2002) as (refer to equation 5.3). The smaller the diversity value, the more effective the algorithm.

$$\triangle = \frac{|d_l + d_m + \int_{i=1}^{nPFs} |d_i - d\ |\ |}{d_l + d_m + (n-1)d} \tag{5.3}$$

where d_l, d_m are Euclidean distances between absolute solutions in the true Pareto front and obtained Pareto front. d_i indicates the Euclidean distance between each point in true Pareto front and obtained Pareto front. n_{PFs} is the total number of produced Pareto-optimal solutions, and d is the average distance of all solutions.

5.3.2.4 Execution Time

In addition to the indicated metrics, Pareto-optimal solutions should be achieved within a feasible time by an efficient algorithm. Thus, the execution time with the IGD, HV, and diversity metrics is also determined in this study to demonstrate the proposed algorithm's efficiency and effectiveness.

5.3.2.5 Data Collected for Pareto Optimal Metrics and Execution Time

The parameter setting presented in Table 5.3 is considered by the MOGA-AMPazy solution to examine the proposed crossover and mutation operators. Tables 5.6 and 5.7 contain all performance metrics collected using parameter settings and average values. The first column in the table describes the 30 cycles of data traces for analysis. The first row displays the Pareto-optimal metrics comprising diversity, hypervolume, inverted generation distance, and algorithm execution time. Metric averages are 0.5476, 0.7490, and 0.0013, whereas the algorithm execution time is 0.4101 hours. Furthermore, the tabulated results show that the diversity metric's best value occurred at the tenth cycle. According to the table, cycles 1 and 30 presented the lowest and highest execution time values, respectively. It may arise because of low and high values (number of generations, number of individuals, and crossover rate).

Based on the data traces, the proposed solution is faster than the existing algorithms. The NSGA-II algorithm performs marginally lower than the MOGA-AMPazy algorithm in terms of execution time; it requires 0.4235 hours on average. However, the table shows that NSGA-II has the best diversity metric ($\triangle = 0.3785$) at cycle 26. It possesses the best extent of spread among the obtained solutions. Furthermore, SPEA2 has the highest (i.e. worst) diversity metric: $\triangle = 0.6465$. The substantial variation of measured diversity values between NSGA-II and MOGA-AMPazy ($\triangle(MOGA - AMPazy) < \triangle(NSGAII)$), as well as between the MOEA/D and SPEA2 ($\triangle(MOEA/D) < \triangle(SPEA2)$), implies that the selection method in the proposed solution is more efficient in terms of diversity metrics. Furthermore, comparing the four algorithms' respective diversity metrics indicates that adaptive multi-parent crossover applied in MOGA-AMPazy provides a distribution closer to the uniform distribution. In this study, the other two metrics (i.e. hypervolume and IGD) were gathered for the four algorithms. Based on the data traces, MOGA-AMPazy has the lowest IGD metric value and the highest HV metric.

5.4 Conclusion

This chapter evaluates the proposed solution's performance by implementing a suitable simulation that supports the UWSN environment. Furthermore, this chapter reported six evaluation components: coverage rates; energy consumption; execution time; and the Pareto-optimal metrics of IGD, hypervolume and diversity. This study also compiles data for packet delivery ratio and throughput.

This chapter presented the data collected to analyse and compare the proposed solution's coverage rate against that of other algorithms (see Table 5.4). Table 5.5 presented the data collected to compare the energy consumption of sensor nodes in UWSNs using the proposed algorithm against that of three existing algorithms. Furthermore, Tables 5.6 and 5.7 presented the data collected to analyse the Pareto-optimal metrics using IGD, hypervolume, diversity metrics, and execution time concerning the MOGA-AMPazy solution and existing algorithms. Tables 5.8 and 5.9 present the data collection related to packet delivery ratio and throughput for data transmission between sensor nodes and sinks.

In this research, benchmarking was performed by comparing and evaluating the proposed MOGA-AMPazy solution with three different existing algorithms: NSGA-II, SPEA2, and MOEA/D. Data was collected by sampling the evaluation parameters in up to 30 data traces for the proposed MOGA-AMPazy algorithm, which the overall assessment revealed to provide better coverage rates, lesser energy consumption, and lower execution time. Additionally, MOGA-AMPazy offered the best diversity metric and the best spread. The results obtained have established that the proposed solution outperformed the existing algorithms. Ultimately, the evaluation test established that the proposed solutions provided higher coverage rates with lower energy consumption and shorter execution time than the other three existing algorithms. The next chapter presents the findings of the data analyses.

Table 5.6: Data Collected for Diversity, Hypervolume, IGD and Execution Time for NSGAII and MOEA/D Algorithms

Data Traces	NSGAII				MOEA/D			
	Diversity	Hypervolume	Inverted Generation Distance (IGD)	Execution time (h)	Diversity	Hypervolume	Inverted Generation Distance (IGD)	Execution time (h)
1	0.5622	0.6557	0.0019	0.3145	0.6722	0.6552	0.0020	0.1450
2	0.4388	0.6934	0.0017	0.2226	0.4552	0.6293	0.0020	0.2260
3	0.6913	0.6375	0.0020	0.2250	0.7653	0.8375	0.0012	0.2425
4	0.6559	0.3594	0.0021	0.2245	0.7433	0.4793	0.0017	0.2450
5	0.5320	0.5733	0.0014	0.3411	0.5555	0.5734	0.0015	0.4110
6	0.6174	0.6483	0.0017	0.2251	0.6544	0.6833	0.0018	0.4251
7	0.6653	0.6522	0.0018	0.2390	0.6532	0.6594	0.0011	0.2390
8	0.5123	0.7524	0.0016	0.4250	0.5166	0.7543	0.0012	0.4425
9	0.4733	0.5682	0.0014	0.4133	0.4783	0.7232	0.0017	0.4130
10	0.3925	0.4372	0.0013	0.4460	0.5922	0.4672	0.0011	0.4460
11	0.5537	0.6776	0.0017	0.3248	0.5566	0.4376	0.0014	0.5348
12	0.5655	0.3863	0.0013	0.4140	0.5788	0.5863	0.0016	0.4140
13	0.6772	0.3772	0.0010	0.5420	0.6672	0.6772	0.0017	0.5420
14	0.7523	0.5221	0.0012	0.2368	0.6533	0.4721	0.0014	0.3680
15	0.6518	0.3784	0.0011	0.4450	0.6718	0.6784	0.0019	0.4450
16	0.6654	0.6442	0.0013	0.2411	0.6658	0.5642	0.0021	0.5411
17	0.5853	0.7642	0.0016	0.7300	0.5858	0.5864	0.0016	0.7300
18	0.4443	0.7713	0.0013	0.3540	0.6565	0.6571	0.0018	0.4354
19	0.4432	0.5677	0.0014	0.4212	0.4666	0.6479	0.0015	0.4120
20	0.6553	0.6376	0.0018	0.5553	0.6766	0.6333	0.0014	0.5530
21	0.6342	0.4394	0.0012	0.5960	0.6781	0.7392	0.0015	0.5496
22	0.5773	0.7661	0.0017	0.3450	0.5765	0.7613	0.0014	0.4345
23	0.6733	0.6577	0.0015	0.3475	0.7100	0.7733	0.0016	0.3750
24	0.6332	0.6557	0.0014	0.4487	0.5633	0.4353	0.0018	0.4870
25	0.4633	0.5745	0.0020	0.6712	0.5333	0.4774	0.0021	0.6712
26	0.3785	0.6482	0.0017	0.5240	0.4523	0.3682	0.0019	0.5524
27	0.4376	0.6665	0.0014	0.4140	0.4377	0.3276	0.0016	0.4414
28	0.5733	0.5729	0.0014	0.5583	0.6753	0.6529	0.0017	0.5830
29	0.7232	0.6772	0.0013	0.4492	0.7230	0.6572	0.0020	0.4920
30	0.5437	0.5773	0.0016	1.0112	0.5444	0.6232	0.0019	1.1040
Mean	**0.5725**	**0.5980**	**0.0015**	**0.4235**	**0.6053**	**0.6073**	**0.0016**	**0.4634**

Table 5.7: Data Collected for Diversity, Hypervolume, IGD and Execution Time for MOEA/D and MOGA-AMPazy Algorithms

Data Traces	SPEA2				MOGA-AMPazy (Proposed Algorithm)			
	Diversity	Hypervolume	Inverted Generation Distance (IGD)	Execution time (h)	Diversity	Hypervolume	Inverted Generation Distance (IGD)	Execution time (h)
1	0.7920	0.6322	0.0021	0.1445	0.4920	0.7000	0.0019	0.1450
2	0.4300	0.7932	0.0020	0.2326	0.4302	0.6930	0.0017	0.2262
3	0.8910	0.8321	0.0013	0.1225	0.5911	0.8750	0.0013	0.2252
4	0.7990	0.7940	0.0017	0.1245	0.6991	0.7940	0.0013	0.2450
5	0.6110	0.7655	0.0015	0.4211	0.5322	0.7300	0.0014	0.4110
6	0.7881	0.6122	0.0016	0.3510	0.6171	0.6830	0.0017	0.2510
7	0.6542	0.6233	0.0011	0.2390	0.5655	0.6520	0.0011	0.2391
8	0.4312	0.7565	0.0012	0.4250	0.5122	0.7540	0.0012	0.4256
9	0.6873	0.6824	0.0014	0.4130	0.4732	0.7820	0.0012	0.4130
10	0.5392	0.6442	0.0011	0.4460	0.3920	0.7200	0.0013	0.4464
11	0.5670	0.5996	0.0014	0.4811	0.5530	0.7600	0.0013	0.4388
12	0.6111	0.5633	0.0013	0.4124	0.5664	0.8630	0.0013	0.4140
13	0.6870	0.5442	0.0017	0.5242	0.6670	0.7720	0.0007	0.4424
14	0.7672	0.6218	0.0014	0.5768	0.5520	0.7210	0.0012	0.3680
15	0.7658	0.6847	0.0018	0.4445	0.6183	0.7840	0.0011	0.4454
16	0.6683	0.5428	0.0013	0.4211	0.6652	0.6420	0.0013	0.4116
17	0.6705	0.5864	0.0016	0.5373	0.5850	0.8640	0.0014	0.3273
18	0.5426	0.6671	0.0012	0.3354	0.4566	0.7100	0.0012	0.3544
19	0.6553	0.5479	0.0014	0.4354	0.4630	0.7900	0.0012	0.4120
20	0.6865	0.6226	0.0014	0.5353	0.6550	0.6760	0.0013	0.4211
21	0.6971	0.6839	0.0012	0.3596	0.6015	0.8390	0.0012	0.5960
22	0.5757	0.7861	0.0012	0.4345	0.5777	0.8610	0.0012	0.3452
23	0.5678	0.7730	0.0016	0.4375	0.6654	0.7700	0.0015	0.3752
24	0.6330	0.5653	0.0018	0.4873	0.6330	0.6530	0.0013	0.4875
25	0.7600	0.7474	0.0021	0.5712	0.4600	0.7740	0.0023	0.5126
26	0.4490	0.6823	0.0017	0.5324	0.3790	0.6820	0.0017	0.5245
27	0.4380	0.7633	0.0014	0.4314	0.4385	0.7600	0.0014	0.4145
28	0.7123	0.4293	0.0017	0.4583	0.5700	0.7290	0.0013	0.5830
29	0.7230	0.5773	0.0019	0.4392	0.4235	0.7720	0.0010	0.4923
30	0.5933	0.6650	0.0017	1.0030	0.5940	0.6650	0.0014	0.9102
Mean	**0.6465**	**0.6596**	**0.0015**	**0.4259**	**0.5476**	**0.7490**	**0.0013**	**0.4101**

