

Cooperative localisation in underwater robotic swarms for ocean bottom seismic imaging.

SABRA, A.

2021

The author of this thesis retains the right to be identified as such on any occasion in which content from this thesis is referenced or re-used. The licence under which this thesis is distributed applies to the text and any original images only – re-use of any third-party content must still be cleared with the original copyright holder.

**Cooperative Localisation in Underwater
Robotic Swarms for Ocean Bottom
Seismic Imaging**

Adham Sabra

PhD

2021

Cooperative Localisation in Underwater Robotic Swarms for Ocean Bottom Seismic Imaging

Author: Adham Sabra



A THESIS SUBMITTED IN PARTIAL FULFILMENT FOR THE DEGREE
OF DOCTOR OF PHILOSOPHY IN ROBOTICS
AT THE SCHOOL OF ENGINEERING
ROBERT GORDON UNIVERSITY
ABERDEEN, SCOTLAND, UNITED KINGDOM

February 2021

Abstract

Spatial information in wide variety of sub-sea applications such as deep sea exploration, environmental monitoring, geological and ecological research, and samples collection must be collected alongside the data modality of interest. Ocean bottom seismic surveys are vital for oil and gas exploration and productivity enhancement of an existing production facility. Ocean bottom seismic sensors are deployed on the seabed to acquire those surveys. Node deployment methods used in industry today are costly, time-consuming and incapable in deep oceans. Autonomous deployment of ocean bottom seismic nodes implemented by a swarm of Autonomous Underwater Vehicles (AUVs) is proposed. In autonomous deployment of ocean bottom seismic nodes, a swarm of seismic sensors-equipped AUVs are deployed to communicate and collaboratively achieve ocean bottom seismic imaging. However, the severely limited bandwidth of the underwater acoustic communications and the high cost of maritime assets limit the number of AUVs that can be deployed for experiments.

A holistic fuzzy-based localisation framework for large underwater robotic swarms (i.e., in hundreds) to dynamically fuse multiple position estimates of an autonomous underwater vehicle is proposed. Simplicity, flexibility and scalability are the main three advantages inherent in the proposed localisation framework when compared to other traditional and commonly adopted underwater localisation methods such as the Extended Kalman Filter. The proposed fuzzy-based localisation algorithm improves the entire swarm mean localisation error and standard deviation by 16.53% and 35.17% respectively at swarm size of 150 AUVs when compared to the Extended Kalman Filter based localisation with round-robin scheduling. The proposed fuzzy based localisation method requires fuzzy rules and fuzzy set parameters tuning if deployment scenario is changed.

Therefore, a cooperative localisation scheme that relies on a scalar localisation confidence value is proposed. A swarm subset is navigationally aided by ultra-short baseline

and a swarm subset (i.e., navigation beacons) is configured to broadcast navigation aids (i.e., range-only) once their confidence values are higher than a predetermined confidence threshold. The confidence value and navigation beacons subset size are two key parameters for the proposed algorithm so that they are optimised using the evolutionary multi-objective optimisation algorithm NSGA-II to enhance its localisation performance. Confidence value based localisation is proposed to control the cooperation dynamics among the swarm agents in terms of acoustic exteroceptive sensors aiding. Given the error characteristics of a commercially available ultra-short baseline system and the covariance matrix of a trilaterated underwater vehicle position, Extended Kalman Filter-based acoustic exteroceptive sensors-aided dead reckoning navigation is performed and controlled by the vehicle's confidence value. The proposed confidence-based localisation algorithm has significantly improved the entire swarm mean localisation error by 67.10% and 59.28% when compared to the fuzzy-based and round-robin Extended Kalman Filter-based localisation methods respectively at swarm size of 150 AUVs.

The proposed fuzzy-based and confidence-based localisation algorithms for cooperative underwater robotic swarms are validated on co-simulation platform. A physics-based co-simulation platform that considers environment's hydrodynamics, industrial grade inertial measurement unit and underwater acoustic communications characteristics is implemented for validation and optimisation purposes.

Acknowledgements

This thesis would not have been possible without the help and support of many friends and colleagues who made the past four years exceptionally fulfilling experience. I would like to thank my supervisor Dr. Wai-Keung Fung who strongly supported and guided me all the way through my research, findings, publications and thesis. I am grateful to Dr. Radhakrishna Prabhu for his support, suggestions and comments especially during the final year of my PhD journey. I would also like to thank the school of engineering administration team at Robert Gordon University (RGU) – especially Rosslyn Shanks and Kirsty Stevenson for their help and continued support throughout the PhD journey. During the second year of my PhD I had the opportunity to visit Edinburgh Centre for Robotics and Ocean Systems Laboratory at Heriot–Watt University (HWU) as a research visiting scholar. Many thanks to ORCA Hub and USMART teams at HWU - especially Prof. Yvan Petillot and Dr. Mauro Dragone for giving me the opportunity of joining the team on a research visit. I very much appreciate Dr. Mauro Dragone’s continued support, feedback and suggestions. I would also like to thank my fellow labmates from the School of Engineering at RGU and the School of Engineering & Physical Sciences at HWU – especially Alasdair, Osama, Joshua, Rahul, Marwa and Jonatan for their stimulating discussions.

I had the opportunity to collaborate with Autonomous Robotics Ltd during the third year of my PhD on Flying Nodes project which made my PhD research even more exciting; I would like to thank Dave Grant and Arran Holloway from Autonomous Robotics Ltd and Mark Robertson from The Oil & Gas Innovation Centre for their continued support and discussions.

At the beginning of the fourth and final year of my PhD I started working full-time as a Software and AI Development Engineer on another exciting project (machine vision for industrial robotic arms) with Astech Projects Ltd and the University of Liverpool. I would like to thank my team from Liverpool University and Astech Projects - especially

Prof. Frans Coenen and Matt Neidhardt for their continued support during COVID–19 outbreak.

The last four years would not have been the same without the many great people I met in Aberdeen, Edinburgh and Liverpool who throughout the years shared with me great personal experiences; I wish them all the best.

Last but not least, I am exceptionally grateful to my mum, dad and siblings for their endless love and support; This would have never been possible without you.

This research was funded in part by the Oil and Gas Innovation Centre - OGIC (currently part of the Oil and Gas Technology Centre - OGTC), 20 Queens Road, Aberdeen, AB15 4ZT, Scotland; and Autonomous Robotics Ltd, Eastleigh Court, Bishopstrow, Warminster, BA12 9HW, England. The PhD scholarship was funded by the School of Engineering at Robert Gordon University in Aberdeen.

“I have no special talents. I am only passionately curious.”

– Albert Einstein

Author's contributions

Publications

- **Sabra A**, Fung WK. Dynamic localization plan for underwater mobile sensor nodes using fuzzy decision support system. In: OCEANS 2017 - Anchorage; 2017. p. 1-8.
- **Sabra A**, Fung WK, Radhakrishna P. Confidence-based Underwater Localization Scheme for Large-Scale Mobile Sensor Networks. In: OCEANS 2018 MTS/IEEE Charleston; 2018. p. 1-6.
- **Sabra A**, Fung WK, Churn P. Multi-objective Optimization of Confidence-Based Localization in Large-Scale Underwater Robotic Swarms. In: Correll N, Schwager M, Otte M, editors. Distributed Autonomous Robotic Systems. Cham: Springer International Publishing; 2019. p. 109-123.
- **Sabra A**, Fung WK. A Fuzzy Cooperative Localisation Framework for Underwater Robotic Swarms. *Sensors*. 2020 Sep; 20(19): 5496.
- **Sabra A**, Fung WK, Dragone M, Petillot Y. An Optimised Confidence-based Cooperative Navigation for Underwater Robotic Swarms. *IEEE Transactions on Mobile Computing* (Submitted)

Funding and media coverage

- The Oil and Gas Innovation Centre (OGIC) research fund; [Offshore Energy news](#), [insider.co.uk](#).

Contents

Abstract	ii
Acknowledgements	iv
Author's contributions	vi
1 Introduction	1
1.1 Motivation	2
1.2 Research Challenges	4
1.3 Research Proposal	5
1.4 Research Aim and Objectives	6
1.5 Contributions	7
1.6 Thesis Structure	8
2 Seismic Imaging Literature	10
2.1 Introduction	10
2.2 Seismic Imaging	11
2.2.1 Seismic Waves	12
2.2.2 Seismic Sensors	12
2.3 Marine Seismic	13
2.3.1 Sensor Deployment Geometry	13
2.3.2 Data Dimensionality	15
2.4 Robotic Systems in Marine Seismic	16
2.4.1 Autonomous Towed Streamers	16
2.4.2 Autonomous Ocean Bottom Nodes Deployment	17
2.5 Conclusion	21
3 Underwater Localisation Literature	22

3.1	Introduction	22
3.2	Inertial Navigation System	23
3.2.1	Pose Estimation	26
3.2.2	The Extended Kalman Filter (EKF)	27
3.3	Underwater Localisation Algorithms	30
3.3.1	Sensor Network Localisation	31
3.3.2	Least-Squares Trilateration	34
3.3.3	Underwater Multi-vehicle Localisation	39
3.4	Underwater Communications	41
3.4.1	Physical Layer	42
3.4.2	Medium Access Control Layer	43
3.5	Conclusion	44
4	Underwater Robotics Simulation Platform	46
4.1	Underwater Acoustic Communications	47
4.1.1	Acoustic Modem and Channel Models	48
4.1.2	Experiments and UnetStack Simulation	49
4.2	Simulation Platform Architecture	52
4.3	Simulation Settings and Scenario	55
4.4	Summary	57
5	A Fuzzy Localisation Framework for Underwater Robotic Swarms	58
5.1	Introduction	58
5.2	Fuzzy-based localisation	60
5.3	Proof Of Concept Simulation	63
5.3.1	Localisation Error Models	64
5.3.2	Simulation Settings	65
5.3.3	Results and Analysis	67
5.4	Physics-based Simulation	70
5.4.1	Implementation	71
5.4.2	Simulation Scenario and Settings	75
5.4.3	Results and Analysis	76
5.5	Summary	80
6	Confidence based Underwater Swarm Localisation and Optimisation	82
6.1	Introduction	82
6.2	Background	84

6.2.1	Evolutionary Multi-objective Optimisation (EMO)	84
6.2.2	Hierarchical Underwater Localisation	87
6.3	Confidence-based localisation Algorithm	88
6.3.1	Confidence Update Rules	88
6.3.2	Parameters Optimisation	91
6.4	Simulation	92
6.4.1	Error Characteristics for Confidence Update	92
6.4.2	Simulation Settings	93
6.4.3	Results and Analysis	94
6.5	Summary	103
7	Confidence based Localisation for Cooperative Underwater Robotic Swarms using the Extended Kalman Filter	105
7.1	Introduction	106
7.2	Cooperative Swarm Localisation	107
7.2.1	Algorithm Overview	107
7.2.2	Confidence Value Update	111
7.2.3	Multilateration in the Presence of Uncertainty	112
7.3	Simulation	116
7.3.1	Deployment Scenario and Settings	116
7.3.2	Results and Analysis	117
7.4	Summary	121
8	Conclusion and Future Work	126
8.1	Summary	126
8.2	Future Directions	128
	Bibliography	131
A	A Fuzzy Localisation Framework for Underwater Robotic Swarms	143
A.1	Fuzzy rules - MATLAB simulation	143
A.2	Fuzzy rules - Webots simulation	144
A.3	One-tail two sample <i>t</i> -test	146
B	Confidence based Localisation for Cooperative Underwater Robotic Swarms using the Extended Kalman Filter	147
B.1	One-tail two sample <i>t</i> -test	147

List of Tables

3.1	Underwater wireless communications technology	42
4.1	UnetStack simulation settings	51
4.2	Ellipse 2 micro IMU properties	57
5.1	Simulation parameters	67
5.2	Ellipse 2 micro IMU properties	76
5.3	Simulation parameters	77
5.4	Intra-swarm communication modem and channel parameters	77
6.1	Simulation parameters	94
7.1	Intra-swarm communication modem and channel parameters	117
7.2	Simulation parameters	118
A.1	One-tail two sample t -test	146
B.1	One-tail two sample t -test	147

List of Figures

1.1	Ocean bottom seismic node, ROV and AUV	2
1.2	Navigation, guidance, and control of an AUV	3
1.3	Underwater wireless network of AUVs and fixed sensors	4
2.1	Towed streamer	14
2.2	Ocean bottom nodes	14
2.3	2D and 3D seismic images	15
2.4	WiMUST system	16
2.5	Wave glider vehicle	17
2.6	Ocean bottom nodes - ROV deployment	18
2.7	Ocean bottom nodes - autonomous deployment	19
2.8	Seismic survey cost and duration with number of source vessels	20
2.9	Seismic survey cost with nodes separation distances	20
3.1	Commercial Time of Flight acoustic navigation methods	30
3.2	Unknown node localisation process in large scale localisation	32
3.3	Localisation error of LLNP and SLMP methods	33
3.4	Trilateration problem	34
3.5	Instantaneous localisation error	39
3.6	Time-division multiple access	44
4.1	UnetStack acoustic modem and channel models	50
4.2	Network topology	51
4.3	AUV plan and elevation	52
4.4	Webots simulation scene example	53
4.5	Simulation platform; Webots-UnetStack	54
4.6	Network mobility considerations when trilateration is performed	55
4.7	FSM to guide the AUV	56

4.8	AUVs trajectories	56
4.9	IMU simulation model	57
5.1	Input-output mapping	61
5.2	Fuzzy-based localisation approach	63
5.3	USBL localisation error	64
5.4	DVL/INS localisation error	65
5.5	The implemented fuzzy system	66
5.6	A Predefined AUV trajectory	68
5.7	Localisation performance - one AUV	68
5.8	Normalised localisation performance - one AUV	69
5.9	Localisation performance - 3 AUVs	69
5.10	Normalised localisation performance - 3 AUVs	70
5.11	Fuzzy inference variable inputs	73
5.12	The implemented fuzzy-based localisation approach	74
5.13	Round-robin scheduling	75
5.14	USBL and Navigation Beacons aids	75
5.15	Fuzzy-based and Round robin based localisation	78
5.16	The entire swarm localisation performance	79
5.17	Instantaneous localisation error	80
6.1	Non-dominated and crowding-distance sortings	86
6.2	NSGA-II multi-objective optimisation	87
6.3	Node density with node's communication range	95
6.4	Confidence threshold and AUV density impacts	96
6.5	Pareto Front	97
6.6	Pareto optimal solutions	98
6.7	Four selected optimal solutions scores	98
6.8	Localisation errors and confidence values histograms	99
6.9	Instantaneous localisation error and confidence value	100
6.10	Entire swarm localisation errors and confidence values histograms	101
6.11	Mean localisation error with swarm size	102
6.12	Error standard deviation with swarm size	102
7.1	The proposed navigation system block diagram	110
7.2	Decoupled unscented trilateration	115
7.3	Pareto-optimal set	119

7.4	Pareto Front	123
7.5	The entire localisation error	124
7.6	An example instantaneous root mean square localisation error	125
7.7	An AUV's RMSE with $3\text{-}\sigma$ error bounds	125
8.1	Task allocation in a heterogeneous robotic swarm	129

List of Acronyms

- ACOMMS** Acoustic Communications.
- AHRS** Attitude Heading Reference System.
- API** Application Programming Interface.
- AUV** Autonomous Underwater Vehicle.
- BPSK** Binary Phase-Shift Keying.
- BSLA** Best Suitable Localisation Algorithm.
- CDMA** Code Division Multiple Access.
- CEP** Circular Error Probability.
- CSAC** Chip Scale Atomic Clock.
- CSMA** Carrier-Sense Multiple Access.
- DPSK** Differential Phase-Shift Keying.
- DR** Dead Reckoning.
- DVL** Doppler Velocity Log.
- EIF** Extended Information Filter.
- EKF** Extended Kalman Filter.
- EMO** Evolutionary Multi-objective Optimisation.
- EO** Evolutionary Optimisation.
- FDMA** Frequency-division multiple access.

FSK Frequency-Shift Keying.

GNSS Global Navigation Satellite System.

IMU Inertial Measurement Unit.

INS Inertial Navigation System.

LBL Long Baseline.

LLNP Large-scale Localisation scheme with No Prediction.

MAC Medium Access Control.

MDSF Multi Data Sensor Fusion.

MEMS Microelectromechanical systems.

NB Navigation Beacon.

OBN Ocean Bottom Node.

ODE Open Dynamics Engine.

OFDM Orthogonal Frequency-Division Multiplexing.

OWTT One-Way Travel Time.

PSK Phase-Shift Keying.

PSO Particle Swarm Optimisation.

QAM Quadrature Amplitude Modulation.

RMSE Root Mean Square Error.

ROV Remotely Operated Vehicle.

SBL Short Baseline.

SEP Spherical Error Probability.

SLAM Simultaneous Localisation and Mapping.

SLMP Scalable Localisation scheme with Mobility Prediction.

SNR Signal-to-Noise Ratio.

TDMA Time-division multiple access.

TDoA Time Difference of Arrival.

ToA Time of Arrival.

ToF Time of Flight.

TWTT Two-Way Travel Time.

UKF Unscented Kalman Filter.

USBL Ultra-short Baseline.

USV Unmanned Surface Vehicle.

UT Unscented Transform.

UWSN Underwater Wireless Sensor Network.

Chapter 1

Introduction

Seventy-one percent of the earth's surface is covered by water and it is commonly believed that we know more about the space than deep oceans [1]. Spatial information in various offshore applications such as deep-sea exploration [2], environmental monitoring, geological and ecological research [3] must be collected alongside the data modality of interest. Advancement in robotics and communications technology have given rise to underwater multi-agent robotic systems to achieve those marine missions. Underwater navigation technology is fundamental to achieve those missions and its inaccuracy is becoming an impediment to deep oceans missions.

This research is primarily driven by the need for an autonomous and cost effective ocean bottom seismic imaging method for oil and gas production [4]. Seismic imaging is being used in oil and gas industry in either fossil fuel exploration or productivity enhancement of an existing onshore or offshore oil field. Four-dimensional (4D) and three-dimensional (3D) seismic surveys currently represent a significant percentage of overall seismic surveys [5]. Seismic sensors are deployed on the seabed or sea surface in a mesh-like geometry for 3D seismic data acquisition [5]. Four-dimensional seismic data is acquired when 3D surveys are repeated over time over the same area of interest [5]. Ocean bottom seismic sensors can provide high resolution 3D and 4D seismic images of sub-seabed which are vital for oil and gas exploration and productivity enhancement of an existent production facility. Recent developments in ocean bottom seismic sensors have given rise to the need for a reliable and cost effective deployment method. Ocean bottom seismic sensors are deployed in deep oceans using [Remotely Operated Vehicle \(ROV\)](#) equipped with robotic arm and driven by on board crew. This is very costly

underwater sensor networks (e.g., underwater swarm robotics) such as deep sea exploration and environmental monitoring have enabled and motivated underwater robotics research for decades [15]. Localisation is one of the most critical problems in robotic swarms as it is required to be successfully obtained in advance of nodes' guidance and control in location aware applications. The navigation module of an autonomous node estimates its position and velocity and feeds them into the control and guidance modules [16]. Figure 1.2 shows the navigation, control and guidance scheme of an AUV.

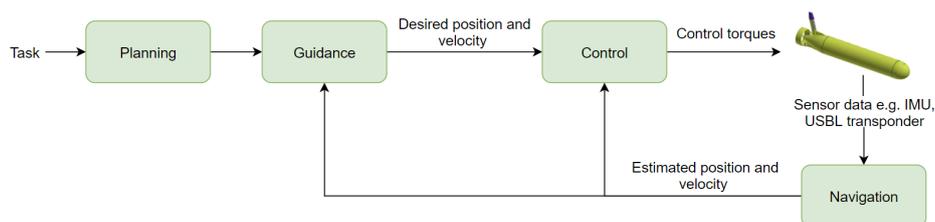


Figure 1.2: Navigation, guidance, and control scheme of an AUV.

In a sub-sea mission, a swarm of mobile and/or static nodes are typically deployed to communicate and collaboratively achieve various predefined tasks in underwater environments. In order to successfully complete assigned missions, locations of individual nodes must be known and tracked all the time in the whole operation for successful missions. Figure 1.3 shows a sketch of an underwater wireless sensor network represented by a swarm of mobile and static nodes.

Given the absence of the [Global Navigation Satellite System \(GNSS\)](#) in underwater environments, [AUV](#) navigation predominately relies on proprioceptive sensors such as [Inertial Measurement Unit \(IMU\)](#) integrated with [Doppler Velocity Log \(DVL\)](#) [17]. [IMU](#)-based navigation is prone to drift and the [DVL](#) is limited to operate close to seabed [18]. Therefore, acoustic exteroceptive sensors are usually utilised as external navigation aids and integrated with the proprioceptive sensors position estimate to reduce the estimated position uncertainty. The severely limited bandwidth and long latency of underwater acoustic communication limit the number of [AUVs](#) that can be deployed at once to collaboratively complete a mission [19, 20, 21]. Underwater multi-agent robotic systems mainly rely on acoustic communication to exchange information among team members with an average propagation speed of 1500 m/s (i.e., speed of sound in water) with a maximum bit rate of around 60 kbps [22]. On the contrary, information among members of multi-agent terrestrial or aerial robotic systems are exchanged in the speed

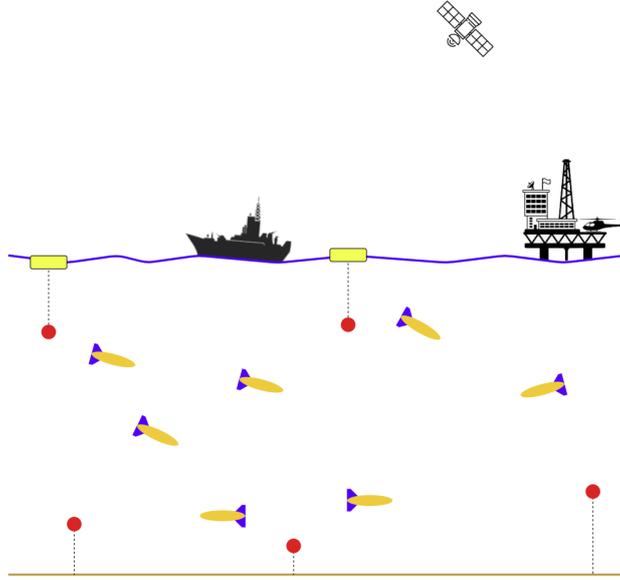


Figure 1.3: Underwater wireless network of AUVs (yellow vessels with blue fins) and anchored sensors (red).

of light of 3×10^8 m/s with bit rate in Mbps.

Advancement in both underwater sensing and robotics together with the rising need for autonomous underwater surveys in deep oceans have entailed expanding the state-of-the-art in underwater swarm robotics navigation. The navigation framework investigated herein is motivated by the robustness and scalability requirements of an underwater robotic swarms for ocean bottom seismic or similar sub-sea missions. The robustness of the proposed navigation frameworks can be stemmed from four different factors, namely redundancy, decentralisation, simplicity of the individuals and multiplicity of sensing.

1.2 Research Challenges

The nature of ocean bottom seismic imaging application that requires swarm deployment of AUVs together with the nature of underwater communication channel pose many challenges that are simply not present in terrestrial or aerial localisation problems such as:

- Unaffordable expenses associated with the field experiments of underwater robotic swarms due to the high cost of maritime assets.

- Different localisation methods have their own merits and limitations which make them suitable for certain marine applications and under certain operating conditions. Therefore, there is no single localisation method that is able to maintain high localisation precision and accuracy under different operating conditions.
- Current trends in underwater localisation rely on fusing multiple localisation estimates using Gaussian filters family such as the [Extended Kalman Filter \(EKF\)](#) [23] which requires substantial efforts when integrating new localisation methods in an existent [EKF](#)-based navigation.
- The severely limited bandwidth of the underwater acoustic communications limits the number of [AUVs](#) (i.e., 1–10 [AUVs](#)) that can be navigationally aided by exteroceptive sensors represented by [Time of Flight \(ToF\)](#) acoustic navigation systems.
- The limited bandwidth of the underwater acoustic communications also limits the number of [AUVs](#) that can broadcast localisation aids in multi-agent cooperative localisation. Therefore, a medium access control protocol should be carefully considered which drastically limits swarm sizes deployed in missions due to lack of localisation aids.
- Difficult and rapidly varying acoustic communication channel conditions that depends on changes in water temperature, geometry of the channel, roughness of the sea surface. This results in high packet loss as compared to electromagnetic signals due to the presence of large Doppler spread caused by time variation of the acoustic channel.

1.3 Research Proposal

To overcome the challenges stated in Section 1.2, the following research steps are considered.

1. Ocean bottom seismic imaging solutions are reviewed to identify their limitations in obtaining cost-efficient high resolution seismic images. This includes identifications of commercially available or under development solutions to clear any possible conflict of interests and proposing a fully autonomous robotic system for ocean bottom seismic to overcome the limitations of the current deployment methods.
2. The current underwater navigation trends in both wireless sensor networks and

cooperative autonomous vehicles to tackle the navigation problem of underwater robotic swarms for ocean bottom seismic imaging are investigated.

3. Implementing a simulation platform that considers both dynamics and underwater communications simulation engines to perform a thorough evaluation of the proposed localisation algorithm in terms of scalability and localisation accuracy.
4. Introducing a holistic underwater navigation framework that harnesses all available underwater navigation schemes under different operating conditions using fuzzy decision support system to facilitate integrating newly introduced localisation methods.
5. Proposing a confidence-based navigation framework and tuning the proposed navigation framework's parameters so that the utilisation of the underwater acoustic channel is optimised. Therefore, intra-swarm collaboration for cooperative localisation is maximised given the constraints imposed by the underwater acoustic communication channel.

1.4 Research Aim and Objectives

The aim of this thesis is to develop a robust and scalable localisation method for underwater robotic swarms for deep-sea missions such as exploration, environmental monitoring, geological and ecological research. To achieve this aim, the following objectives are identified to influence the research development taken in this thesis.

- Develop a comprehensive localisation framework for large scale underwater robotic swarms to overcome the limitations of the Gaussian filters family based navigation methods in design simplicity, localisation accuracy and scalability.
- Develop a confidence-based cooperative swarm navigation algorithm that guarantees distributive harmony among the nodes with acknowledging the uncertainty associated with the localisation process to nominate only a limited number of AUVs (i.e., AUVs with high confidence on their localisation accuracy) to broadcast localisation aids.
- Optimise the utilisation of the underwater acoustic channel so that intra-swarm cooperation is maximised to improve the localisation accuracy when swarm size increases.

- Evaluate and compare the proposed localisation algorithms' performances in terms of scalability and localisation accuracy with other underwater localisation methods in literature that are all evaluated on a developed co-simulation platform that considers both dynamics and underwater communications simulation engines.