CHAPTER 6: RESULTS AND DISCUSSION ON THE PERFORMANCE OF MOGA-AMPAZY METHOD

This chapter presents the performance evaluation of the MOGA-AMPazy solution. This chapter describes the data analysis performed for this research (see Chapter 5) and the outcomes of comparative analysis against existing algorithms. The proposed MOGA-AMPazy solution has also been critically evaluated using various tests and techniques. This chapter also highlights the MOGA-AMPazy solution's effectiveness in terms of coverage rates, energy consumption, Pareto-optimal metrics, and execution time by comparing it with existing algorithms.

This chapter is organised as follows: Section 6.1 presents details of the performance evaluation of the MOGA-AMPazy solution; Section 6.2 presents details from the analysis of the MOGA-AMPazy coverage rate against that of existing algorithms; Section 6.3 compares the MOGA-AMPazy solution's energy consumption against existing algorithms; Section 6.4 provides details of performance evaluation concerning unconstrained test problems; Section 6.5 compares MOGA-AMPazy execution time with existing algorithms; and Section 6.6 summarises the chapter.

6.1 MOGA-AMPazy Evaluation Parameters

This section describes several evaluation metrics applied in analysing the performance of the proposed MOGA-AMPazy algorithm. One of the purposes of evaluating the results is to demonstrate that the deployment problem can be adequately addressed when formulated as a multi-objective problem. This section presents the Pareto front representation between two objectives: the energy consumption and coverage functions. Multi-objective optimisation aids decision-makers by enabling them to designate any coverage threshold needed for the application while also significantly reducing energy consumption and extending sensor lifetime compared to what they could achieve with a full-coverage network. Single-objective optimisation procedures cannot make such decisions. Specific performance metrics are needed to evaluate and justify the performance of the multi-objective algorithms. The performance of a multi-objective optimisation method is evaluated in terms of the Pareto front's convergence to the ideal front. The variability of the collected front allows for the most accurate estimation of the optimum Pareto front.

The performance of the MOGA-AMPazy solution has been measured using the following parameters:

1. Coverage rates
2. Energy consumption
3. Pareto Optimal Metrics
4. Execution Time

6.2 MOGA-AMPazy Performance Analysis on Coverage Rates

In this study, two case studies were conducted to evaluate the performance of the proposed MOGA-AMPazy algorithm and that of three other existing algorithms. For Case Study 1, the experiment observes node density on a sparse network. As shown in Test A, Table 5.1 (Chapter 5), the nodes deployed were 15 nodes. The coverage ratios were obtained using the maximum distance travelled by all sensors for all tests. In comparison to NSGA-II, SPEA2, and MOEA/D, it is observed that MOGA-AMPazy achieves the greatest coverage rates and travelled distance. It is seen that (refer to Figure 6.1) the relative improvement in coverage rate is small, but the increase in travelled distance is significant. Through the experiments conducted, this study found that with the implementation of the NSGA-II algorithm, the sensor nodes' movement duration increased over time to fully cover the region of interest. It is due to NSGA-II requiring several iterations to discover the optimum solution, particularly when starting with a random initial solution. The results are also supported by findings from (Khoufi, Hadded, Minet, & Laouiti, 2015).

In contrast, the objective function of the MOGA-AMPazy algorithm considers energy consumed during movement and sensing, as well as coverage rate. However, compared to the SPEA2 algorithm, MOGA-AMPazy represents an improvement over all test coverage rates. The MOGA-AMPazy method's improvement is based on limiting the average distance travelled by all network nodes. In contrast, the SPEA2 algorithm is based on the penalty function concerning the maximum distance moved. Moreover, the MOGA-AMPazy algorithm limits the sensing radius of all nodes and saves on energy-sensing for all tests. In contrast, the SPEA2 easily falls into local optima and has a low convergence rate in the iterative process. MOGA-AMPazy guarantees the population's diversity; simultaneously, it avoids becoming 'stuck' at a local optimum and accelerates convergence.

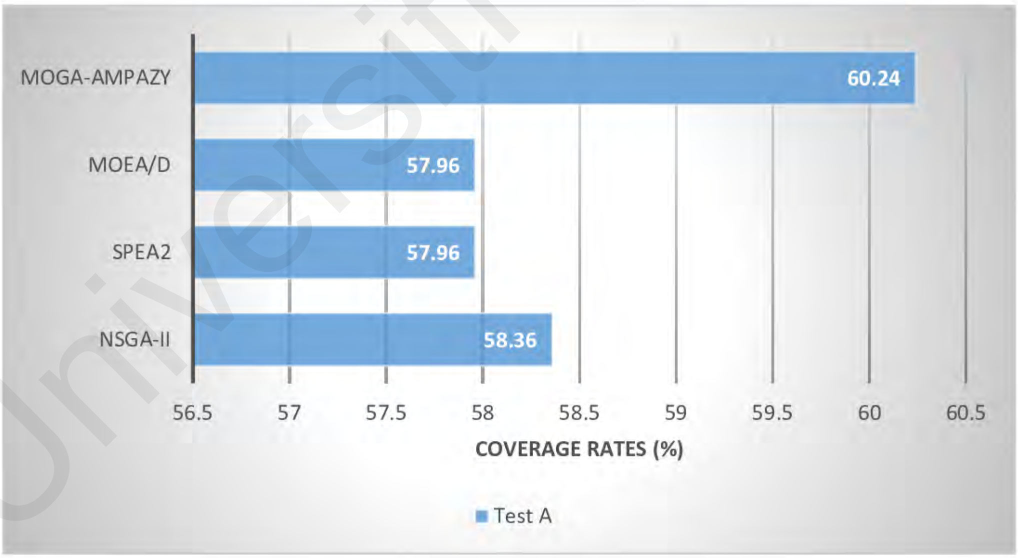

Figure 6.1: Result of Case Study 1 (The Effect of Coverage Area Ratio on Sparse Network)

For Case Study 2 (refer to Table 5.1 for Test B), the experiment is conducted to perceive the effect of node density on the dense network. This study observes the impact of node density on the proposed algorithm's performance compared to the NSGA-II algorithm.

This experiment involved varying the number of randomly deployed nodes between 100 and 200 (using a step size of 10 nodes) within a 1000 m x 1000 m x 1000 m^3 field. Figure 6.2 shows the coverage area ratio for the three algorithms versus the number of deployed nodes. Due to its ability to balance the trade-off between coverage and energy consumption, the proposed MOGA-AMPazy algorithm gives higher and more stable coverage rates. It also consumes less energy than the NSGA-II and MOPSO algorithms for different node densities. The NSGA-II algorithm performs well in a densely distributed environment, although its efficiency decreases as node density increases.

However, the MOGA-AMPazy algorithm can adjust the sensing radius of all sensor nodes to maximise coverage and preserve energy at different node densities. The analysis has also been conducted using 30 cycles to observe algorithm performance. According to Figure 6.3, the proposed solution (i.e. the MOGA-AMPazy algorithm) demonstrates its superiority over the other existing algorithms. This study again proves that MOGA-AMPazy achieved a higher coverage percentage compared to other algorithms.

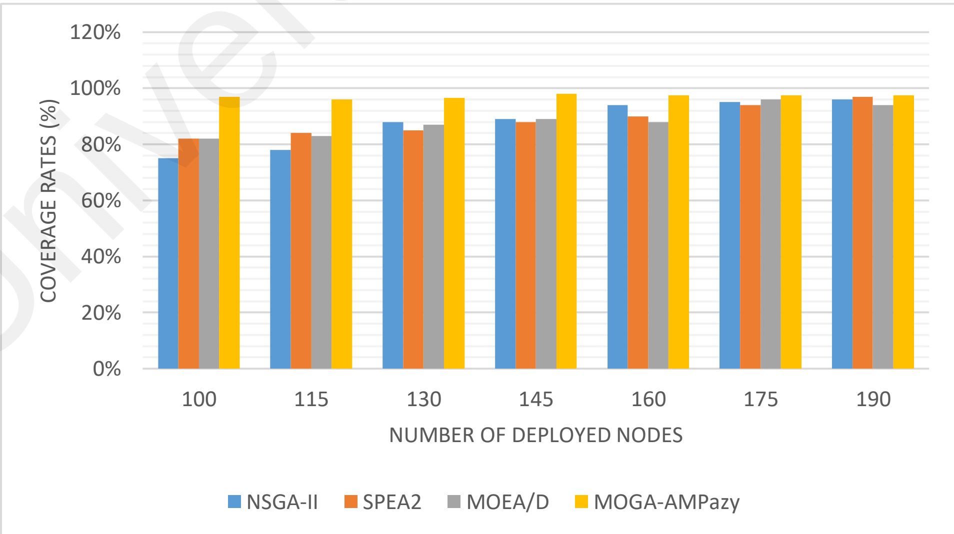

Figure 6.2: Result of Case Study 2 (The Effect of Coverage Area Ratio on Dense Network)

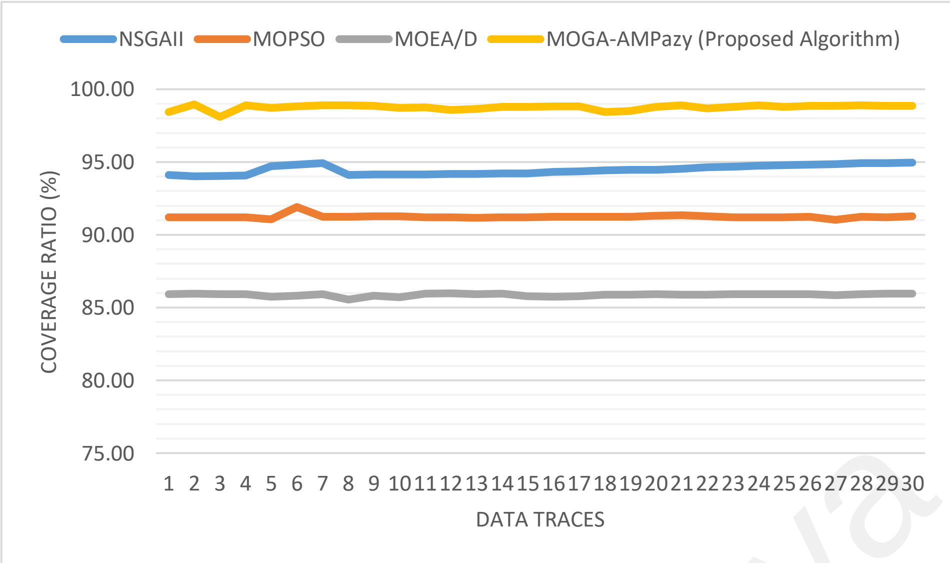

Figure 6.3: Comparison of Coverage Rates between Existing and Proposed Algorithms

6.3 MOGA-AMPazy Performance Analysis on Energy Consumption

In Figure 6.4, the energy consumption of sensor nodes of different UWSN algorithms is compared for a different number of nodes. Once again, the MOGA-AMPazy algorithm conserved more energy for the UWSN for every experiment round. The network setup for this experiment is as described in Table 5.2.