1.5 Contributions

The main focus of this thesis is to expand the state-of-the-art in underwater swarm navigation by proposing a distributive cooperative localisation algorithm. This thesis provides a set of methodologies to achieve the objectives listed in the previous section. The contribution of this thesis is summarised as follows:

- An extensive investigation into the current state-of-the-art techniques in ocean bottom seismic imaging and underwater navigation which resulted in establishing research collaboration with Autonomous Robotics Ltd [24] and securing research fund of GBP 27660 from the Oil and Gas Innovation Centre (currently part of the Oil and Gas Technology Centre [25]). The project with Autonomous Robotics Ltd was completed in October 2019.
- A physics-based co-simulation platform that considers underwater environments hydrodynamics and underwater acoustic communications characteristics to validate the proposed localisation algorithms in terms of localisation accuracy and scalability in a cost-effective way. The hydrodynamics properties of underwater environments i.e., density, viscosity and stream velocity are simulated to generate external static and dynamic forces which are then applied to the modelled AUV's geometrical structure. In addition, the characteristics of underwater acoustic communications including communication channel, modems, [Medium Access Control \(MAC\)](#) protocol and time synchronisation are all simulated.
- An underwater localisation method to dynamically fuse multiple localisation estimates of an AUV using fuzzy decision support system [26, 27]. A number of underwater localisation methods have been presented in the literature for wireless sensor networks. The proposed navigation framework harnesses established localisation methods to provide navigation aids in the absence of the acoustic exteroceptive sensors navigation aid (i.e., [Ultra-short Baseline \(USBL\)](#)) and can be easily extended to accommodate some other localisation methods by expanding

the fuzzy rule base. Simulation studies have shown that the proposed fuzzy-based localisation outperforms the [EKF](#)-based localisation in localisation accuracy, scalability and design simplicity.

- A confidence-based distributed cooperative localisation algorithm for a swarm of mobile underwater sensor nodes [\[28\]](#). The localisation accuracy is improved by promoting high-precision localised ordinary nodes to reference nodes based on their confidence values. The confidence value of a node is dynamically updated by the proposed confidence update rules.
- Optimisation of the proposed confidence based algorithm's parameters. The algorithm's parameters are optimised so that intra-swarm collaboration is maximised given the constraints imposed by the underwater environments and swarm systems [\[29\]](#). Simulation studies have shown that parameter optimisation improves the localisation accuracy specially at large swarm sizes i.e., 100-150 [AUVs](#).

1.6 Thesis Structure

The remainder of the thesis is structured in seven chapters as follows:

- Chapter [2](#) provides a comprehensive review of seismic imaging and marine seismic in terms of seismic sensors and seismic geographic data. The current available robotic systems in marine seismic and the autonomous deployment of ocean bottom nodes are discussed from the feasibility and associated challenges perspectives.
- Chapter [3](#) surveys related underwater localisation and communication algorithms in the literature and provides background information that is necessary for the remainder of the thesis.
- Chapter [4](#) illustrates the implemented co-simulation platform elements including the high-fidelity physics-based robotic simulator, the Underwater Network Project for acoustic simulation (i.e., [UnetStack](#)), the simulated [AUV](#) and the adopted [IMU](#) model.
- Chapter [5](#) presents a fuzzy-based localisation framework. A simple proof-of-concept simulation and high fidelity physics-based simulation based on the developed co-simulation platform presented in Chapter [4](#) have been conducted to

validate the proposed fuzzy-based algorithm. The proposed fuzzy-based localisation framework is compared to other traditional and commonly adopted underwater localisation methods such as the Extended Kalman Filter in terms of scalability and localisation accuracy under the same simulation settings.

- Chapter 6 proposes an optimised confidence-based localisation algorithm for improving localisation accuracy by promoting nodes with high confidence of location estimates to references for their neighboring nodes. The confidence value is updated based on the adopted localisation methods' error characteristics where expected localisation error is generated based on redundant sensory information to control the cooperation dynamics in the swarm.
- Chapter 7 improves the confidence-based cooperative localisation scheme for underwater robotic swarm. The confidence value update method is derived intuitively to represent the localisation precision in a single scalar measurement. The algorithm's key parameters are optimised as in Chapter 6 and the algorithm's performance is evaluated on the developed co-simulation platform presented in Chapter 4.
- Chapter 8 concludes the thesis and summarises the findings of this work. Future directions and recommendations are discussed to highlight possible further considerations to potentially improve the proposed underwater swarm navigation algorithm.

Chapter 2

Seismic Imaging Literature

Ocean bottom seismic surveys are vital for oil and gas exploration and productivity enhancement of an existent production facility. Ocean bottom seismic nodes are deployed on the seabed to acquire those surveys. Nodes deployment methods used in industry today are costly, time-consuming and inapplicable in deep oceans. Autonomous deployment of ocean bottom seismic nodes implemented by a swarm of [AUVs](#) is considered.

2.1 Introduction

The most dominant source of energy (i.e., oil and gas) is hidden beneath land and sea surfaces. Some of oil and gas reservoirs are at shallow depth and other at ultra-deep layers in which drilling cost sharply increases. High resolution marine seismic is crucial technology in reducing oil exploration and drilling cost. Offshore oil/gas reserves have not been fully explored.

Global energy demand is expected to rise to 470 million barrels of oil per day in 2060 while it was 100.1 million barrels per day in 2019 [30, 31], thereby ultra-deep layers must be explored at reasonable cost. In the last decade, renewable energy has greatly improved; however, it still cannot match the global demand of energy. The Gulf of Mexico and the Pre-salt area off Brazil may contain 100,000 million barrels of crude oil and due to the ultra-deep water, the drilling cost in this area is between 175 and 200 million US dollars [32]. Due to the high cost of drilling, oil industries have been looking forward to acquiring subsurface images of better resolution. Thus, better estimation of hydrocarbon reserve before drilling is achieved.

High resolution seismic imaging of substructure has always been an ultimate goal in oil industry, whether to be used in the exploration phase or to enhance the productivity of an existence oil platform. Autonomous deployment of seismic imaging devices can further reduce oil production cost and personnel hazards, especially in remote and harsh environment. **Ocean Bottom Node (OBN)** seismic has attracted a lot of interest when it comes to high resolution marine seismic imaging. **OBNs** are simply seismic sensors with the capability of providing relatively high resolution seismic images of sub-seabed [4]. Recent developments in **OBN** have risen the need for reliable and cost-effective deployment methods [4]. **OBNs** are currently deployed using a **ROV** equipped with a robotic arm and driven by on board crew. The main disadvantage of the current deployment method is the associated high cost due to the long time needed. Moreover, the current deployment method becomes tedious and not cost effective when the number of **OBN** is large i.e., in hundreds. Therefore, there is a need for an autonomous deployment system of **OBNs** to overcome the limitations of the current **OBNs** deployment method. The key challenge underlying the autonomous deployment of **OBNs** is underwater sensor nodes localisation which has been an active research area in recent years given that the **GNSS** is absent in underwater environments. Radio signals can propagate underwater only for very short distances at low frequencies i.e., a few meters at 10 kHz [33]. Hence underwater wireless communication relies predominately on acoustic waves to propagate for long distances i.e., up to 10 km [34]. The rest of this chapter is organised as follows. A generic review of seismic imaging is given in section 2.2 and marine seismic in particular is explained in terms of seismic sensors and seismic data in section 2.3. The current available robotic systems in marine seismic are then discussed in section 2.4. Finally, this chapter is concluded in section 2.5.

2.2 Seismic Imaging

Bats cannot see very well; however, they still can fly in cluttered and unstructured environments. Doctors can identify embryos gender and monitor their health status in the fourteenth week of human pregnancy without any operation. Bat sends sound waves that bounce off an object and its ears receive the reflected sound waves. Doctors use the same principle i.e., sonar to identify embryos gender using transabdominal fetal ultrasound image. Geologists apply the same principles to look into deep earth layers and thus, determine whether or not there is an oil and gas deposit in a particular geographical area [35]. A vibrator is employed on the earth surface to send sound waves

into the subsurface layers. Sensors that receive the reflected waves are deployed on land in land seismic or on sea surface in marine seismic. Acoustic images are generated based on the recorded data. The generated sound image looks like a bunch of squiggles [36]. To generate a picture that actually looks like the earth beneath us, the data has to be processed by a series of sophisticated signal processing and model reconstruction algorithms on high performance computing facilities [36].

2.2.1 Seismic Waves

There are two different types of sound waves involved in seismic imaging [5]. The first type is compressional waves (i.e., P-waves, where P stands for Primary) they propagate through solid or liquid mediums such as water and rocks with propagation speed depending on medium's properties [37]. The second type is shear waves (i.e., S-waves, where S stands for Secondary) which propagate through solid mediums only such as rocks [37]. Sound waves are generated by vibrator (i.e., sound-waves source on the surface) it can propagate and reflect through rock layers as P-waves and it can be converted to S-waves depends on the medium. P-waves contain the most important information that geologists need to reveal possible oil and gas deposits. However, S-waves, in some cases, must be recorded and analysed as they contain information cannot be extracted from P-waves. In general, S-waves have higher frequency components and travel slower than P-waves and they reveal valuable information about rocks physical properties for many fields of research including petrophysics, geology, geophysics, geochemistry, geotechnical engineering and materials science [37].

2.2.2 Seismic Sensors

Seismic sensors can be deployed on land surface, sea surface or sea floor in different deployment geometries. P-waves are acquired by hydrophones which are essentially pressure sensors; it measures the amplitude of P-waves [38]. On the other hand, S-waves are acquired by geophones which are essentially accelerometers [38]. Geophones detect the ground motion in a single axis. For accurate recording, three geophones can be used to detect ground motion in three orthogonal axes (North, East, and Down) and it is called three component geophone [5]. In onshore seismic, geophones are deployed on land surface ready to record reflected S-waves [5]. Whereas in offshore seismic, only hydrophones are deployed on the sea surface ready to record reflected P-waves [5].

Moreover, hydrophones and geophones are used in offshore seismic when sensors are deployed on the seabed. Usually each seabed node contains either a hydrophone and a geophone, called 2-component node, or a hydrophone and 3-component geophone, called 4-component node.

2.3 Marine Seismic

In this section marine seismic methods in term of sensors deployment geometries and data dimensionality are reviewed.

Offshore oil and gas reservoirs are hidden underneath the seabed. Offshore oil production represents more than thirty percent of world's oil production; hence there is an increasing demand for marine seismic surveys for oil and gas reservoirs characterisation and localisation. Marine seismic methods can be differentiated in terms of sensors types, sensors deployment geometries and sensors densities.

2.3.1 Sensor Deployment Geometry

Towed streamers, Ocean bottom and vertical seismic profiling are common marine seismic methods differed in sensors deployment geometry. One or many streamers are towed behind the survey vessel. A single vessel could tow from one to sixteen streamers in a length of 3 to 12 km [5]. Streamers and a vibrator (i.e., sound waves source) are usually towed not more than few metres apart from each other on the sea surface by the same vessel. Hydrophones are located in an isolated rubber tube one metre apart and every 12 to 25 metres length of tube are electrically connected [5]. To allow easy maintenance, streamers are divided into sections (modular) every 25 to 50 metres [5]. Simple mechanism of depth control birds are placed every 300 metres to keep the streamer within 4 to 10 metres depth [5]. Streamer's tubes are filled with extruded foam so it can help in streamers buoyancy and reduce wave propagation inside the streamer [5]. Towed streamers are usually used to explore open and large areas. Although this method provides adequate geological information, it is not enough for geoscientists to make decisions on drilling. Figure 2.1 shows a simple diagram of a towed streamer for seismic imaging.

Receivers can be deployed at designated locations on seabed (i.e., ocean bottom geometry) and each receiver consists of a hydrophone and a 3-component geophone to acquire higher resolution seismic images in which any hidden hazards such as gas pockets or

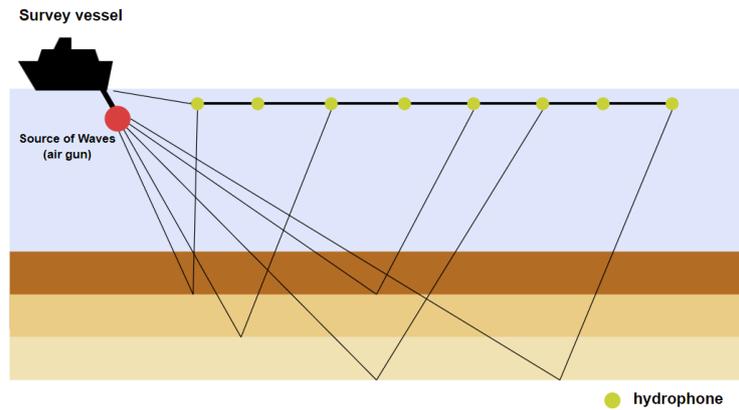


Figure 2.1: A streamer of hydrophone towed behind a vessel equipped with a source of waves to capture the reflected signals of different sub-seabed layers.

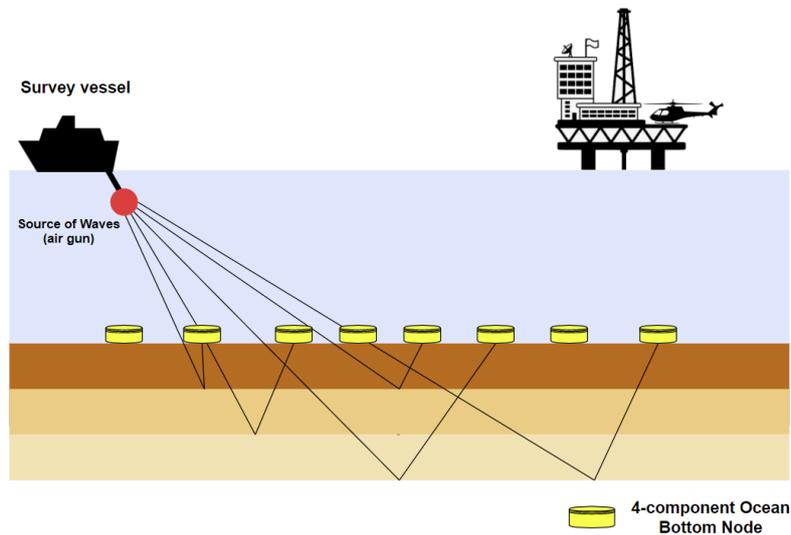


Figure 2.2: Ocean bottom nodes are deployed in the vicinity of oil and gas platform to record the reflected signals of the different sub-seabed layers.

buried river channels can be detected. Seabed deployment is mostly adopted in the vicinity of oil and gas platform or any sub-sea constructions. Traditional streamers are almost impossible to be towed in the vicinity of any offshore constructions. The vessel can be 100 metres long and 30 metres wide plus 160 km length of sixteen streamers. Figure 2.2 shows an example of OBN deployment geometry. Multiple source vessels can be used simultaneously in one survey if OBNs are deployed on large areas to shorten the survey duration [5]. Vertical deployment geometry, called vertical seismic profiling,

is applied while drilling in which seismic sensors are deployed in a well hole. The main inherent advantage of this method is that high sampling rate can be achieved in recording as the receivers are located in the well hole so that reflected waves do not need to travel for long distances to be captured.

2.3.2 Data Dimensionality

There are three different types of the acquired seismic data. Two dimensional data 2D, in which receivers are deployed in a line. The acquired data is the cross-sectional area of the sub-seabed along the vessel sail line. A single streamer is towed behind a vessel is an example of 2D seismic imaging acquisition. Three dimensional data 3D, in which the sensors are deployed in a grid. Three dimensional model of sub-seabed can be constructed by the multiple acquired cross-sectional areas. For example, several streamers are towed behind a vessel or a mesh-like deployment of OBNs. Three-dimensional seismic currently represents ninety percent of seismic surveys. When 3D seismic survey is being repeated over time across the same area or patch, the acquired data then has a fourth dimension 4D. The extracted data can be visualised as spatial-temporal volume of crude reservoirs. Figure 2.3 presents an example of 2D and 3D seismic images available from [36] and [39].

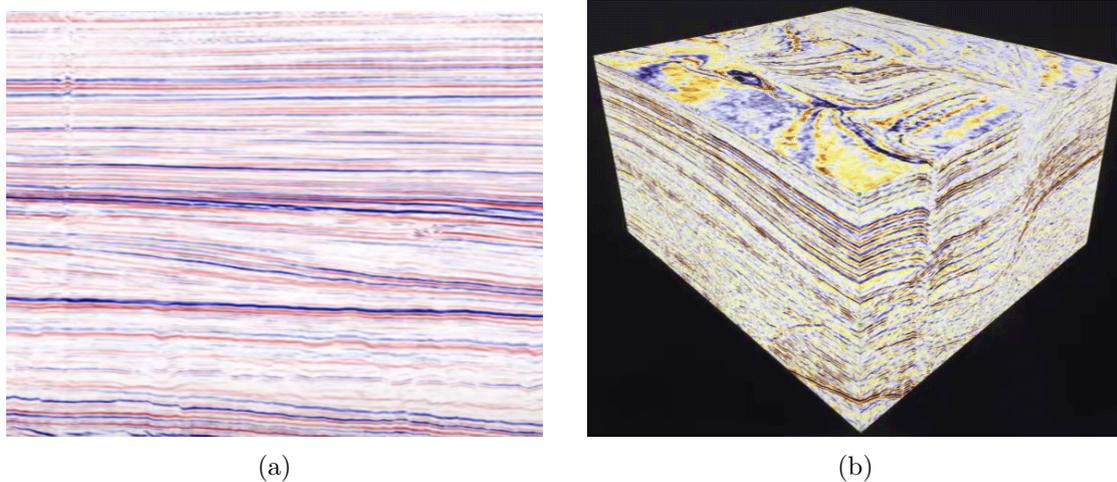


Figure 2.3: An example of (a) a 2D seismic image acquired by deploying a single streamer on the sea surface [36] and (b) a 3D seismic image acquired by deploying multiple streamers to form a grid of hydrophones on the sea surface [39].

2.4 Robotic Systems in Marine Seismic

Performance, safety, efficiency and cost-effectiveness can be greatly improved when robotic systems are applied. Robotic systems are widely applied in various industrial applications. Researchers in seismic acquisition are always keen to mitigate traditional seismic methods limitations such as cost and intensive labour by implementing efficient and cost-effective automated systems.

2.4.1 Autonomous Towed Streamers

The Widely scalable Mobile Underwater Sonar Technology (WiMUST) is a project funded by the European Commission aimed at developing a swarm of AUVs for geophysical and geotechnical surveys such as marine seismic imaging [40]. The main objective of WiMUST project was to convert the traditional marine seismic surveys acquired by towed streamers into a distributed, manageable and re-configurable system where the sound source (i.e., vibrator) and the streamers are decoupled [40]. Each streamer in WiMUST is towed by an AUV and the source is towed by the operating vessel as shown in Figure 2.4. The project has been concluded in 2018; an overview of the navigation, control and guidance of WiMUST project is provided in [41].

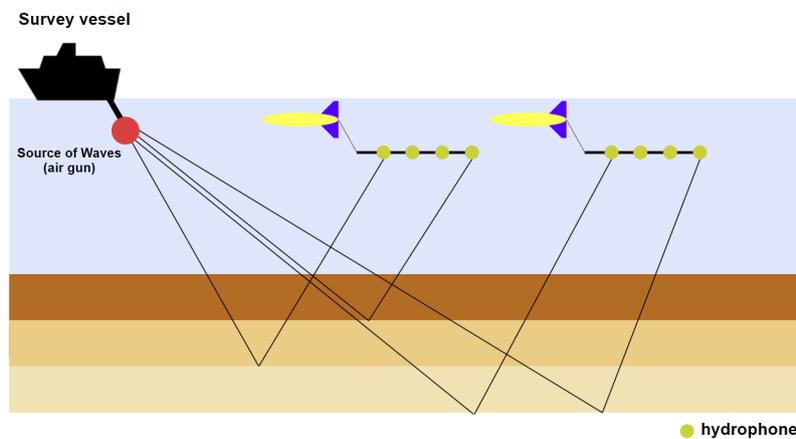


Figure 2.4: Streamers of hydrophones are separated from the source vessel and towed by multiple AUVs; WiMUST project's concept.

Wave glider is an autonomous marine vehicle which has been introduced by Liquid robotics [42]. Wave glider vehicle is powered by ocean waves and it is comprised of a

float part and a submerged part, connected by an umbilical tether as shown in Figure 2.5 [43]. In 2014, Moldoveanu et al. carried out a field experiment in which they used

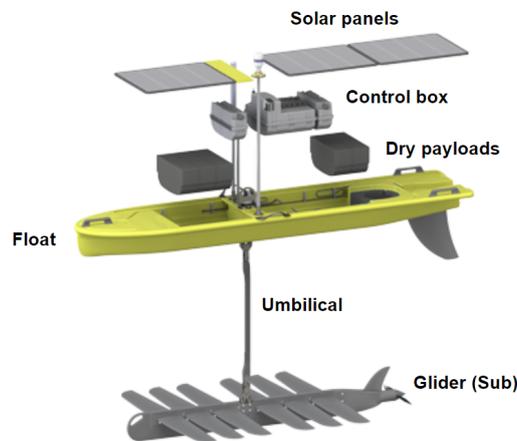


Figure 2.5: Wave glider vehicle schematics. Image courtesy of Liquid Robotics [43].

a wave glider to tow a single streamer of 31 metres length in a depth of 6 to 7 metres below the sea surface over a shallow water area [44]. This field experiment was carried out over the same area at the same time using **OBNs**. By comparing the recorded reflected waves by both the towed streamer and **OBNs**, no significant difference was found between them. However, towed streamer recorded data must be processed in advance for noise removal. An **OBN** is comprised of a hydrophone and a 3-component geophone and a towed streamer contains only hydrophones. In this field experiment, only hydrophone data was compared. Another similar field experiment has been carried out in 2016 by Moldoveanu et al. in which a 3D seismic sensor array is towed instead of a simple streamer [45]. The 3D seismic sensor array is comprised of 15 hydrophones, a buoyancy engine and orientation sensors. They observed only slight differences between the 3D seismic sensor array recorded data and the **OBNs** recorded data. Frequency components of the **OBNs** recorded data was slightly less than those in 3D seismic sensor array. However, the signal to noise ratio in 3D seismic sensor array recorded results was higher than those in a single streamer.

2.4.2 Autonomous Ocean Bottom Nodes Deployment

OBNs are deployed on the seabed by either a submarine or an **ROV**. Trilobit is a 4-component **OBN** that was introduced by the geophysical services company CGG, it is

composed of a hydrophone, a 3-component geophone, its own recording unit and its own battery so that it can survive for up to 90 days on seabed [46]. Z3000 is another OBN has the same features of Trilobit but with longer battery life [47]. MANTA is the latest OBN launched by Seabed GeoSolution [48]. Likewise, MANTA has the same general features of Z3000 and Trilobit but with a lighter weight and a smaller size. Those OBNs are deployed on the seabed by ROVs or cables that tie them together and keep them separated in mesh-like geometry i.e., cabled nodes. Usually they are being deployed in a grid with 200-400 metres apart from each other [5]. The positional relative accuracy of the ROV deployment is 5-10 metres in 2 kilometres water depth (independent of the GNSS accuracy) [49]. Figure 2.6 shows a diagram of OBNs being deployed by ROVs. To speed up the deployment process a cage loaded with OBNs is first deployed on the seabed from which the ROVs are deploying the OBNs at some designated seabed positions.

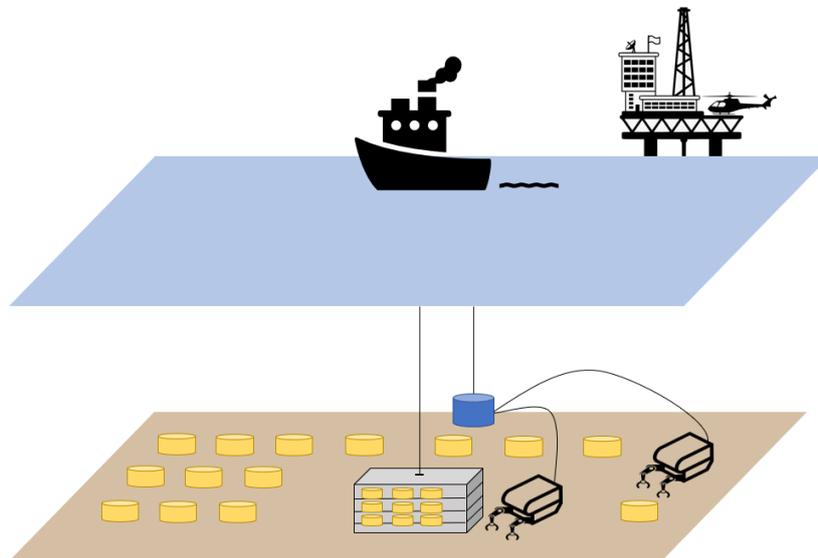


Figure 2.6: Ocean bottom nodes represented by yellow cylinders are deployed in a grid on the seabed by remotely operated vehicles in the vicinity of oil and gas platform.

OBNs deployment by ROVs is very costly as it requires the existence of ROV operating vessel with its crews on board. Nevertheless, the cost can sharply increase when it comes to deep water and wide coverage deployment.

Autonomous Robotics Limited is currently working on a project called Flying Nodes, the next generation of ocean bottom sensing acquisition technology for which they are manufacturing a swarm of small size AUVs for autonomous deployment of OBNs [24]. Their aim is to convert each OBN to an AUV so there will be no need for an ROV to

deploy them. Each AUV in the swarm will be deployed from around 20 metres depth of sea surface and descend to their pre-programmed seabed positions with at least the same accuracy of ROV deployment i.e., 5-10 metres in 2 kilometres water depth. Similar to the Flying Nodes, another project called SpiceRack by CGG [6] and Saudi Aramco [50], on the other hand, focuses on shallow water deployment. Both SpiceRack and Flying Nodes are still under development. Figure 2.7 illustrates the idea of utilising a swarm of AUVs for autonomous ocean bottom seismic imaging where each AUV is equipped with a 4-component OBN. Holloway et al. have presented an interesting feasibility study in

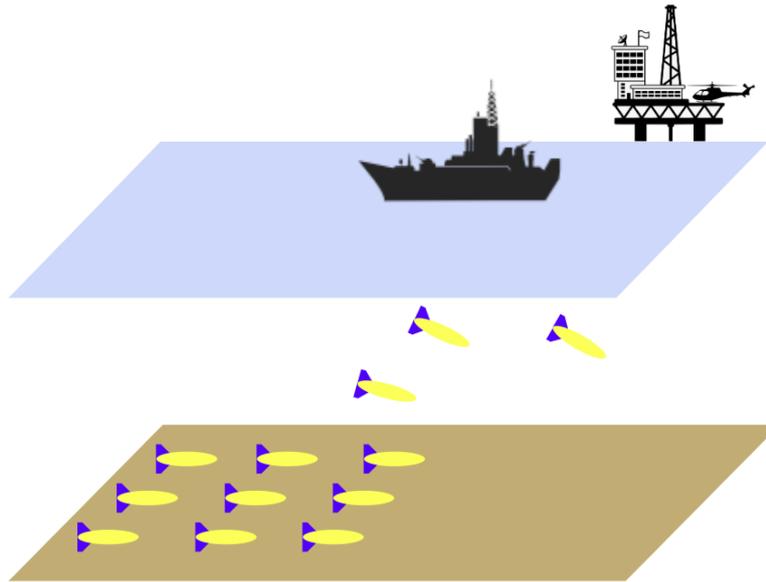


Figure 2.7: A swarm of AUVs utilised for autonomous ocean bottom seismic imaging. Each AUV is equipped with an ocean bottom seismic node.