NSGA-II overcomes the computational difficulty associated with sorting non-dominated solutions, the absence of elitism, and the need to provide a sharing value. In this respect, Figure 6.4 demonstrates the efficacy of the proposed MOGA-AMPazy algorithm over NSGA-II. The superiority of the proposed solution is evident in test instances with a different number of sensor nodes. The proposed solution has less computational complexity in optimisation due to little communication overhead. Thus, it consumes less energy. Figure 6.5 presents the comparison of energy consumption for the four algorithms. The y-axis shows the energy consumption ratio, and the x-axis represents the four algorithms for thirty different data traces. The results indicate that the energy consumption rate of MOGA-AMPazy is 40.93% lower than that of the other three algorithms. The results are computed at data traces 1 to 30 for all the processed data.

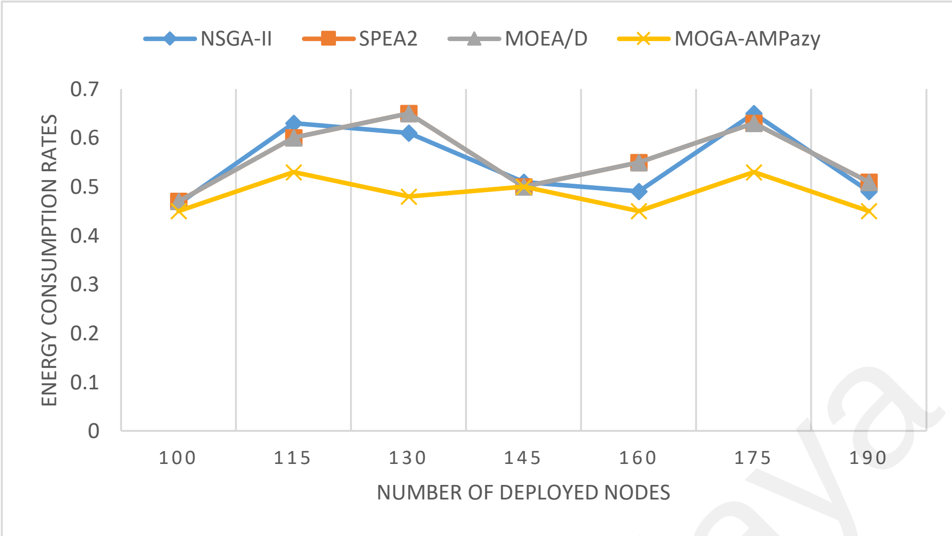

Figure 6.4: The effect of Energy Consumption Ratio on Dense Network

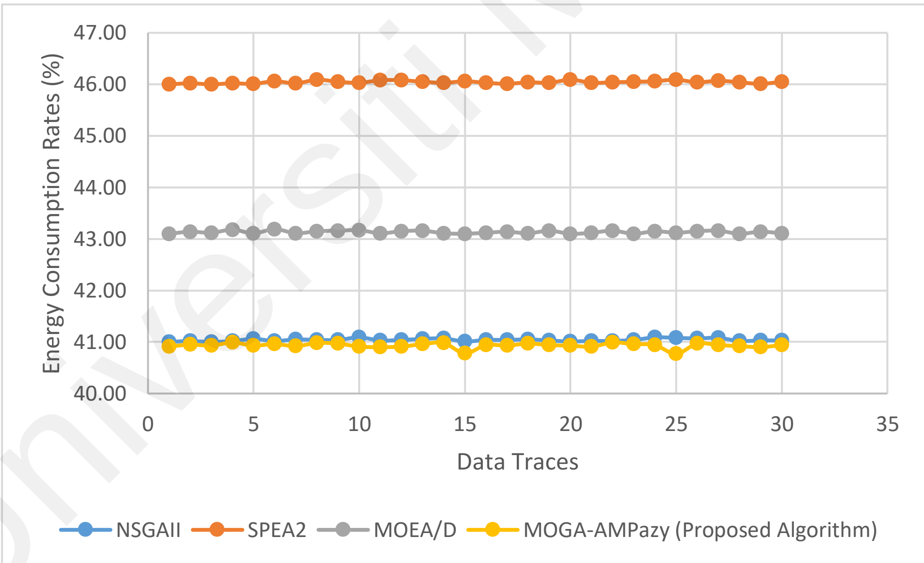

Figure 6.5: Comparison of Energy Consumption Rates between Existing and Proposed Algorithms

6.4 MOGA-AMPazy Performance Analysis on Pareto Optimal Metrics

This study utilises five widely used two-objective ZDT test instances (ZDT-1, ZDT-2, ZDT-3, ZDT- 4, and ZDT-6) as benchmark tests (Hansen & Jaszkiewicz, 1998) to exploit specific problem characteristics to benchmark the proposed solution and three other existing systems. The test instances consist of seven iterations. For each iteration, the experiment generates 50 approximation fronts that compete for survival in the candidate pool. Performance assessment uses IGD, hypervolume, and diversity. Each process involves binary tournaments to identify losers and thereby prevent further consideration. Pareto fronts are preserved in a loser bracket for the next iteration. When all the iterations are completed, the winner (in terms of the benchmark function) is identified.

1. **ZDT-1 Function**

   The ZDT-1 function has a convex Pareto-optimal front. The objective functions are:

$$f_1(x) = x_1 \tag{6.1}$$

$$f_2(x) = g(x)|1 - \sqrt{x_1/g(x)} \tag{6.2}$$

   Where $g(x)$ is defined as:

$$g(x) = 1 + \frac{9(\sum_{i=2}^{n} x_i)}{(n-1)} \tag{6.3}$$

   In this ZDT-1 function, thirty design variables x_i were chosen ($n = 30$). Each design variable ranged in value from 0 to 1. The Pareto-optimal front appears when $g = 1.0$.

2. **ZDT-2 Function**

   The ZDT-2 function has a convex Pareto-optimal front. The objective functions are:

$$f_1(x) = x_1 \tag{6.4}$$

$$f_2(x) = g(x)|1 - (\frac{x_1}{g(x)})^2| \tag{6.5}$$

Where $g(x)$ is defined as:

$$g(x) = 1 + \frac{9(\sum_{i=2}^{n} x_i)}{(n-1)} \tag{6.6}$$

In this ZDT-2 function, thirty design variables x_i were chosen ($n = 30$). Each design variable ranged in value from 0 to 1. The Pareto-optimal front appears when $g = 1.0$.

3. **ZDT-3 Function**

The ZDT-3 function adds a discreteness feature to the front. Its Pareto-optimal front consists of several non-contiguous convex parts. The introduction of a sine function in this objective function generates discontinuities in the Pareto-optimal front but not in the parameter space. The objective functions are:

$$f_1(x) = x_1 \tag{6.7}$$

$$f_2(x) = g(x)|1 - \sqrt{x_1/g(x)} - \frac{x_1}{g(x)} sin(10\pi x_1)| \tag{6.8}$$

Where $g(x)$ is defined as:

$$g(x) = 1 + \frac{9(\sum_{i=2}^{n} x_i)}{(n-1)} \tag{6.9}$$

In this ZDT-3 function, thirty design variables x_i were chosen ($n = 30$). Each design variable ranged in value from 0 to 1. The Pareto-optimal front appears when $g = 1.0$.

4. **ZDT-4 Function**

The ZDT-4 function has 21 local Pareto-optimal fronts and therefore is highly multi-modal. The objective functions are:

$$f_1(x) = x_1 \tag{6.10}$$

$$f_2(x) = g(x)|1 - \sqrt{x_1/g(x)} \tag{6.11}$$

Where $g(x)$ is defined as:

$$g(x) = 1 + 10(n-1) + \sum_{i=2}^{n}[x_k^2 - 10cos(4\pi x_i)] \tag{6.12}$$

In this ZDT-4 function, ten design variables x_i were chosen ($n = 10$). The design variable ranges are from -5 to 5 for the last nine design variables and from 0 to 1 for x_1. The global Pareto-optimal front appears when $g = 1.0$.

5. **ZDT-6 Function**

The ZDT-6 function has a non-uniform search space: the Pareto-optimal solutions are non-uniformly distributed along the global Pareto front; additionally, the density of the solutions is lowest near the Pareto-optimal front and highest away from the front. The objective functions are defined as:

$$f_1(x) = 1 - e^{-4x_1} sin^6(6\pi x_1) \tag{6.13}$$

$$f_2(x) = g(x)[1 - (\frac{f_1(x)}{g(x)})^2] \tag{6.14}$$

Where $g(x)$ is defined as:

$$g(x) = 1 + 9[\frac{\sum_{i=2}^{n} x_i}{(n-1)}]^{\frac{1}{4}} \tag{6.15}$$

In this ZDT-6 function, ten design variables x_i were chosen ($n = 10$). The design variable ranges are from 0 to 1. The global Pareto-optimal front appears when $g = 1.0$.

The statistical results shown in Table 6.1 suggest that the MOGA-AMPazy algorithm effectively outperforms most of the unconstrained test functions compared to the NSGA-II, SPEA2, and MOEA/D. The proposed MOGA-AMPazy solution's effectiveness can be seen in Table 6.1, showing greater robustness and accuracy in mean and standard deviation for IGD, hypervolume, diversity, and execution time. However, the results for MOGA-AMPazy are not vastly superior to those of NSGA-II and MOEA/D; in some cases, these algorithms perform better than MOGA-AMPazy. The Pareto front obtained by the MOGA-AMPazy solution shows almost complete coverage concerning the actual Pareto front. The results for the best obtained Pareto-optimal values of the proposed MOGA-AMPazy algorithm are presented in Figure 6.6. This study observed that MOGA-AMPazy performs well with higher accuracy and better robustness. This research requires a decision-maker to choose the optimal compromise option that controls the deployed nodes' placement based on the Pareto achieved in the proposed solution. Overall, decision-makers are applied in order to find the best compromise solution using problem-specific information. In the case of sensor node deployment with coverage and energy consumption as two objectives, the decision has been made to choose the lowest energy solution below a certain coverage level. This study implements a coverage threshold of 95%.

6.5 MOGA-AMPazy Performance Analysis on Execution Time

This section presents the performance analysis to validate the proposed MOGA-AMPazy solution's execution time with four existing algorithms—the entire execution time of algorithms calculated along with the unconstrained test problem. For a two-objective

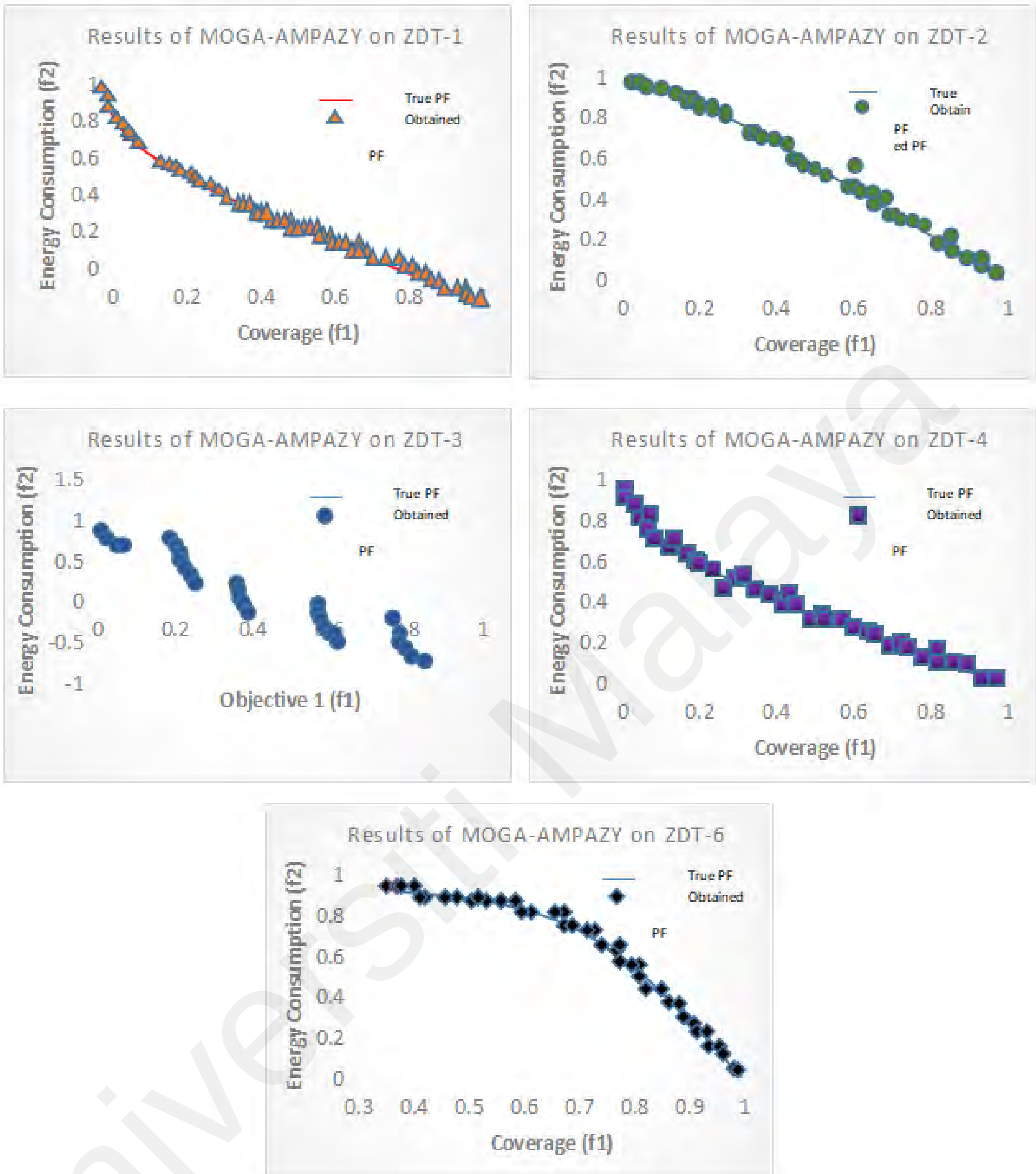

Figure 6.6: Obtained Pareto Optimal Solutions by MOGA-AMpazy for ZDT-1, ZDT-2, ZDT-3, ZDT-4, and ZDT-6

unconstrained test problem, ZDT 1, assuming each approximation front to have 100 solutions and the true Pareto front to have 15 solutions, the average execution time for the proposed performance metrics is 2.169s. In contrast, the maximum time needed is 3.811s and a minimum of 1.247s. Figure 6.7 shows the computational time of the performance metrics for each test problem. This study observed that different test problems with the same number of objectives call for nearly the same computational time (e.g. ZDTs 1–6).

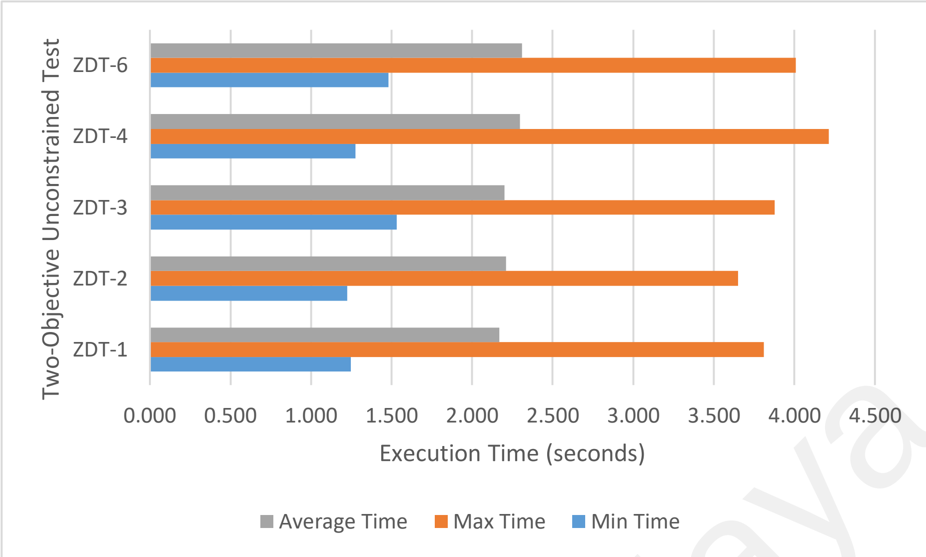

Figure 6.7: Computational time of MOGA-AMPazy for Each Test Problem

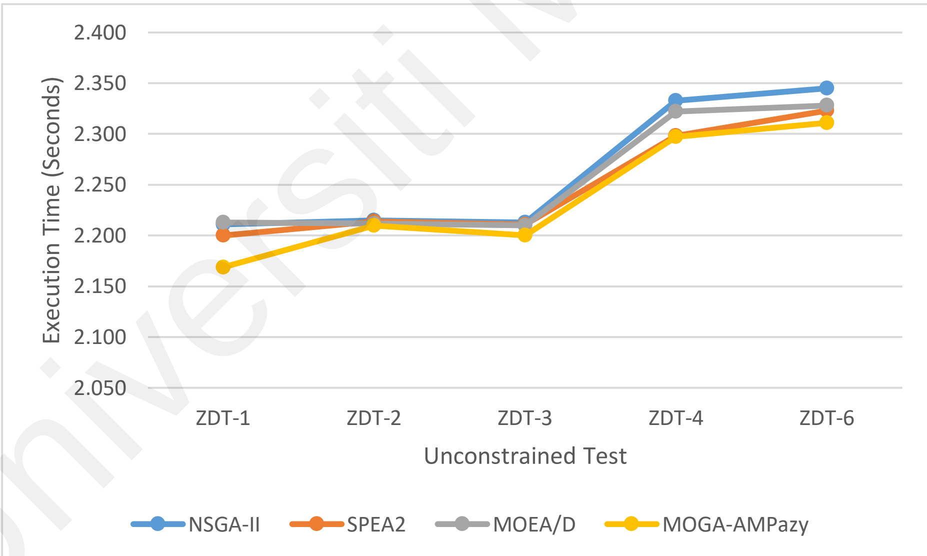

Figure 6.8: Execution Time for Proposed MOGA-AMPazy Compared with Three Existing Algorithms

Figure 6.8 illustrates the impact of execution time on the algorithms. Based on Figure 6.8, it is evident that the algorithm proposed in this study has achieved a shorter execution time compared to the other existing algorithms. Factors contributing to these results are due to the capability of the proposed solution to achieve Pareto-optimal within a feasible time. Adopting the genetic algorithm based on adaptive multi-parent crossover and fuzzy dominance solutions offers optimal value in the number of generations, individuals, and the crossover rate.

6.6 MOGA-AMPazy Performance Analysis on Node Density

Because of the geographical complexity and variability of the target area, in order to obtain precise monitoring data, several sensor nodes must be deployed to obtain a correlation between the coverage rate and the number of sensor nodes deployed. MOGA-AMPazy relocates the mobile sensor nodes based on controlling each node's mobility, the sensing range and energy consumption to maximise the coverage, conserve the dissipated energy in mobility, sensing and redundant coverage. In this study, there are two tests conducted with different numbers of nodes, as shown in Table 5.1. The study is also conducted in two different scenarios to examine the effect of node density on sparse and dense networks.

6.6.1 Effect of Node Density on Sparse Network

In Case Study 1, to examine the effect of node density on the sparse network, coverage ratios were calculated for all tests, using the maximum distance travelled by all sensors and the maximum dissipated mobility energy. As listed in Table 6.3, the average sensing radius and the power consumption in sensing are also determined. Compared to NSGA-II, MOGA-AMPazy is observed to increase the coverage rate and the distance covered by the three measures. It was shown that the coverage rate increase is limited, although there is a substantial improvement in the distance travelled. This is because in its objective function, the NSGA-II algorithm considers only the coverage rate, whereas the MOGA-AMPazy algorithm includes energy consumption in both mobility and sensing in its objective function, in addition to the coverage rate.