[51] where they compared the total cost and time needed to deploy 2501 OBNs with 200 metres spacing by cables, ROVs or autonomous nodes i.e., Flying Nodes. They found that an ocean bottom seismic survey can be completed in 90 days if ROV deployment is adopted regardless the number of source vessels. However, when autonomous deployment such as Flying Nodes or SpiceRack is adopted, the time needed to complete an ocean bottom seismic survey significantly decreases (i.e., from 53 to 31 days) when one more source vessel is used as shown in Figure 2.8 [51]. Autonomous deployment method of OBNs such as Flying Nodes would always shorten the survey duration and reduce the total cost regardless the number of source vessels as shown in Figure 2.8 and regardless nodes separation as shown in Figure 2.9 [51].

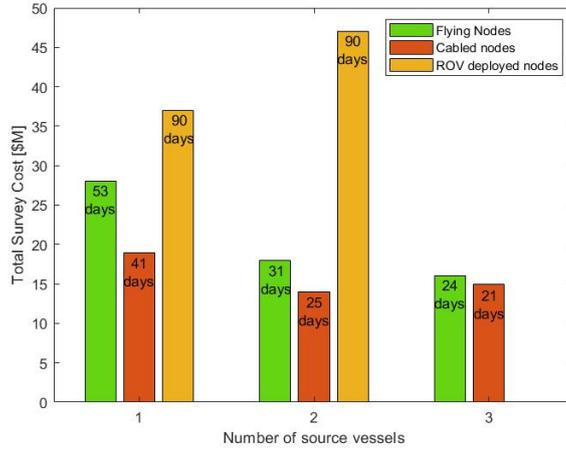


Figure 2.8: The number of source vessels effect on total survey cost and survey duration in ocean bottom seismic imaging by three different deployment methods [51].

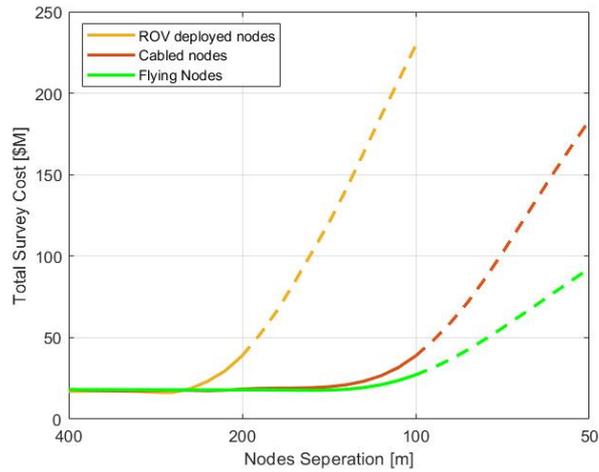


Figure 2.9: Ocean bottom nodes separation effect on total survey cost in three different deployment methods - data depicted by the dashed lines are projected [51].

However, there are several challenges associated with autonomous deployment of OBNs for seismic imaging, it is expected that such system will have a significant share of total seismic surveys when it becomes commercially available as total survey cost and duration are considerably reduced. Challenges associated with underwater swarm systems including underwater communications, AUV's control, formation control, stabilisation, path planning, efficient energy consumption and underwater localisation. Underwater swarm localisation is one of the most enabling technology for autonomous deployment of

OBNs. Many challenges such as **AUV**'s control, formation control, **AUV**'s stabilisation and path planning required the **AUVs** to be localised first.

2.5 Conclusion

Ocean bottom seismic imaging acquired by **OBNs** can greatly improve seismic data quality to either make a better informed decision on drilling or enhance the productivity of an existing oil field. **OBNs** are 4-component seismic sensors, each is comprised of a hydrophone, a 3-component geophone, a battery and a recording unit. **OBN** are currently predominately deployed by **ROVs**. It is time consuming, very costly and the number of **OBNs** that can be deployed by **ROVs** is very limited. A swarm of **AUVs** where each **AUV** is equipped with a 4-component seismic sensor has been proposed to reduce the cost and the duration of the current deployment methods of **OBNs**.

There are various technical challenges for autonomous deployment of **OBNs** such as underwater swarm localisation, **AUV**'s control, formation control, stabilisation and path planning. Underwater swarm localisation importance is emphasised in autonomous deployment of **OBNs** systems as the location of each individual swarm's node (i.e., **AUV** equipped with ocean bottom seismic sensor) is required to be known and tracked all the time during operation. A collaborative research has been established with Autonomous Robotics Ltd as part of this research to mitigate underwater swarm localisation challenges for the Flying Nodes project.

Chapter 3

Underwater Localisation Literature

3.1 Introduction

The wide variety of sub-sea applications such as deep sea exploration, environmental monitoring, geological and ecological research, and samples collection have enabled and motivated [Underwater Wireless Sensor Network \(UWSN\)](#) localisation researchers for decades. In an [UWSN](#) application, a swarm of sensor nodes are deployed to communicate and collaboratively achieve various predefined tasks in underwater environments such as ocean bottom seismic imaging. In order to successfully complete assigned missions, locations of individual sensor nodes must be known and tracked during operation for location aware applications. Individual sensor node's location must also be known and tracked in underwater swarms application for guidance and control in autonomous deployment of sensor nodes.

Usually, an [UWSN](#) consists of a few nodes with known positions (i.e., reference nodes) and a large number of nodes with unknown positions i.e., unknown nodes. Underwater environments inherently introduce many constraints including limited communications bandwidth, long communications latency in case of acoustic communications and nodes drifting. Therefore, unknown nodes usually rely on reference nodes for localisation updates. Each node in an [UWSN](#) could be static or mobile and some of the nodes maybe situated on the sea surface for [GNSS](#) information and some could be located sub-sea surface. An unknown node interrogates the network by broadcasting a signal (e.g., acoustic

signal) when it needs to be localised. The localisation is then performed given the reference nodes' positions and the distance between the unknown node and reference nodes through e.g., the [Time of Arrival \(ToA\)](#) or [Time Difference of Arrival \(TDoA\)](#) methods [52]. Reference nodes can obtain their positions by i) the [GNSS](#) if they are situated on the sea surface ii) keeping them anchored at certain known positions i.e., anchor nodes iii) external [ToF](#) acoustic navigation aid.

The localisation of each individual sensor node in a mobile [UWSN](#) (i.e., [AUVs](#) swarm) relies on its proprioceptive sensors (i.e., [IMU](#)) with high update rate and exteroceptive sensors with low update rate. External navigation aids usually rely on reference locations and range measurements that can be achieved by means of [ToF](#) methods which require robust underwater communications. The importance of the underwater communications (i.e., physical layer and [MAC](#) layer) is further emphasised in cooperative localisation where each localised sensor node may broadcast (depends on some predetermined criteria) localisation aids to their neighbouring nodes.

The remainder of this chapter is organised as follows. Section 3.2 explains the backbone system of underwater navigation (i.e., [Inertial Navigation System \(INS\)](#)) and the [EKF](#) for pose estimation and sensor fusion. Section 3.3 reviews selected underwater localisation algorithms and introduces the readers to the trilateration problem and the most common methods of solving it in literature. A brief review of underwater communications is given in section 3.4. Finally, this chapter is concluded in section 3.5.

3.2 Inertial Navigation System

Mobile sensor nodes (i.e., [AUVs](#)) rely predominately on their proprioceptive sensors such as [IMU](#) for localisation. Although [IMU](#)-based localisation (i.e., [INS](#)) is prone to drift, the recent development in [Microelectromechanical systems \(MEMS\)](#) technology has enabled the [IMU](#) sensors development. A three-axis gyroscope, three-axis accelerometer and three-axis compass are often encased in a single chip and referred to as nine-axis [IMU](#). Some the industrial-grade electronic navigation boards consist of a 9-axis [IMU](#) and optionally run Kalman filter to estimate Roll, Pitch and Yaw angles as well as to fuse external navigation aids i.e., from the [GNSS](#) [53]. The navigation system that fuses the measurements of the gyroscope, accelerometer and magnetometer to estimate the vehicle's orientation is called [Attitude Heading Reference System \(AHRS\)](#) whereas the [INS](#) estimates the vehicle's position by double integrating the accelerometer's measurements given the initial position of the vehicle. The process of determining the vehicle's

position, velocity and attitude from the **IMU**'s raw measurements through solving the system motion's differential equations is often referred to as **INS** mechanisation or strap-down mechanisation. Strapdown relates to the way the inertial sensors attached to the vehicle's body as they can be either directly strapped to the structure of the vehicle or gimbaled [54]. Gimbaled **IMU** is mechanically complicated and not as common as strapdown ones. We only consider a strapdown **INS** in our application for mobile sensor nodes (i.e., **AUVs**) navigation.

The following frames definitions and nomenclature are used in the inertial measurements:

- The body frame of reference **b** is the **IMU** sensor frame that all inertial measurements are resolved with respect to. Its origin is located at the centre of the accelerometer triad and aligned with the casing.
- The inertial frame of reference **i** is the frame in which the **IMU**'s measurements are returned with respect to. Its origin is located at the centre of the earth and its axes are aligned with the stars.
- The earth frame of reference **e** rotates with earth and its origin is located at the centre of the earth.
- The navigation frame of reference **n** is the frame in which we are interested to localise the **AUV**. It is often considered stationary with respect to the earth unless the moving body travels for long distances (i.e., comparable to the size of the earth) then it should be rotated with the earth. It will always considered stationary with respect to the earth in our application for **AUV** navigation.

A vector's frame of reference is indicated by a superscript e.g vector A is written as bA if it is in **b** frame of reference and the rotation matrix that rotates bA from **b** to **n** frame of reference is ${}^n\mathcal{R}_b$ as ${}^nA = {}^n\mathcal{R}_b {}^bA$.

The gyroscope returns the **AUV**'s angular velocity ${}^i\omega_{b,t}$ at time t in the inertial frame of reference **i** corrupted by a time-varying bias \mathcal{B}_t and noise $e_{\omega,t}$ and therefore the row measurements model of the gyroscope $y_{\omega,t}$ is

$$y_{\omega,t} = {}^i\omega_{b,t} + \mathcal{B}_t + e_{\omega,t} \quad (3.1)$$

where the gyroscope error $e_{\omega,t}$ is drawn from a Gaussian distribution $\mathcal{N}(0, \Sigma_\omega) : \Sigma_\omega \in$

$\mathbb{R}^{3 \times 3}$ and the bias \mathcal{B}_t can be modelled as a random walk by

$$\mathcal{B}_{t+1} = \mathcal{B}_t + W_{\mathcal{B},t} \quad (3.2)$$

where $W_{\mathcal{B},t} \sim \mathcal{N}(0, \Sigma_{\mathcal{B}})$ that represents how random the bias is.

Likewise, the accelerometer measures the force in the body frame of reference ${}^{\mathbf{b}}f_t$ at each time step t and its readings are corrupted by a time-varying bias \mathcal{A}_t that can be modelled as a random walk similar to Equation 3.2 and zero mean Gaussian noise $e_{a,t}$; the accelerometer measurements $y_{a,t}$ are then modelled by

$$y_{a,t} = {}^{\mathbf{b}}f_t + \mathcal{A}_t + e_{a,t} \quad (3.3)$$

$${}^{\mathbf{b}}f_t = {}^{\mathbf{b}}\mathcal{R}_{\mathbf{n},t}({}^{\mathbf{n}}a_t - {}^{\mathbf{n}}g_t) \quad (3.4)$$

where ${}^{\mathbf{b}}\mathcal{R}_{\mathbf{n},t}$ rotates the resultant vector of $({}^{\mathbf{n}}a_t - {}^{\mathbf{n}}g_t)$ from the navigation \mathbf{n} to the body frame \mathbf{b} . The accelerometer measurements are dominated by the gravity vector ${}^{\mathbf{n}}g_t$ and therefore the linear acceleration ${}^{\mathbf{n}}a_t$ can be neglected

$$y_{a,t} = -{}^{\mathbf{b}}\mathcal{R}_{\mathbf{n},t} {}^{\mathbf{n}}g_t + \mathcal{A}_t + e_{a,t} \quad (3.5)$$

$$\mathcal{A}_{t+1} = \mathcal{A}_t + W_{\mathcal{A},t} \quad (3.6)$$

where $W_{\mathcal{A},t} \sim \mathcal{N}(0, \Sigma_{\mathcal{A}})$ that represents how random the bias is.

On the other hand, the magnetometer complements the accelerometer to find the AUV's heading around the gravity vector i.e., yaw angle. The local magnetic field flux density of the earth ${}^{\mathbf{n}}m$ has a horizontal and a vertical components, the ratio between them depends on the *dip angle* [55]. The dip angle δ changes with the experiment's location on the earth e.g. at the earth's magnetic north pole the dip angle is equal to 90° and therefore the local magnetic field ${}^{\mathbf{n}}m$ would only have a vertical component [55]. The local magnetic field of the earth ${}^{\mathbf{n}}m$ can be expressed in terms of the dip angle as

$${}^{\mathbf{n}}m = (\cos \delta \ 0 \ \sin \delta)^\top : \|{}^{\mathbf{n}}m\| = 1 \quad (3.7)$$

Given the local magnetic field of the earth ${}^{\mathbf{n}}m$, the magnetometer measurements $y_{m,t}$ are modelled by

$$y_{m,t} = {}^{\mathbf{b}}\mathcal{R}_{\mathbf{n},t} {}^{\mathbf{n}}m + e_{m,t} \quad (3.8)$$

where $e_{m,t} \sim \mathcal{N}(0, \Sigma_m) : \Sigma_m \in \mathbb{R}^{3 \times 3}$ represents the magnetometer measurement noise.