On the other hand, it can be shown that MOGA-AMPazy increases the coverage rates of all tests relative to the NSGA-II, SPEA2, and MOEA/2 algorithms. This is because the MOGA-AMPazy approach is focused on restricting the average moving distances of all network nodes. Furthermore, MOGA-AMPazy limits the sensing radius of all nodes and conserves energy for all tests. As shown in Table 6.3, the execution time of the proposed method is shorter than that of the existing methods. The SPEA2 and MOEAD/2 in the iterative process easily fall into the local optimum and have low convergence rates. Although MOGA-AMPazy guarantees population diversity, it also avoids becoming 'stuck' at a local optimum and accelerates its convergence.

6.6.2 Effect of Node Density on Dense Network

In order to perceive the influence of node density on the dense network, Case Study 2 investigates the effects of node density on the efficiency of the proposed MOGA-AMPazy algorithm compared to the NSGA-II algorithm. This is accomplished by altering the number of randomly deployed nodes inside the 150 m x 150 m x 150 m area from 100 to 200 by phase 10. The proposed algorithm offers a higher and more stable coverage rate as well as lower energy consumption compared to the NSGA-II, SPEA2, and MOEA/D algorithms for different node densities because of its ability to balance the trade-off between coverage and dissipated energy. In a densely deployed setting, the NSGA-II algorithm works well, but decreasing node density causes its performance to decline. In contrast, MOGA-AMPazy is capable of changing the sensing radius of all sensor nodes in the network to maintain coverage and conserve the network's energy at various node-density levels.

6.7 MOGA-AMPazy Performance Analysis on Network Parameters

This section focuses on another significant performance analysis of the proposed solution. This research evaluates the performance using network parameters in packet delivery ratio and throughput analysis.

6.7.1 Packet Delivery Ratio (PDR)

The packet delivery ratio is obtained by dividing the number of packets received by the number of intended packets received.

Figure 6.9 shows how the proposed system compares to existing systems in terms of the percentage of packets delivered and the number of sensor nodes. For several sensor nodes, the proposed method has a better rate of packet delivery than existing methods. The number of sensor nodes can be anywhere between 0 and 400. Compared to other methods, the proposed solution has a higher rate of packet delivery as the number of sensor nodes goes up. For instance, when there are 400 nodes, the proposed system's packet delivery ratio is 4.11 percent higher than with SPEA2, 8.82 percent higher than with NSGAII, and 11.76 percent higher than with MOEA/D. Figure 6.9 shows how the number of sensor nodes and the percentage of packets that are delivered change over time. Figure 6.9 shows that the proposed solution is worth more than the systems that are already in place.

6.7.2 Throughput (TP)

Throughput is the amount of data that can be sent over a network within a certain bandwidth per unit of time, which is measured in seconds. Randomly placing the sensor nodes in a strong coverage area increases the throughput automatically based on the size of the data packets in bytes. In this case, megabits per second (Mb/s) is used to show how fast data is sent.

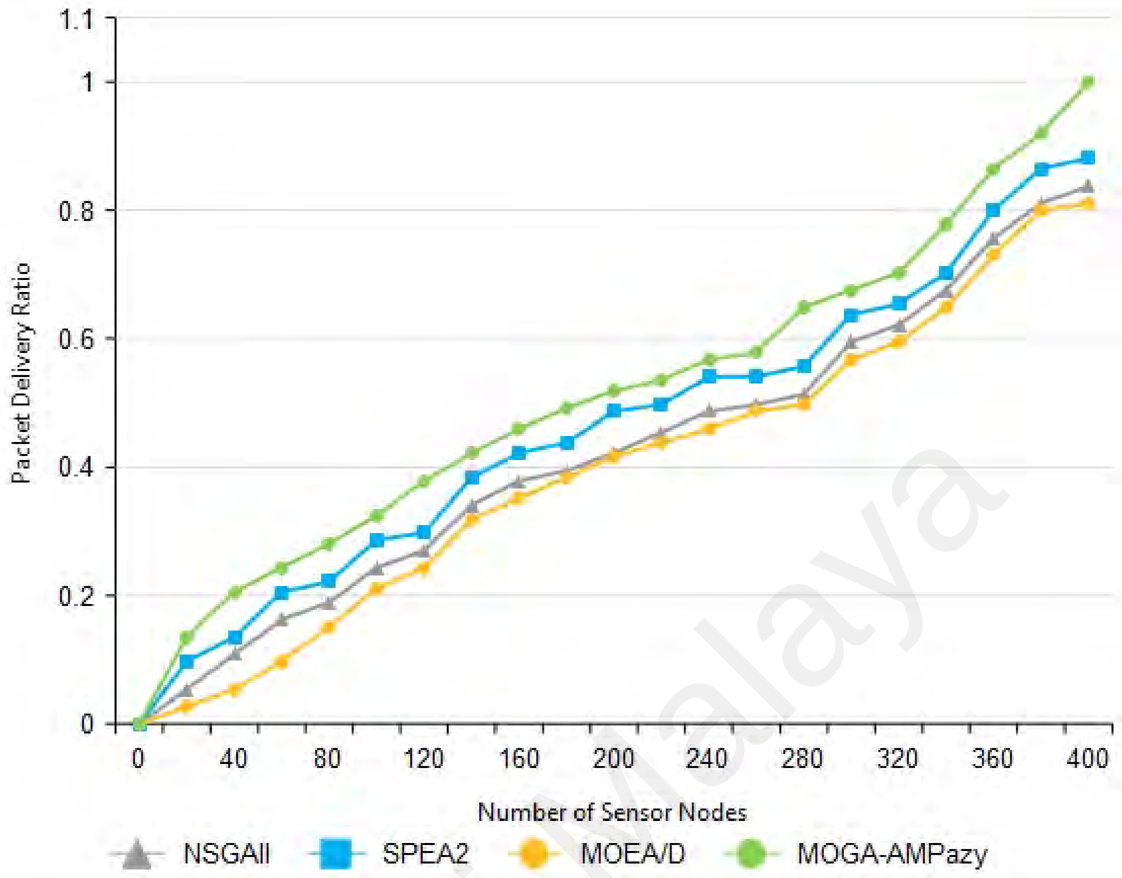

Figure 6.9: Performance of Packet Delivery Ratio

Figure 6.10 compares the proposed system and existing methods in terms of sensor node throughput and number of data transmissions. In comparison to existing systems, the proposed approach demonstrates a greater percentage of throughput value for varied numbers of data transmission of sensor nodes. The number of sensor node data transmissions is adjusted in 20-step increments from 0 to 400. As the number of data transmissions from the sensor nodes is 400, the proposed system's throughput increases by 3.7 percent, 2.4 percent, and 4.9 percent, respectively, compared to NSGAII, SPEA2, and MOEA/D.

MOGA-AMPazy demonstrated more consistent performance for two objective functions across 30 time runs among the algorithms in the studies. In Figure 6.10, it is obvious that the proposed solution has a greater value compared to the current system.

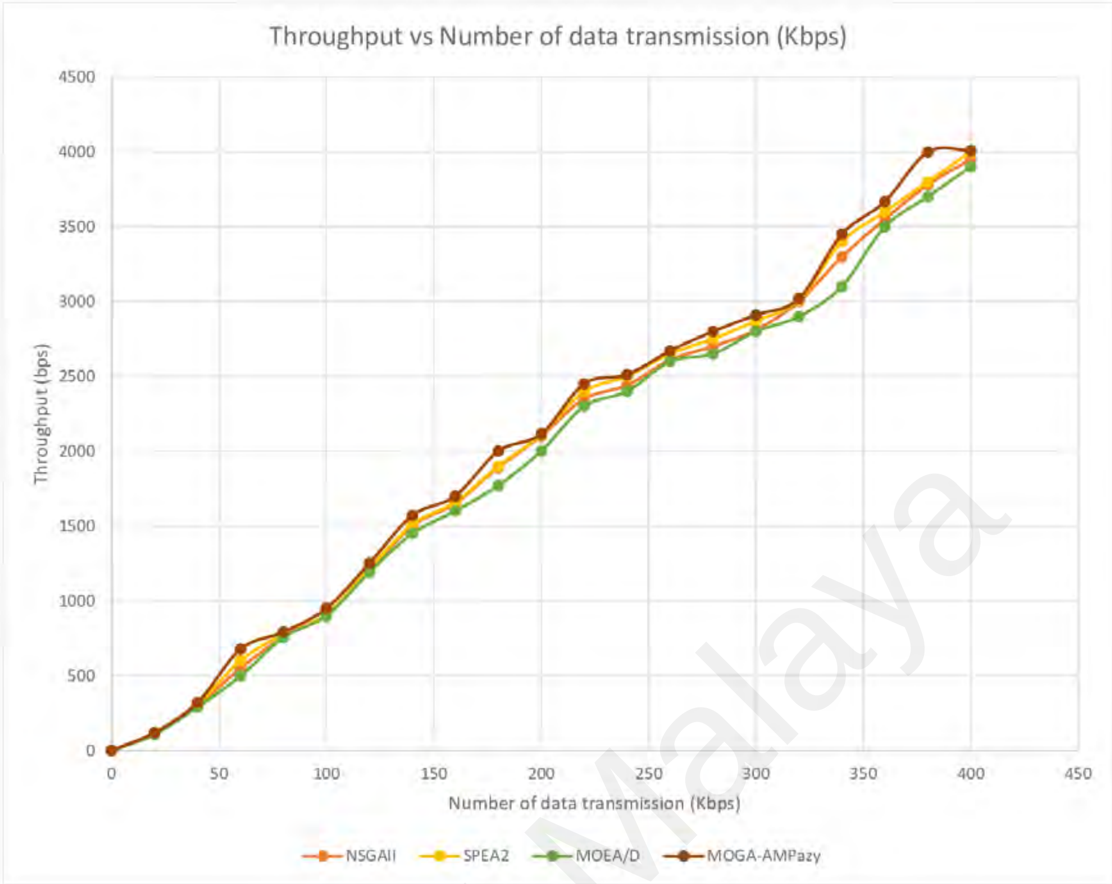

Figure 6.10: Performance of Network Throughput

6.8 Conclusion

This chapter presented the results of experiments in a UWSN environment to evaluate the performance of MOGA-AMPazy, having measured its coverage rates, energy consumption ratio, Pareto-optimal metrics and execution time. This research adopted three Pareto-optimal metrics (i.e. IGD, hypervolume and diversity) to determine the proposed algorithm's effectiveness. Furthermore, this research utilises five commonly used two-objective ZDT test instances as a benchmark test, namely ZDT 1, ZDT 2, ZDT 3, ZDT 4 and ZDT 6, to employ specific problem characteristics to impose the underlying proposed solution (i.e. MOGA-AMPazy) and that of the three other existing systems. The Pareto-optimal values obtained indicate that the proposed MOGA-AMPazy solution provides almost complete coverage involving the true Pareto front. Thus, the MOGA-AMPazy can achieve a favourable trade-off between coverage rates and energy consumption under a proper multi-objective sensor node deployment method. The proposed MOGA-AMPazy solution was measured in terms of coverage rates and energy consumption using two case studies to analyse the effect of the number of sensor nodes on the networks. The analysis establishes that coverage rates in dense networks are higher than in sparse networks. The same observation holds for energy consumption rates; in Case Study 2, less energy is required for all four algorithms. Furthermore, this study has measured the overall execution time of algorithms calculated along with the unconstrained test problem. Based on the results, it is evident that the MOGA-AMPazy algorithm proposed in this study has achieved shorter execution time compared to the other existing algorithms.

In order to increase coverage and conserve the dissipated energy of U-MWSN, the MOGA-AMPazy algorithm consists of two phases to change the locations, sensing ranges and communication range of MSNs. Compared to other algorithms, the key advantage of the MOGA-AMPazy algorithm is that, in addition to improving network coverage, the dissipated energy in mobility, sensing and redundant coverage are considered simultaneously by the objective function. The probabilistic model to improve the coverage of MWSNs with and without barriers was considered. Moreover, MOGA-AMPazy ensures maximum global convergence and has less computational complexity. In terms of network coverage and energy saving, the simulation results show that the MOGA-AMPazy algorithm outperforms the other algorithms by minimising coverage energy consumption and also saves costs by stimulating only the nodes needed.

Overall, the results and comparative analysis indicate that the MOGA-AMPazy is a better solution to the multi-objective sensor node deployment problem, outperforming the NSGA-II, SPEA2 and MOEA/D algorithms.

Table 6.1: Result of NSGAII, SPEA2, MOEA/D and MOGA-AMPazy Algorithms using Diversity, Hypervolume, and Inverted Generation Distance on Unconstrained Test Functions

Algorithm / Function	Metrics		NSGAII	SPEA2	MOEA/D	MOGA-AMPazy (Proposed Algorithm)
ZDT-1	IGD	Mean	0.00154	0.00151	0.00162	0.00101
		Std.Deviation	0.00133	0.00131	0.00143	0.00130
	HV	Mean	0.59800	0.65962	0.60730	0.74911
		Std.Deviation	0.04223	0.06211	0.05211	0.06744
	Δ	Mean	0.57250	0.64653	0.60532	0.54762
		Std.Deviation	0.09231	0.09961	0.09110	0.08995
ZDT-2	IGD	Mean	0.00162	0.00174	0.00140	0.00113
		Std.Deviation	0.00139	0.00141	0.00131	0.00212
	HV	Mean	0.56448	0.65962	0.61420	0.65322
		Std.Deviation	0.04102	0.06211	0.06101	0.05374
	Δ	Mean	0.61173	0.64653	0.60002	0.57656
		Std.Deviation	0.09980	0.09961	0.09962	0.09670
ZDT-3	IGD	Mean	0.00162	0.00174	0.00144	0.00157
		Std.Deviation	0.00139	0.00141	0.00133	0.00030
	HV	Mean	0.57768	0.71620	0.58871	0.70153
		Std.Deviation	0.04123	0.07211	0.04122	0.06476
	Δ	Mean	0.61173	0.64653	0.72555	0.60013
		Std.Deviation	0.09980	0.09961	0.08221	0.10311
ZDT-4	IGD	Mean	0.00172	0.00175	0.00158	0.00174
		Std.Deviation	0.00144	0.00145	0.00133	0.00113
	HV	Mean	0.56599	0.66855	0.60020	0.69965
		Std.Deviation	0.04145	0.03221	0.05105	0.06335
	Δ	Mean	0.61173	0.64653	0.64337	0.55887
		Std.Deviation	0.09980	0.09961	0.08221	0.09121
ZDT-6	IGD	Mean	0.00168	0.00164	0.00160	0.00167
		Std.Deviation	0.00439	0.00424	0.00141	0.00422
	HV	Mean	0.57448	0.65882	0.61137	0.72337
		Std.Deviation	0.04302	0.06200	0.05221	0.06537
	Δ	Mean	0.61173	0.70653	0.64102	0.54556
		Std.Deviation	0.09980	0.10200	0.09932	0.07534

Table 6.2: Results of the Effect of Node Density on Sparse Network

Tests	Algorithm	Maximum Travel Distances	Maximum dissipated energy in mobility	Average sensing radius	Execution time
A	NSGA-II	27.6	27.6	8.2	15.4
	SPEA2	20.3	20.3	8.2	17.1
	MOEA/D	21.4	21.3	8.2	16.5
	Proposed	9.5	9.5	8.2	9.3
B	NSGA-II	44.4	44.4	8.2	113.9
	SPEA2	20.7	20.7	8.2	115.8
	MOEA/D	21.6	21.5	8.2	112.7
	Proposed	19.7	19.7	8	81.7
C	NSGA-II	43.5	43.5	8.2	248.5
	SPEA2	20.8	20.8	8.2	251.5
	MOEA/D	22.3	22.5	8.2	252.2
	Proposed	17.6	17.6	7.6	202.4

CHAPTER 7: CONCLUSION

This chapter presents conclusions drawn from the research conducted in this thesis, and it is divided into four parts. Section 7.1 re-examines the research objectives to ensure that the aims and objectives of the research have been achieved. Section 7.2 lists the research contributions of this study. Section 7.3 presents the significance and limitations of the proposed solution. Finally, Section 7.4 discusses potential avenues for future research.

7.1 Retrospection of the Research Objectives

This research aimed to enhance mobile sensor node deployment performance in UWSNs by introducing a multi-objective optimisation genetic algorithm based on adaptive multi-parent crossover and fuzzy dominance. The solution is further improved by introducing prospect theory to guarantee convergence using risk evaluation. The simulated grid's complexity was improved as the two conflicting objectives were specified and presented uncertainty in the form of a stochastic underwater environment. This section revisits the four research objectives identified in Chapter 1 and affirms that they have been met.

7.1.1 Objective 1. To analyse several mobile underwater node deployment algorithms by evaluating coverage rate, node energy consumption, and execution time

The first objective was to empirically investigate and analyse state-of-the-art mobile underwater sensor node deployment solutions to establish the problem via experimentation in the UWSN environment. This study evaluated the performance of the existing methods by measuring the following attributes:

1. Coverage rates
2. Energy consumption
3. Pareto Optimal Metrics

4. Execution Time

Empirical analysis revealed that the existing solutions were likely to significantly degrade the performance of mobile sensor node deployment by seeking to simultaneously maximise coverage while minimising energy consumption. In some cases, longer execution time was found to result in higher coverage rates and greater energy consumption. This observation led us to propose a multi-objective technique solution to resolve the multiple and conflicting objectives in the UWSN environment.

7.1.2 Objective 2. To develop a hybrid fuzzy dominance-based decomposition technique and adaptive multi-parent crossover Genetic Algorithm for mobile UWSNs to optimise conflicting deployment objectives

The second objective was to propose a hybrid fuzzy dominance-based decomposition technique and adaptive multi-parent crossover genetic algorithm. This method adapts the original NSGA-II by introducing a hybridisation of adaptive multi-parent crossover genetic algorithm and fuzzy dominance-based decomposition techniques. The algorithm compares two solutions using a fuzzy Pareto dominance approach and applies the scalar decomposition algorithm when one solution cannot dominate the other on a fuzzy dominance level. The solution also proposes an adaptive multi-parent crossover (AMP) in the algorithm to balance exploration and exploitation by providing new offspring. The results of this study show that MOGA-AMPazy is an efficient and comprehensive deployment solution for mobile sensor nodes in UWSNs. The MOGA-AMPazy deployment algorithm appears to be more capable of resolving the multi-objective sensor deployment problem for underwater environments compared to other solutions.

7.1.3 Objective 3. To evaluate the proposed solution and to validate and compare its performance with that of other existing techniques

The third and final objective was to evaluate and validate the proposed solution's performance by implementing it in the UWSN environment and comparing it with state-of-the-art mobile underwater sensor node solutions. The following attributes were measured to evaluate the performance of the MOGA-AMPazy solution:

1. Coverage rates
2. Energy consumption
3. Pareto Optimal Metrics
4. Execution Time

The proposed MOGA-AMPazy solution's coverage rate was measured using two case studies that compared the number of sensor nodes against coverage rates. This analysis established that MOGA-AMPazy coverage rates are higher in dense networks than in sparse networks. The observations are similar for energy consumption rates; in Case Study 2, all four algorithms require less energy. Furthermore, this study measured the overall execution time of algorithms for unconstrained test problems. Based on the results, it was found that the MOGA-AMPazy algorithm proposed in this study achieved shorter execution time compared to the other existing algorithms.

The results and comparative analysis of MOGA-AMPazy against the other existing multi-objective algorithms indicate that MOGA-AMPazy works better on the multi-objective sensor node deployment problem than NSGA-II, SPEA2, and MOEA/D.

7.2 Research Contributions

This study's contributions to the existing body of knowledge are summarised below:

7.2.1 Thematic Taxonomy

This research presents a classification of UWSN-domain underwater mobile sensor deployment using different aspects of thematic taxonomy. The proposed taxonomy highlights the commonalities and differences among existing deployment solutions. The literature analysis also contributes by highlighting research gaps in the area of UWSNs and mobile deployment of underwater sensor networks.

7.2.2 MOGA-AMPazy Solution

This research proposed a multi-objective approach for a mobile underwater sensor called MOGA-AMPazy. The proposed solution hybridises a multi-objective genetic algorithm to introduce prospect theory to ensure convergence using risk evaluation in an uncertain stochastic underwater environment. The research has contributed to the body of knowledge by designating and discussing various statistical methods to determine the proposed solution's quality. In this regard, inverted generation distance (IGD), hypervolume, and diversity metrics determine the proposed MOGA-AMPazy algorithm's effectiveness. Furthermore, this research utilises five commonly used two-objective ZDT test instances as a benchmark test – namely ZDT-1, ZDT-2, ZDT-3, ZDT-4 and ZDT-6 – to employ specific problem characteristics to impose the underlying proposed solution and three other existing systems. The obtained Pareto-optimal values have shown that the proposed solution appears to provide almost complete coverage involving the true Pareto front.