The local magnetic field of the earth can be accurately determined from geophysical studies [56] as it solely depends on the experiment's location. The integration of the gyroscope measurements estimates the sensor's orientation and the double-integration of the accelerometer measurements estimates the sensor's position after subtracting the earth's gravity. The integration process and the noisy measurements of the IMU result in *integration drift* and therefore external navigation aids must be fused. The problem of integration drift is even exacerbated when a low cost IMU is utilised.

Vehicle's position and orientation can be estimated based on the IMU measurements and an external navigation aid in many different ways. For example, position and orientation estimation can be interpreted as an optimisation or filtering problem [55]. The use of Gauss-Newton optimisation in this framework is detailed in [55]. However, Kalman filtering is the mostly adopted method in literature for fusing IMU measurements such as magnetometer with gyroscope measurements to estimate the geomagnetic vector as well as IMU measurements with an external navigation aid e.g. GNSS or USBL measurements in underwater navigation. A considerable amount of literature has been published on Kalman filter based externally aided Dead Reckoning (DR) navigation; see for example [57, 55, 58, 59].

3.2.1 Pose Estimation

In pose estimation, the state vector includes the position ${}^{\mathbf{n}}p_t$, the velocity ${}^{\mathbf{n}}v_t$ and the orientation ${}^{\mathbf{n}}q_{\mathbf{b},t}$ of the vehicle. The position and velocity dynamics are given by

$${}^{\mathbf{n}}p_{t+1} = {}^{\mathbf{n}}p_t + \Delta T {}^{\mathbf{n}}v_t + \frac{\Delta T^2}{2} ({}^{\mathbf{n}}\mathcal{R}_{\mathbf{b},t}(y_{a,t} - \mathcal{A}_t) + {}^{\mathbf{n}}g + e_{a,t}) \quad (3.9)$$

$${}^{\mathbf{n}}v_{t+1} = {}^{\mathbf{n}}v_t + \Delta T ({}^{\mathbf{n}}\mathcal{R}_{\mathbf{b},t}(y_{a,t} - \mathcal{A}_t) + {}^{\mathbf{n}}g + e_{a,t}) \quad (3.10)$$

where ΔT is the time step. The orientation ${}^{\mathbf{n}}q_{\mathbf{b},t}$ is encoded in term of linearisation point as unit quaternion ${}^{\mathbf{n}}\tilde{q}_{\mathbf{b},t}$ and in term of orientation deviation as a rotation vector η_t ; given by

$${}^{\mathbf{n}}q_{\mathbf{b},t} = \exp\left(\frac{\bar{\eta}_t}{2}\right) \odot {}^{\mathbf{n}}\tilde{q}_{\mathbf{b},t} \quad (3.11)$$

where the quaternion multiplication is denoted by \odot , η_t is expressed in the navigation frame, $\bar{\eta}_t = (0 \quad \eta_t^\top)^\top$ and $\exp(\bar{\eta})$ is given by

$$\exp(\bar{\eta}) = \begin{pmatrix} \cos \|\eta\|_2 \\ \frac{\eta}{\|\eta\|_2} \sin \|\eta\|_2 \end{pmatrix} \quad (3.12)$$

where $\|\eta\|_2$ denotes the L^2 -norm of vector η . The justification of this quaternion encoding in term of the linearisation point and the orientation deviation is detailed in [60] and [61] and used in the navigation framework in [55]. The quaternion dynamics of the orientation is given by

$${}^{\mathbf{n}}q_{\mathbf{b},t+1} = {}^{\mathbf{n}}q_{\mathbf{b},t} \odot \exp_q\left(\frac{\Delta T}{2}(y_{\omega,t} - \mathcal{B}_t - e_{\omega,t})\right) \quad (3.13)$$

where $\exp_q(\eta) = \exp(\bar{\eta})$. The state space model of vector x_t for pose estimation is then given by

$$\begin{pmatrix} {}^{\mathbf{n}}p_{t+1} \\ {}^{\mathbf{n}}v_{t+1} \\ {}^{\mathbf{n}}q_{\mathbf{b},t+1} \end{pmatrix} = \begin{pmatrix} {}^{\mathbf{n}}p_t + \Delta T {}^{\mathbf{n}}v_t + \frac{\Delta T^2}{2} ({}^{\mathbf{n}}\mathcal{R}_{\mathbf{b},t}(y_{a,t} - \mathcal{A}_t) + {}^{\mathbf{n}}g + e_{p,a,t}) \\ {}^{\mathbf{n}}v_t + \Delta T ({}^{\mathbf{n}}\mathcal{R}_{\mathbf{b},t}(y_{a,t} - \mathcal{A}_t) + {}^{\mathbf{n}}g + e_{v,a,t}) \\ {}^{\mathbf{n}}q_{\mathbf{b},t} \odot \exp_q\left(\frac{\Delta T}{2}(y_{\omega,t} - \mathcal{B}_t - e_{\omega,t})\right) \end{pmatrix} \quad (3.14)$$

where $e_{p,a,t} \sim \mathcal{N}(0, \Sigma_a)$, $e_{v,a,t} \sim \mathcal{N}(0, \Sigma_a)$ and $e_{\omega,t} \sim \mathcal{N}(0, \Sigma_\omega)$ with $\Sigma_a = \sigma_a^2 \mathcal{I}_3$ and $\Sigma_\omega = \sigma_\omega^2 \mathcal{I}_3$: \mathcal{I}_3 is 3×3 identity matrix. In case of external navigation aid ($y_{p,t}$) (i.e., **GNSS** or **USBL** in underwater navigation) and magnetometer measurement ($y_{m,t}$) availability, the measurement model y_t can be given by

$$y_t = \begin{pmatrix} y_{p,t} \\ y_{m,t} \end{pmatrix} \quad (3.15)$$

with

$$y_{p,t} = {}^{\mathbf{n}}p_t + e_{p,t} \quad (3.16)$$

$$y_{m,t} = {}^{\mathbf{b}}\mathcal{R}_{\mathbf{n}} {}^{\mathbf{n}}m + e_{m,t} \quad (3.17)$$

where $e_{p,t} \sim \mathcal{N}(0, \Sigma_p)$ and $e_{m,t} \sim \mathcal{N}(0, \Sigma_m)$. Note that in 3.16 it is assumed that the measurements provided by the external navigation aid are pre-processed positioning data.

3.2.2 The Extended Kalman Filter (EKF)

The **EKF** is well proven and commonly known technique for fusion of **IMU** measurements and external navigation aids [58]. However, there are different architectures of the navigation filter design, namely direct integration, indirect feed-forward integration and indirect feedback integration [62]. Each approach of designing the navigation filter of the state estimator leads to different performance. In direct integration the set of

variables are directly related to the motion model e.g. position and orientation are the estimated variable in the [EKF](#). However, the variables' errors (i.e., position and orientation errors) are instead estimated in the indirect navigation filter and it is subtracted from either the motion model variables in the feed forward model or from the estimated motion model variables of the previous time step in the feedback model [\[62\]](#). Indirect navigation filter is the mostly adopted architectures in the literature and third party implemented libraries [\[58, 63, 64\]](#).

External navigation aids e.g., [GNSS](#) or [USBL](#) measurements can be integrated with the vehicle's [INS](#) estimated position and velocity in three different ways based on the external navigation aid data, namely loosely coupled, tightly coupled and ultra-tightly coupled [\[65\]](#). In loosely coupled way the [GNSS](#) data is pre-processed so that the sensor's position and velocity are integrated with the [INS](#) estimated position and velocity. In contrast, [GNSS](#) raw data (i.e., pseudoranges and pseudorange rates) are fused with the [INS](#) estimated position and velocity in the tightly coupled way. Likewise, raw [GNSS](#) data is fused in the ultra-tightly coupled way but with some differences in the architecture of the [GNSS](#) receivers.

External navigation aids in underwater navigation such as [USBL](#), [Long Baseline \(LBL\)](#) fusion with the [AUV's INS](#) estimated position and velocity is loosely coupled as the underwater external navigation aid means are normally integrated with their own filters to process the raw data (i.e., range measurements and [ToA](#)) to output the target's position and velocity [\[66, 67\]](#). In cases where the external navigation aid means (i.e., [USBL](#) or [LBL](#)) are custom-built, tightly coupled approach can be adopted [\[68\]](#) and [\[69\]](#). The [EKF](#) is used to find the best estimate of the sensor's position and orientation by the linearisation of the motion model i.e., [INS](#) model. The [EKF](#) linearises the nonlinear model and measurement model at some operating states and apply the original Kalman filter update rules and it can be interpreted as one iteration of Gauss-Newton optimisation [\[55\]](#) and hence it is computationally efficient. The [EKF](#) implementation is divided into time update or prediction equations and measurement update or correction equations [\[23\]](#). Assume that the measurement noise is additive, the process noise and measurement noise are zero-mean Gaussian with constant covariance. The nonlinear dynamic model and the nonlinear measurement model are given in [Equation 3.18](#) and [Equation 3.19](#) respectively,

$$x_{t+1} = f_t(x_t, u_t, w_t) \tag{3.18}$$

$$y_t = h_t(x_t) + e_t \tag{3.19}$$

where $w_t \sim \mathcal{N}(0, Q)$ and $e_t \sim \mathcal{N}(0, R)$: Q and R are error covariance matrices. The following shows the equations of the **EKF** to estimate the state vector \hat{x} by performing a time update using the model in 3.18 to predict the state of the next time step

$$\hat{x}_{t+1|t} = f_t(\hat{x}_{t|t}, u_t, w_t) \quad (3.20)$$

$$\Sigma_{t+1|t} = F_t \Sigma_{t|t} F_t^\top + G_t Q G_t^\top \quad (3.21)$$

with

$$F_t \approx \left. \frac{\partial f_t(x_t, u_t, w_t)}{\partial x_t} \right|_{\substack{w_t=0 \\ x_t=\hat{x}_{t|t}}} \quad (3.22)$$

$$G_t \approx \left. \frac{\partial f_t(x_t, u_t, w_t)}{\partial w_t} \right|_{\substack{w_t=0 \\ x_t=\hat{x}_{t|t}}} \quad (3.23)$$

The prior state estimate at time $t + 1$ given the measurements up to time t is denoted by $\hat{x}_{t+1|t}$ and its covariance matrix is denoted by $\Sigma_{t+1|t}$. Likewise, the posterior state estimate is denoted by $\hat{x}_{t|t}$ which is the state estimate at time t up to time t and its covariance matrix is denoted by $\Sigma_{t|t}$. The **EKF** performs a measurement update using the model in 3.19 to update the predicated state estimate as

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t(y_t - \hat{y}_{t|t-1}) \quad (3.24)$$

$$\Sigma_{t|t} = \Sigma_{t|t-1} - K_t S_t K_t^\top \quad (3.25)$$

where

$$S_t = H_t \Sigma_{t|t-1} H_t^\top + R, \quad K_t = \Sigma_{t|t-1} H_t^\top S_t^{-1} \quad (3.26)$$

and

$$\hat{y}_{t|t-1} = h(\hat{x}_{t|t-1}), \quad H_t \approx \left. \frac{\partial h_t(x_t)}{\partial x_t} \right|_{x_t=\hat{x}_{t|t-1}} \quad (3.27)$$

Time and measurement updates in the **EKF** are performed iteratively to estimate the state vector and its covariance matrix. The matrices F_t , G_t and H_t can be computed given the state space model in Equations 3.14 and 3.15 to perform the **EKF** for pose estimation. Interested readers are referred to [55] for the complete derivation of the **EKF** for pose estimation.

3.3 Underwater Localisation Algorithms

ToF acoustic navigation methods such as **LBL** and **USBL** have dominated the underwater localisation industry since 1960's [70]. The **LBL** is an acoustic localisation system where the distances among the baseline stations (i.e., transceivers) are long relative to the distance between them and the vehicle i.e., transponder. The baseline stations are fixed at some known positions on the sea surface or on the seabed as shown in Figure 3.1a. The vehicle interrogates the baseline station network by broadcasting a signal and each baseline station would respond back with its location; given the sound speed, range measurements can be obtained by **ToF** and therefore the **AUV**'s position can be obtained by trilateration [71].

The **Short Baseline (SBL)** has the same working principles of the **LBL** but the distances among the base stations do not exceed 20-50 meters. The base stations are normally mounted on different locations on the operation vessel as shown in Figure 3.1b.

The **USBL** is a hull-mounted system on a surface vehicle or buoy with hydrophones arrays (i.e., transducer) separated by very short distances typically 10 cm apart where acoustic signals **ToA** and phase delays are detected to triangulate a limited number of transponders' positions within localisation accuracy of 0.13-0.27% of slant range e.g., 10 **AUVs** can be navigationally aided with an update rate of 1 Hz by a hull-mounted **USBL** system from Sonardyne [66]. A larger number of **AUVs** can be navigationally aided by the **USBL** but with a lower update rate. The error characteristics of Sonardyne's **USBL** system [66] is utilised throughout the thesis and a small number of **AUVs** (i.e., 10) are typically navigationally aided with an update rate of 1 Hz due to the swarm mobility.

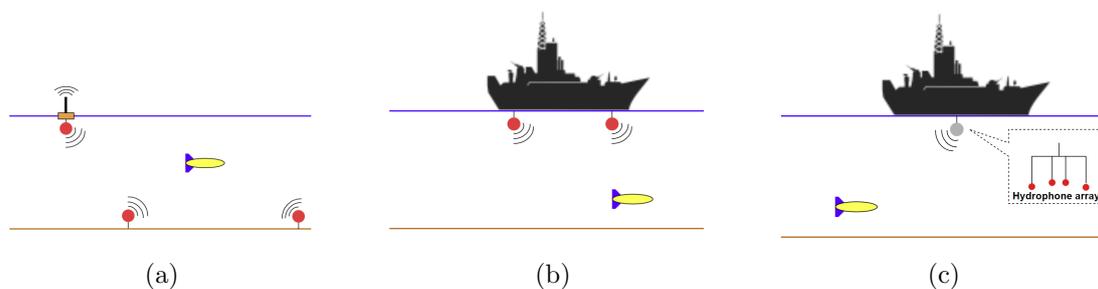


Figure 3.1: (a) Long Baseline (b) Short Baseline (c) Ultra-short Baseline.