7.2.3 Proposed Algorithms

This research assesses four algorithms for mobile sensor node deployment based on the multi-objective optimisation approach. The integration of these algorithms, which are listed below, distinguishes this research from other approaches to addressing the research problem described in Chapter 1. This research critically evaluated the effectiveness of the proposed algorithms using three different test settings and two case studies. The evaluation results showed that the proposed solution (i.e. MOGA-AMPazy) outperformed all of the existing algorithms in terms of coverage rates, energy consumption, Pareto-optimal assessment, and execution time.

1. Algorithm 1: To calculate coverage ratio
2. Algorithm 2: To calculate energy consumption ratio
3. Algorithm 3: To compare two solutions when one solution cannot dominate the other (Fuzzy Pareto Dominance)
4. Algorithm 4: To balance exploration and exploitation with new offspring (Adaptive Multi-Parent Crossover)
5. Algorithm 5: To guarantee convergence through risk evaluation (Prospect Theory)
6. Algorithm 6: To balance conflicts concerning node deployment objectives (Multi-Objective Optimization)

7.3 Significance and Limitations of the Proposed Solution

Some researchers have proposed a multi-objective solution in terrestrial wireless sensor networks (WSN), but such solutions could not be applied directly to UWSNs because of the underwater environment's unique features. This research conducted a detailed analysis to modify the original NSGA-II into the proposed solution, which was evaluated using four parameters – coverage rates, energy consumption, Pareto-optimal metrics, and execution time – to demonstrate the effectiveness of the proposed solution in addressing the problems identified in this research.

This research has some limitations – most notably the unavailability of the most recent data. We used openly available validated datasets, but it is worth mentioning that this research would have been more precise had it employed the most recent deployment datasets, such as military and surveillance monitoring datasets. However, these datasets are not available due to privacy and data protection policies.

7.4 Future Work

Even though this research has not examined all the problems associated with MOGA-AMPazy, various research efforts were conducted to address the identified problems. This research has only stressed incorporating multi-objective genetic algorithms with adaptive multi-parent crossover and fuzzy dominance algorithms. Investigating the open research challenges calls for further efforts in several future research directions. Further research is recommended to extend the analysis in the following areas:

1. This research has used the following four parameters to measure the proposed solution for experimental and concept validation. However, the proposed solution's data can also validate other parameters, such as detection rates, package data rates and throughput. Future researchers can enhance proposals by including these parameters for validation.
2. Future researchers are encouraged to use real-time datasets for mobile sensor nodes deployment to increase the validity of the proposed solution (i.e. MOGA-AMPazy).
3. The MOGA-AMPazy solution was implemented and validated in the UWSN environment. Due to limited resources, it was not feasible for this research to test the abilities of MOGA-AMPazy on target detection algorithms with routing protocols. Thus, future work may pursue the proposed deployment algorithm to improve routing protocols and target detection.
4. Furthermore, future researchers may implement the proposed solution in real-time multi-objective mobile sensor node deployment of the internet of underwater things environment, infrastructures and applications.

REFERENCES

Abbas, M. Z., Bakar, K. A., Arshad, M. A., Tayyab, M., Mohamed, M. H. (2016). Scalable nodes deployment algorithm for the monitoring of underwater pipeline. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, *14*(3), 1183–1191.

Abidin, Z. Z., Arshad, M. R., Ngah, U. K. (2011). A simulation based fly optimization algorithm for swarms of mini autonomous surface vehicles application.

Akyildiz, I. F., Pompili, D., Melodia, T. (2005). Underwater acoustic sensor networks: research challenges. *Ad hoc networks*, *3*(3), 257–279. doi: 10.1016/j.adhoc.2005.01.004

Antonelli, G., Caffaz, A., Casalino, G., Volpi, N. C., De Jong, I. B., De Palma, D., ... others (2016). The widely scalable mobile underwater sonar technology (wimust) h2020 project: First year status. In *Oceans 2016-shanghai* (pp. 1–8).

Arora, S., Singh, S. (2017). An improved butterfly optimization algorithm with chaos. *Journal of Intelligent & Fuzzy Systems*, *32*(1), 1079–1088.

Barolli, L., Koyama, A., Shiratori, N. (2003). A qos routing method for ad-hoc networks based on genetic algorithm. In *14th international workshop on database and expert systems applications, 2003. proceedings.* (pp. 175–179).

Beniwal, M., Singh, R. P., Sangwan, A. (2016). A localization scheme for underwater sensor networks without time synchronization. *Wireless Personal Communications*, *88*(3), 537–552.

Box, G. E. (1957). Evolutionary operation: A method for increasing industrial productivity. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, *6*(2), 81–101.

Cardei, M., Wu, J., Lu, M., Pervaiz, M. O. (2005). Maximum network lifetime in wireless sensor networks with adjustable sensing ranges. In *Wimob'2005), ieee international conference on wireless and mobile computing, networking and communications, 2005.* (Vol. 3, pp. 438–445).

Caruso, A., Paparella, F., Vieira, L. F. M., Erol, M., Gerla, M. (2008). The meandering current mobility model and its impact on underwater mobile sensor networks.

In *Ieee infocom 2008-the 27th conference on computer communications* (pp. 221–225).

Chakraborty, B. (2013). Particle swarm optimization algorithm and its hybrid variants for feature subset selection. In *Handbook of research on computational intelligence for engineering, science, and business* (pp. 449–466). IGI Global.

Chen, Z., Shatara, S., Tan, X. (2009). Modeling of biomimetic robotic fish propelled by an ionic polymer–metal composite caudal fin. *IEEE/ASME transactions on mechatronics*, *15*(3), 448–459.

Chengming, L., Xinnan, F., Gaifang, X., Hai, Y., Xuewu, Z. (2017). Optimization coverage of wireless nodes for target probabilistic positioning in underwater sensor network. In *2017 36th chinese control conference (ccc)* (pp. 9847–9852).

Coleman, P., Pellon, M., Zhang, Y. (2007). Towards human decision-making in multi-agent systems. *Cognitive psychology*, *1*, 7.

Dang, X., Shao, C., Hao, Z. (2019). Target detection coverage algorithm based on 3d-voronoi partition for three-dimensional wireless sensor networks. *Mobile Information Systems*, *2019*.

Deb, K. (1999). Multi-objective genetic algorithms: Problem difficulties and construction of test problems. *Evolutionary computation*, *7*(3), 205–230.

Deb, K., Pratap, A., Agarwal, S., Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, *6*(2), 182–197.

De Berg, M., Van Kreveld, M., Overmars, M., Schwarzkopf, O. (1997). Computational geometry. In *Computational geometry* (pp. 1–17). Springer.

Deif, D. S., Gadallah, Y. (2013). Classification of wireless sensor networks deployment techniques. *IEEE Communications Surveys & Tutorials*, *16*(2), 834–855.

Desale, S., Rasool, A., Andhale, S., Rane, P. (2015). Heuristic and meta-heuristic algorithms and their relevance to the real world: a survey. *Int. J. Comput. Eng. Res. Trends*, *351*(5), 2349–7084.

Dhawan, A., Vu, C. T., Zelikovsky, A., Li, Y., Prasad, S. K. (2006). Maximum lifetime of sensor networks with adjustable sensing range. In *Seventh acis international conference on software engineering, artificial intelligence, networking, and parallel/distributed computing (snpd'06)* (pp. 285–289).

Dorigo, M., Gambardella, L. M. (1997). Ant colony system: a cooperative learning approach to the traveling salesman problem. *IEEE Transactions on evolutionary computation*, *1*(1), 53–66.

Du, H., Xia, N., Zheng, R. (2014). Particle swarm inspired underwater sensor self-deployment. *Sensors*, *14*(8), 15262–15281.

Fang, W., Song, X., Wu, X., Sun, J., Hu, M. (2018). Novel efficient deployment schemes for sensor coverage in mobile wireless sensor networks. *Information Fusion*, *41*, 25–36.

Farmer, S., Poulos, C. D. (2015). Tax increment financing in chicago, il, building neoliberal exclusion one school at a time. *Critical Sociology*, *41*(1), 153–171.

Felemban, E., Shaikh, F. K., Qureshi, U. M., Sheikh, A. A., Qaisar, S. B. (2015). Underwater sensor network applications: A comprehensive survey. *International Journal of Distributed Sensor Networks*, *11*(11), 896832.

Friedman, G. J. (1959). Digital simulation of an evlutionary process. *General Systems Yearbook*, 171–184.

Gartner. (2018, October). *Gartner top 10 strategic technology trends for 2019.* Retrieved from `https://www.gartner.com/smarterwithgartner/gartner-top-10-strategic-technology-trends-for-2019/`

Grosenick, L., Clement, T. S., Fernald, R. D. (2007). Fish can infer social rank by observation alone. *Nature*, *445*(7126), 429–432.

Hackwood, S., Beni, G. (1992). Self-organization of sensors for swarm intelligence. In *Proceedings 1992 ieee international conference on robotics and automation* (pp. 819–820).

Hansen, M., Jaszkiewicz, A. (1998). *Evaluating the quality of approximations to the non-dominated set. technical university of denmark* (Tech. Rep.). IMM Technical

Report IMMREP1998-7.

Harizan, S., Kuila, P. (2020). Evolutionary algorithms for coverage and connectivity problems in wireless sensor networks: a study. In *Design frameworks for wireless networks* (pp. 257–280). Springer.

Holland, J. (1975). Adaptation in natural and artificial systems: an introductory analysis with application to biology. *Control and artificial intelligence*.

Hu, Y., Yang, S. X. (2004). A knowledge based genetic algorithm for path planning of a mobile robot. In *Ieee international conference on robotics and automation, 2004. proceedings. icra'04. 2004* (Vol. 5, pp. 4350–4355).

Jain, N., Virmani, D. (2017). Feature classification for underwater seismic prediction using wireless sensor nodes. In *Proceedings of the international multiconference of engineers and computer scientists* (Vol. 1).

Jawhar, I., Mohamed, N., Agrawal, D. P. (2011). Linear wireless sensor networks: Classification and applications. *Journal of Network and Computer Applications*, *34*(5), 1671–1682.

Jiang, P., Wang, X., Liu, J. (2018). A sensor redeployment algorithm based on virtual forces for underwater sensor networks. *Chinese Journal of Electronics*, *27*(2), 413–421.

Jiang, P., Xu, Y., Wu, F. (2016). Node self-deployment algorithm based on an uneven cluster with radius adjusting for underwater sensor networks. *Sensors*, *16*, 98. doi: 10.3390/s16010098

Kemna, S., Hamilton, M. J., Hughes, D. T., LePage, K. D. (2011). Adaptive autonomous underwater vehicles for littoral surveillance. *Intelligent service robotics*, *4*(4), 245.

Khatib, O. (1986). Real-time obstacle avoidance for manipulators and mobile robots. In *Autonomous robot vehicles* (pp. 396–404). Springer.

Khoufi, I., Hadded, M., Minet, P., Laouiti, A. (2015). Optimized trajectories of multi-robot deploying wireless sensor nodes. In *2015 ieee 11th international conference on wireless and mobile computing, networking and communications (wimob)* (pp. 724–731).

Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P. (1983). Optimization by simulated annealing. *science*, *220*(4598), 671–680.

Kumar, S., Perry, A., Moeller, C., Skvoretz, D., Ebbert, M., Ostrom, R., ... Czipott, P. (2004). Real-time tracking magnetic gradiometer for underwater mine detection. In *Oceans'04. mtts/ieee techno-ocean'04* (Vol. 2, pp. 874–878).

Lazaropoulos, A. G. (2016). Designing the undersea internet of things (iot) and machine-to-machine (m2m) communications using underwater acoustic mimo networks. *Trends in Renewable Energy*, *2*(1), 13–50. doi: 10.17737/tre.2016.2.1.0017

Lee, K. Y., El-Sharkawi, M. A. (2008). *Modern heuristic optimization techniques: theory and applications to power systems* (Vol. 39). John Wiley & Sons.

Li, X., Qian, J. (2003). Studies on artificial fish swarm optimization algorithm based on decomposition and coordination techniques. *Journal of circuits and systems*, *1*, 1–6.

Lin, C.-C., Deng, D.-J., Wang, S.-B. (2015). Extending the lifetime of dynamic underwater acoustic sensor networks using multi-population harmony search algorithm. *IEEE Sensors Journal*, *16*(11), 4034–4042.

Lin, W., Li, D., Tan, Y., Chen, J., Sun, T. (2008). Architecture of underwater acoustic sensor networks: A survey. In *Intelligent networks and intelligent systems, 2008. icinis'08. first international conference on* (pp. 155–159).

Lin, Y.-T. (2019). Three-dimensional boundary fitted parabolic-equation model of underwater sound propagation. *The Journal of the Acoustical Society of America*, *146*(3), 2058–2067.

Liu, C., Zhao, Z., Qu, W., Qiu, T., Sangaiah, A. K. (2019). A distributed node deployment algorithm for underwater wireless sensor networks based on virtual forces. *Journal of Systems Architecture*, *97*, 9–19.

Liu, Y., Gong, D., Sun, J., Jin, Y. (2017). A many-objective evolutionary algorithm using a one-by-one selection strategy. *IEEE Transactions on Cybernetics*, *47*(9), 2689–2702.

Lončar, I., Babić, A., Arbanas, B., Vasiljević, G., Petrović, T., Bogdan, S., Mišković,

N. (2019). A heterogeneous robotic swarm for long-term monitoring of marine environments. *Applied Sciences*, *9*(7), 1388.

Luo, H., Wu, K., Ruby, R., Hong, F., Guo, Z., Ni, L. M. (2017). Simulation and experimentation platforms for underwater acoustic sensor networks: Advancements and challenges. *ACM Computing Surveys (CSUR)*, *50*(2), 1–44.

Lurton, X. (2002). *An introduction to underwater acoustics: principles and applications*. Springer Science & Business Media.

Lv, X., Li, H., Li, H. (2017). A node coverage algorithm for a wireless-sensor-network-based water resources monitoring system. *Cluster Computing*, *20*(4), 3061–3070.

Mahboubi, H., Aghdam, A. G. (2016). Distributed deployment algorithms for coverage improvement in a network of wireless mobile sensors: Relocation by virtual force. *IEEE Transactions on Control of Network Systems*, *4*(4), 736–748.

Majid, A., Azam, I., Waheed, A., Zain-ul Abidin, M., Hafeez, T., Khan, Z. A., ... Javaid, N. (2016). An energy efficient and balanced energy consumption cluster based routing protocol for underwater wireless sensor networks. In *Advanced information networking and applications (aina), 2016 ieee 30th international conference on* (pp. 324–333).

MarketsandMarkets. (2019, December). *Wireless sensor network market worth 93.86 billion usd by 2023.* Retrieved from `https://www.marketsandmarkets.com/PressReleases/wireless-sensor-network.asp`

MarketWatch. (2019, December). *Wireless sensor network market 2019 global industry trends, statistics, size, share, growth factors, regional analysis, competitive landscape and forecast to 2022.* Retrieved from `https://www.marketwatch.com/press-release/wireless-sensor-network-market-2019-global-industry-trends-statistics-size-share-growth-factors-regional-analysis-competitive-landscape-and-forecast-to-2022-2019-12-18`

Mei, Y., Lu, Y.-H., Hu, Y. C., Lee, C. G. (2004). Energy-efficient motion planning for mobile robots. In *Ieee international conference on robotics and automation, 2004. proceedings. icra'04. 2004* (Vol. 5, pp. 4344–4349).

Merico, D. (2010). Tracking with high-density, large-scale wireless sensor networks. *Journal of Ambient Intelligence and Smart Environments*, *2*(4), 441–442.

Mohamed, S. M., Hamza, H. S., Saroit, I. A. (2017). Coverage in mobile wireless sensor networks (m-wsn): A survey. *Computer Communications*, *110*, 133–150.

Ojha, T., Misra, S., Obaidat, M. S. (2020). Seal: Self-adaptive auv-based localization for sparsely deployed underwater sensor networks. *Computer Communications*, *154*, 204–215.

Pan, W.-T. (2011). A new evolutionary computation approach: fruit fly optimization algorithm. In *2011 conference of digital technology and innovation management* (pp. 382–391).

Pedlosky, J. (2013). *Ocean circulation theory*. Springer Science & Business Media.

Priyadarshini, R. R., Sivakumar, N. (2021). Cluster head selection based on minimum connected dominating set and bi-partite inspired methodology for energy conservation in wsns. *Journal of King Saud University-Computer and Information Sciences*, *33*(9), 1132–1144.

Qu, Y., Georgakopoulos, S. V. (2012). A centralized algorithm for prolonging the lifetime of wireless sensor networks using particle swarm optimization. In *Wamicon 2012 ieee wireless & microwave technology conference* (pp. 1–6).

Raj Priyadarshini, R., Sivakumar, N. (2020). Enhancing coverage and connectivity using energy prediction method in underwater acoustic wsn. *Journal of Ambient Intelligence and Humanized Computing*, *11*(7), 2751–2760.

Reshko, G., Mason, M. T., Nourbakhsh, I. R. (2002). *Rapid prototyping of small robots*. Carnegie Mellon University, The Robotics Institute.

Saeed, H., Ali, S., Rashid, S., Qaisar, S., Felemban, E. (2014). Reliable monitoring of oil and gas pipelines using wireless sensor network (wsn)—remong. In *2014 9th international conference on system of systems engineering (sose)* (pp. 230–235).

Shen, W., Zhang, C., Zhang, X., Shi, J. (2019). A fully distributed deployment algorithm for underwater strong k-barrier coverage using mobile sensors. *International Journal of Distributed Sensor Networks*, *15*(4), 1550147719838196.

Singh, A., Sharma, S., Singh, J. (2021). Nature-inspired algorithms for wireless sensor networks: A comprehensive survey. *Computer Science Review*, *39*, 100342.

Spector, L. (1994). *Advances in genetic programming* (Vol. 1). MIT Press.

Stojanovic, M. (2003). Acoustic (underwater) communications. *Wiley Encyclopedia of Telecommunications*.

Tam, N. T., Hung, T. H., Binh, H. T. T., et al. (2021). A decomposition-based multi-objective optimization approach for balancing the energy consumption of wireless sensor networks. *Applied Soft Computing*, *107*, 107365.

Tversky, A., Kahneman, D. (1992). Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and uncertainty*, *5*(4), 297–323.

Ullah, I., Liu, Y., Su, X., Kim, P. (2019). Efficient and accurate target localization in underwater environment. *IEEE Access*, *7*, 101415–101426.

Vilela, J., Kashino, Z., Ly, R., Nejat, G., Benhabib, B. (2016). A dynamic approach to sensor network deployment for mobile-target detection in unstructured, expanding search areas. *IEEE Sensors Journal*, *16*(11), 4405–4417.

Von Foerster, H. (1960). On self-organizing systems and their environments. *Self-organizing systems*, 31–50.

Wang, G., Cao, G., La Porta, T. F. (2006). Movement-assisted sensor deployment. *IEEE Transactions on Mobile Computing*, *5*(6), 640–652.

Wang, H., Li, Y., Chang, T., Chang, S., Fan, Y. (2018). Event-driven sensor deployment in an underwater environment using a distributed hybrid fish swarm optimization algorithm. *Applied Sciences*, *8*(9), 1638.

Wang, H., Li, Y., Zhang, L., Fan, Y., Li, Z. (2019). A self-deployment algorithm for maintaining maximum coverage and connectivity in underwater acoustic sensor networks based on an ant colony optimization. *Applied Sciences*, *9*(7), 1479.

Wang, W., Wu, H., Kong, X., Zhang, Y., Ye, Y., Zeng, Z., . . . Zhang, Q. (2022). Reinforcement learning-based dynamic position control of mobile node for ocean sensor networks. *Transactions of the Institute of Measurement and Control*, *44*(4), 926–940.

Wang, X., Li, N., Liu, F., Ding, Y. (2017). The double-coverage algorithm for mobile node deployment in underwater sensor network. In *International conference on geo-spatial knowledge and intelligence* (pp. 198–211).

Xiao, Y. (2010). *Underwater acoustic sensor networks*. CRC Press.

Yakıcı, E., Karatas, M. (2021). Solving a multi-objective heterogeneous sensor network location problem with genetic algorithm. *Computer Networks*, *192*, 108041.

Zambrano-Vega, C., Nebro, A. J., García-Nieto, J., Aldana-Montes, J. F. (2017). Comparing multi-objective metaheuristics for solving a three-objective formulation of multiple sequence alignment. *Progress in Artificial Intelligence*, *6*(3), 195–210.

Zhang, Y., Wang, M., Liang, J., Zhang, H., Chen, W., Jiang, S. (2017). Coverage enhancing of 3d underwater sensor networks based on improved fruit fly optimization algorithm. *Soft Computing*, *21*(20), 6019–6029.

Zitzler, E., Deb, K., Thiele, L. (1999). Comparison of multiobjective evolutionary algorithms on test functions of different difficulty. In *Proceedings of the 1999 genetic and evolutionary computation conference. workshop program* (pp. 121–122).

Zou, J., Gundry, S., Kusyk, J., Sahin, C. S., Uyar, M. Ü. (2013). Bio-inspired topology control mechanism for autonomous underwater vehicles used in maritime surveillance. In *2013 ieee international conference on technologies for homeland security (hst)* (pp. 201–206).