Readers are referred to [72] and [73] for details on those systems. The localisation accuracy of **ToF** acoustic navigation can be further improved through careful utilisation of optimal filtering or sensor fusion techniques such as Kalman filtering [74]. The number

of underwater targets that can be localised in **ToF** acoustic navigation methods is limited and the update rate decreases proportionally with the number of targets due to **Time-division multiple access (TDMA)** utilisation in network interrogation [70]. Therefore, a large number of underwater localisation algorithms have been proposed in literature to obtain mass network localisation based on a few localised nodes [75]. Each localisation algorithm has its own merits and limitations which make each algorithm suits certain marine applications and underwater operating conditions but not all.

3.3.1 Sensor Network Localisation

A large and growing body of literature has investigated the localisation problem in **UWSN** [76, 77, 78]. In [79], authors proposed a localisation scheme **Large-scale Localisation scheme with No Prediction (LLNP)** for large-scale underwater sensor network. The authors in [80] extended the work proposed in [79] and introduced a new localisation scheme **Scalable Localisation scheme with Mobility Prediction (SLMP)**. Both **LLNP** and **SLMP** schemes considered scenarios of 3-dimensional **UWSN** that consists of anchor nodes with high communications capability and ordinary nodes i.e., unknown nodes. The **UWSN** considered in both **LLNP** and **SLMP** schemes consists of at least four surface buoys, anchor nodes and unknown nodes. Surface buoys are localised by the **GNSS** and anchor nodes have high communications capabilities so that they can contact the surface buoys for localisation. Both **LLNP** and **SLMP** schemes attempt to localise the unknown nodes. Distances among nodes are estimated in both schemes through **ToA**. In **LLNP**, authors assumed that anchor nodes are localised by contacting surface buoys. Unknown nodes localisation process is elaborated in Figure 3.2. During unknown node localisation process, each node maintains a counter, n , of broadcasted localisation messages with a predefined threshold of N and a counter, m , of reference nodes to which the distance is known with a threshold of 4.

The confidence value of any anchor node is set to 1 and λ is the confidence threshold which controls the possibility of a localised unknown node to become a new reference node. The confidence threshold in both **LLNP** and **SLMP** schemes was set to 0.98. The authors successfully demonstrated that the proposed scheme outperforms the recursive scheme proposed in [81] and the euclidean scheme proposed in [82].

Based on the group movement properties of underwater objects, the authors extended the proposed **LLNP** scheme to the **SLMP** scheme presented in [80]. Researchers in hydrodynamics showed that underwater objects move in semi-periodic manner [83] and

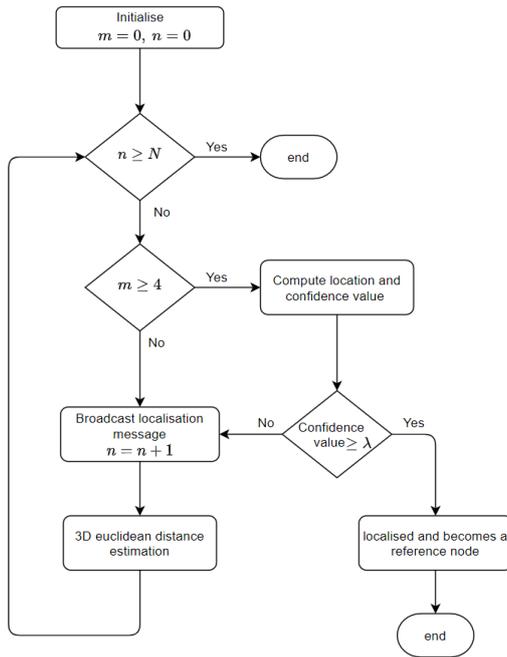


Figure 3.2: Unknown node localisation process in LLNP [79]

[84]. The SLMP is one of the very few algorithms investigated the impact of nodes location history on its next location. Similar to LLNP, anchor nodes are assumed to be localised by contacting four or more surface buoys. However, both anchor nodes and unknown nodes attempt to predict their mobility pattern based on a linear prediction method [85] and nodes mobility patterns are exchanged among nodes. Unknown nodes localisation process in SLMP is to a great extent similar to that in LLNP except the linear mobility prediction part. Interested readers can refer to [80] for more details. Figure 3.3 shows the average normalised localisation error with node density in LLNP and SLMP under the same simulation sittings. The authors in [79] and [80] defined the node density as the expected number of nodes that lies in a node’s communication range.

The power consumption is a crucial aspect of underwater wireless sensor nodes survival and localisation, the researchers in [86] proposed a reactive localisation algorithm. As its name implies, a node is localised when an event is detected [86]. Nodes that detect an event broadcast a “hello” message includes their ID and their energy level. By K-node coverage algorithm [86] four anchor nodes are detected and received the message. Anchor nodes reply back with their location and hence unknown node can localise itself

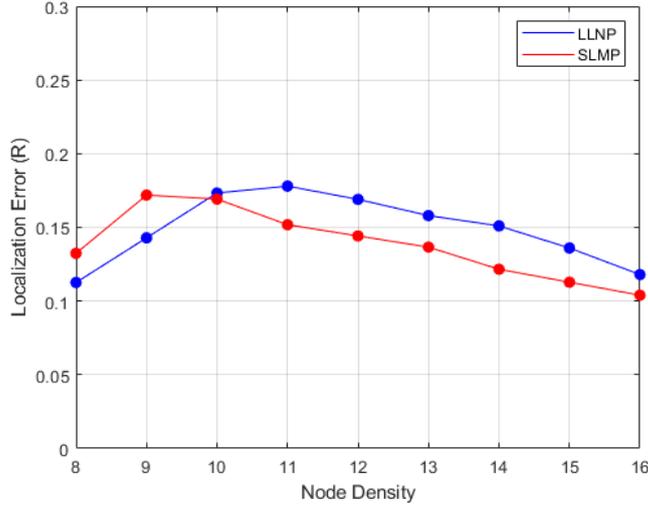


Figure 3.3: Relationship between the normalised localisation error to the node’s communication range and the node density in LLNP [79] and SLMP [80].

by multilateration [86].

On the other hand, a more power consuming but more accurate localisation algorithm was proposed in [87]. The authors assumed that there are three anchor nodes (A, B and C) with known positions and one unknown node (S) [87]. (A) is a master node which periodically broadcasts beacon messages. Node (B) starts broadcasting as soon as it receives messages from node (A) and node (C) starts broadcasting when it receives messages from node (A) and node (B). Unknown node (S) will not run its localisation algorithm till it receives messages from all other nodes (A, B and C). Time differences of receiving and broadcasting will be included in the transmitted messages. Hence, unknown node (S) will be able to measure its distance to all other anchor nodes by TDoA and localise itself [87]. The whole algorithm would fail and put on hold, if an anchor node e.g. (B) does not receive a message from anchor node e.g. (A). Therefore, a time-out method was proposed in an enhanced version [88] which significantly improves the localisation speed. The researchers in [52] went even further to improve the previous algorithm in which they increased reference nodes number. A localised unknown node can act as a reference node and, therefore, any unknown node lies outside anchor nodes communication range can be localised by the recent localised unknown nodes.

In all above-mentioned algorithms, wave propagation characteristics and sound speed variation in water column are not considered. All algorithms assumed that acoustic waves propagate in straight lines and acoustic signals have constant speed in water.

Acoustic waves propagation and speed in water depend on salinity, depth and temperature [89]. Localisation using ray tracing presented in [90] considered the propagation path of acoustic signals and sound speed variation with water depth. Simulation results showed that localisation accuracy has been improved. Another algorithm proposed in [91] “dive and rise” considers a realistic underwater mobility model called meandering current mobility [92]. A probabilistic approach have been considered for the localisation problem of a mobile UWSN that moves freely with the ocean tides. Moreover, the authors in [93] have investigated the probabilistic locality model and k -trees theory to estimate the uncertainty associated with a localised node.

3.3.2 Least-Squares Trilateration

Locating an object by its range measurements (e.g., achieved by means of ToF) to three reference locations is known as *trilateration*. The term *multilateration* is used when there are four or more references. This problem in its simplest form can be interpreted as finding the intersection of four spheres. The three-dimensional localisation problem can be converted into its 2D equivalence via orthogonal projection [94]. Figure 3.4 shows a 2D trilateration problem, with no error in range measurements, interpreted as finding the intersection of three circles. A considerable amount of literature has been published on solving the trilateration problem in robotics. In principle, an object

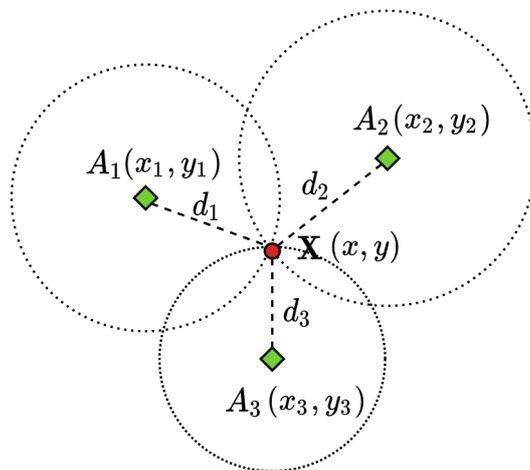


Figure 3.4: Two-dimensional trilateration problem of determining an object (red solid circle) location \mathbf{X} given the location of three stations (green diamonds) A_i and the range measurements/distances d_i ($i = 1, 2, 3$).

location (x, y, z) can be determined given the location of multiple references (x_i, y_i, z_i) and their corresponding range measurements d_i by solving a system of equations in the form of Equation 3.28.

$$(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 = d_i^2 \quad (3.28)$$

Closed-form and numerical solutions have been proposed in the literature to solve the trilateration problem. A closed-form solution has been presented by Thomas and Ros in [95] that derives a formula containing a few numbers of Cayley-Menger determinants [96] related to the geometry of tetrahedra. Coope in [97] presented a rather generic closed-form method to find the intersection points on I spheres in \mathbb{R}^I . All closed-form solutions have relatively low computational complexity and do not accommodate the situation when a solution does not exist. However numerical methods are more computationally complex but they estimate the best solution in case of unique intersection point of the spheres does not occur due to for example range measurements errors. A numerical method called Taylor series estimation was presented in [98] to iteratively improve the initial guess by finding the local linear least-sum-squared-error correction. A study by Nadivi et al. compared three statistical methods in the trilateration context namely linear least-squares estimator, iteratively re-weighted least-squares estimator and nonlinear least-squares estimator and showed that nonlinear least-squares estimators perform the best if error in range measurements is considered [99]. In nonlinear least-squares the problem of trilateration in Equation 3.28 is re-written as in Equation 3.29. Equation 3.29 can be minimised either by local deterministic optimisers such as Gauss-newton [100] that requires finding the derivative to minimise the residual of Equation 3.29 or by global stochastic optimisers such as Particle Swarm Optimisation (PSO) [101].

$$\min_{\mathbf{X}} \sum_{i=1}^I (\|\mathbf{X} - A_i\|_2 - d_i)^2 \quad (3.29)$$

where $\|\cdot\|_2$ denotes the Euclidean norm, I is the total number of reference nodes and A_i is the position of reference node of index (i) i.e., $A_i = \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix}$ and d_i is the corresponding distance between \mathbf{X} and A_i . We seek \mathbf{X} coordinates i.e., $\mathbf{X} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$ such that Equation 3.29 is minimised.

Although local deterministic optimisers such as Newton's methods rely heavily on the initial guess and they do not converge in some cases where singularities may occur i.e., when the references are almost aligned, trilateration has been usually solved by

Newton's methods in literature [100]. The Gauss-Newton algorithm [100] starts with an initial guess $\mathbf{X}^{(0)}$ till the cost Equation 3.29 is iteratively minimised. Given the residual functions $\mathbf{r}_i(\mathbf{X}) = \|\mathbf{X} - A_i\|_2 - d_i$, Gauss-Newton step proceeds as follows

$$\mathbf{X}^{(s+1)} = \mathbf{X}^{(s)} - (\mathbf{J}_r^\top \mathbf{J}_r)^{-1} \mathbf{J}_r^\top \mathbf{r}(\mathbf{X}^{(s)}) \quad (3.30)$$

where the superscript (s) is the iteration index, $\mathbf{r}(\mathbf{X}) = \begin{pmatrix} r_1(\mathbf{X}) \\ \vdots \\ r_I(\mathbf{X}) \end{pmatrix}$ is the vector of residuals, the symbol $^\top$ denotes matrix transpose and \mathbf{J}_r is the Jacobian matrix of $\mathbf{r}(\mathbf{X})$. This iterative process is carried out until either the maximum number of iteration is reached or no further improvement is achieved i.e., $\|\mathbf{X}^{(s)} - \mathbf{X}^{(s-1)}\|_2$ is lower than a predefined threshold.

In general, global stochastic optimisers tend to perform better in nonlinear least-squares problems but they are computationally expensive. PSO starts with initialising a set of candidate solutions (particles) randomly over the search space, each particle keeps updating its best experience ($\mathbf{P}best$) over the objective function and the algorithm keeps updating a record of the entire swarm's best experience ($\mathbf{G}best$) over the objective function. The velocity V_j and the position \mathbf{X}_j of each particle j are then determined in every iteration based on ($\mathbf{P}best$) and ($\mathbf{G}best$) according to the following formula

$$V_j^{(s+1)} = \omega V_j^{(s)} + c_1 r_1 (\mathbf{P}best_j^{(s)} - \mathbf{X}_j^{(s)}) + c_2 r_2 (\mathbf{G}best^{(s)} - \mathbf{X}_j^{(s)}) \quad (3.31)$$

$$\mathbf{X}_j^{(s+1)} = \mathbf{X}_j^{(s)} + V_j^{(s+1)} \quad (3.32)$$

where (s) is the iteration index, c_1 and c_2 are two positive constants, r_1 and r_2 are two randomly generated numbers $\sim \text{unif}(0,1)$, ω is the inertia constant such that $\omega = 0.9 - (0.005s)$, $\mathbf{P}best$ is the best position particle based on its own experience and $\mathbf{G}best$ is the global best position particle based on the entire swarm's experience. The constants c_1 and c_2 are typically set to 2 as per the common practice of PSO [102]. In PSO each particle represents a candidate solution and particles are randomly generated over the search space at the first iteration. Each particle $\mathbf{X}_j^{(s)}$ is evaluated at the iteration (s) on the objective function i.e., Equation 3.29. $\mathbf{P}best$ and $\mathbf{G}best$ are updated at each iteration accordingly. Each particle's velocity and position are then updated as in Equations 3.31 and 3.32 respectively.

Zhou in [103] proposed an algorithm with low computational complexity and high operational robustness for solving the nonlinear least squares trilateration /multilateration problem in a closed-form solution using standard linear algebra techniques. In other words, the algorithm in [103] can find the best approximation in a closed-form solution even if the solution cannot be found using others closed-form methods due to errors in range measurements. The nonlinear least squares trilateration /multilateration problem in Equation 3.29 can be rewritten as

$$\min_{\mathbf{X}} \sum_{i=1}^I ((A_i - \mathbf{X})^\top (A_i - \mathbf{X}) - d_i^2)^2 \quad (3.33)$$

Consider a matrix \mathbf{H} where \mathbf{H} is obtained by Equation 3.34 and the k-th row of \mathbf{H} is denoted by h_k^\top .

$$\begin{aligned} \mathbf{H} &= -\frac{2}{I} \sum_{i=1}^I A_i A_i^\top + 2\mathbf{c}\mathbf{c}^\top \\ \mathbf{c} &= \frac{1}{I} \sum_{i=1}^I A_i \end{aligned} \quad (3.34)$$

A matrix of $(n-1) \times n$ can be constructed as $\mathbf{H}' = [h_1 - h_n, \dots, h_{n-1} - h_n]^\top$ where $n = \text{rank}(\mathbf{H})$. Using orthogonal decomposition [104], the orthogonal matrix \mathbf{Q} and the upper diagonal matrix \mathbf{U} can be obtained, \mathbf{H}' can be obtained by

$$\mathbf{H}' = \mathbf{Q}\mathbf{U} \quad (3.35)$$

Consider a new matrix \mathbf{f} defined by Equation 3.36 where the k-th component of \mathbf{f} is denoted by f_k .

$$\mathbf{f} = \mathbf{a} + \mathbf{B}\mathbf{c} + 2\mathbf{c}\mathbf{c}^\top \mathbf{c} \quad (3.36)$$

where \mathbf{a} and \mathbf{B} are defined as in the following

$$\begin{aligned} \mathbf{a} &= \frac{1}{I} \sum_{i=1}^I (A_i A_i^\top A_i - d_i^2 A_i) \\ \mathbf{B} &= \frac{1}{I} \sum_{i=1}^I (2A_i A_i^\top - (A_i^\top A_i)\mathbf{I} + d_i^2 \mathbf{I}) \end{aligned} \quad (3.37)$$

A vector of $n-1$ elements can be constructed as $\mathbf{f}' = [f_1 - f_n, \dots, f_{n-1} - f_n]^\top$.

Given the orthogonal matrix \mathbf{Q} and \mathbf{f}' , a matrix \mathbf{V} is defined by

$$\mathbf{V} = \mathbf{Q}^\top \mathbf{f}' \quad (3.38)$$

Consider a matrix \mathbf{q} where q_k is the k-th component of \mathbf{q} ; q_1 and q_2 are defined by

$$\begin{aligned} q_1 &= \left(\frac{u_{12}v_2}{u_{11}u_{22}} - \frac{v_1}{u_{11}} \right) + \left(\frac{u_{12}u_{23}}{u_{11}u_{22}} - \frac{u_{13}}{u_{11}} \right) q_3 \\ q_2 &= -\frac{v_2}{u_{22}} - \frac{u_{23}}{u_{22}} q_3 \end{aligned} \quad (3.39)$$

where v_k is the k-th component of \mathbf{V} and u_{kj} is the (k, j) element of the upper diagonal matrix \mathbf{U} . The following Equation 3.40 represents a valid constraint as $q_1^2 + q_2^2 + q_3^2 = \mathbf{q}^\top \mathbf{q}$

$$q_1^2 + q_2^2 + q_3^2 = -\frac{1}{I} \sum_{i=1}^I A_i^\top A_i + \frac{1}{I} \sum_{i=1}^I d_i^2 + \mathbf{c}^\top \mathbf{c} \quad (3.40)$$

The parameters q_1 , q_2 and q_3 can be obtained by solving the equations in 3.39 with Equation 3.40 as we have 3 unknowns and 3 equations. Given \mathbf{q} and \mathbf{c} vectors, the unknown \mathbf{X} (i.e., **AUV**'s position) can be obtained from the following linear equation

$$\mathbf{X} = \mathbf{q} + \mathbf{c} \quad (3.41)$$

Interested readers are referred to [103] for the mathematical derivation of the the aforementioned equations.

The three methods are implemented and compared to solve the least-squares multilateration problem i.e., Gauss-Newton [100], PSO [101] and analytical closed-form approach [103]. Monte-Carlo simulations of 100 random walkers in a confined region of 100 m³ was considered to compare the three aforementioned nonlinear least-squares solvers for the multilateration problem. If any 3 or more walkers are within a communication range of 25 m of another walker, trilateration/multilateration is performed with zero error in range measurements. The following histograms compare the accuracy of each method i.e., Gauss-Newton (local gradient-based optimiser), PSO (global stochastic optimiser) and analytical approach presented in [103] in solving the least-squares trilateration/multilateration problem. The three aforementioned methods have been adopted to solve the least-squares trilateration/multilateration problem on the exact same network topology for each simulation step. The trilateration/multilateration process has been carried out around 15000 times for each method/optimiser. The maximum number of iteration and the population size for the PSO were set to 200 iterations and 300 particles.

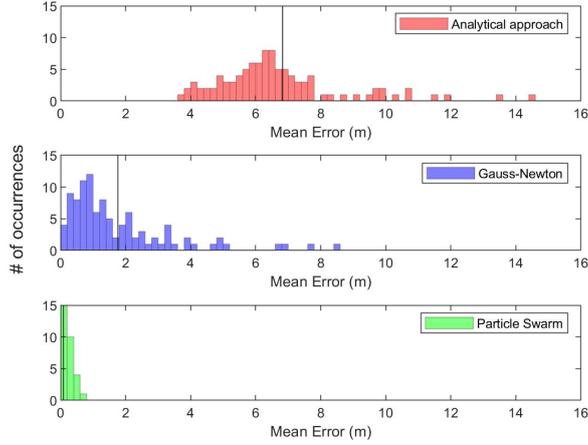


Figure 3.5: Histograms of mean multilateration error of around 15000 multilateration process carried out in 100 walkers (nodes); The vertical line in each histogram represents the mean error of the entire simulation.

Given that Equation 3.29 is a nonconvex problem that may have local extrema, PSO performs the best in minimising it as PSO method is capable of exploring the entire solution space due to its population-based nature. On the contrary, Gauss-Newton [100] heavily relies on the initial guess (i.e., the proximity of the initial guess to an extremum). Although, the analytical approach presented in [103] is the least accurate, it is a highly computationally efficient approach and it may perform well if Equation 3.29 is convexified as in [105]. Figure 3.5 clearly shows the superior performance of PSO [101] over the analytical approach [103] and Gauss-Newton [100] in solving the multilateration problem. The entire swarm mean localisation error when the analytical approach [103], Gauss-Newton [100] and PSO [101] are adopted is 6.82 m, 1.76 m and 0.089 m respectively. There is an improvement of 74% in mean trilateration error when Gauss-Newton optimiser is compared to the analytical approach presented in [103] and an improvement of 94% when PSO is compared to Gauss-Newton. Therefore, PSO is adopted in subsequent chapters to solve the least-squares trilateration/multilateration problem.

3.3.3 Underwater Multi-vehicle Localisation

Over the past two decades, multi-agent and swarm robotic systems have become a very attractive research area as it has been established, in many applications, that swarm

robotics can provide relatively low-cost solutions when tasks are too complicated or when wide region coverage is required e.g. surveillance missions [9]. A large and growing body of literature has investigated swarm robotic research [9, 106]. Sahin in [9] identified five criteria to distinguish swarm from other multi-agent robotic systems, namely autonomous robots, relatively incapable individually, large number of robots, a few homogeneous groups of robots and local sensing capability.

Localisation is one of most investigated research topics in terrestrial multi-robot and swarm systems [107, 108]. Most terrestrial and aerial swarm localisation algorithms reported in the literature are not directly applicable to underwater robotic swarm systems as it is always assumed that information among swarm nodes can be easily exchanged. The severely limited bandwidth and long latency of the underwater acoustic communications limit the number of AUVs that can be deployed at once to collaboratively complete a mission [19, 20, 21]. Underwater multi-agent robotic systems mainly rely on acoustic communications to exchange information among team members with an average propagation speed of 1500 m/s (i.e., speed of sound in water) with a maximum bit rate of around 60 kbps. On the contrary, information among members of multi-agent terrestrial or aerial robotic systems are exchanged in the speed of light of 3×10^8 m/s with bit rate in Mbps. In addition, the high cost of maritime assets and sea-trials in comparison to aerial and terrestrial assets has limited the deployment of underwater swarm systems. Much of the recent research in cooperative multi-agent maritime systems has focused on the provision of path planning algorithms for oceanic field sampling [109, 110]. A fleet of three gliders has been deployed for temperature observations in [111] and a fleet of six gliders has been deployed for adaptive sampling and prediction in [110]. In both [111, 110], gliders coordinate their motion for ocean sampling but they rely on radio communications for cooperation when they are on the sea surface. The work in [109] considered a team of three AUVs for adaptive ocean sampling where the problem of reconstructing an oceanic field has been approached as a deterministic optimisation problem. Each node in the team is in favour of covering the area of interest with minimum sampling points while keeping in contact with the rest of the team [109]. Likewise in [112], the authors proposed a cooperative sampling approach “sampling-on-demand” to support a fleet of underwater gliders to optimise the sampling process with a pre-defined acceptable uncertainty. A behaviour-based cooperative algorithm for a mobile sensor node was proposed in [113] where each sensor node is mounted on an AUV. The algorithm in [113] has been evaluated based on the coverage performance. It is worth mentioning that most of the work in cooperative underwater robotics have considered a heterogeneous maritime assets i.e., a group of sea-surface vehicles collaborating with

a few underwater vehicles to complete a mission e.g., oceanic field sampling. Moreover, when a team of AUVs is deployed, they typically keep in contact with a command and control centre to facilitate their cooperation [15].

A few articles have addressed the acoustic localisation problem of a cooperative team of a few number of AUVs i.e., 3-4 AUVs. A centralised EKF algorithm was proposed in [57] where the algorithm has access to all sensor data including range measurements to reduce the AUV's location estimate uncertainty. A decentralised approach was proposed in [114] where the Extended Information Filter (EIF) is utilised to enhance the performance of the algorithm proposed in [57]. The authors in [114] considered a server and a client AUVs and showed that the decentralised EIF is able to estimate the client's state and track the joint probability distribution between the server and the client with a similar performance of the centralised EKF reported in [57] without access to the server's sensor data. Both [57, 114] have considered a single beacon navigation aid (i.e., range measurements) to reduce the AUV's location estimate uncertainty. Similarly, the authors in [115] employed the EKF for range measurements aid and ToF acoustic navigation aid (i.e., USBL fixes) in a network of a USBL, an AUV and two static sensor nodes. A single beacon navigation aid represented in range measurements update between two cooperative AUVs was investigated in [116]. The results in [116] showed that particle filter provides better location estimates of the AUV than the EKF in range-only measurements update. The development of smart mobile sensor network that provides node localisation as a service for an existing acoustic network was discussed in [117]. A fleet of small and low cost AUVs (i.e., ecoSUB [118]) was utilised in [117] where range measurements aided DR navigation was implemented for node localisation [119]. Based on the same principles of range measurements update that was presented in [116], the authors in [119] considered the Two-Way Travel Time (TWTT) method for range measurements and emphasised the node's mobility impact on the localisation process by both adopting time window of past range measurements and accounting for nodes' velocity.

3.4 Underwater Communications

Cooperative systems are in general indispensable to establish reliable communications channel among the nodes. Effective and well established communications means are in particular vital to underwater range measurements and localisation which leads to the success of underwater robotic missions. Acoustic channel is the most utilised channel for wireless communications in underwater environments [15]. However, there are other

communications technologies that can be utilised in underwater environments such as RF electromagnetic and optical links but they are rather immature technologies for underwater wireless communication. RF electromagnetic waves suffer from high attenuation in underwater environments caused by electrical conductivity and high permittivity of the water [120]. RF electromagnetic technology is featured in its high propagation speed and its immunity to both acoustic noise and water turbidity [121]. Therefore, it can be adopted for underwater wireless communication over short distances (i.e., less than 10 m) with data rate of 3 Mbps in fresh water and 3 kbps in sea water for applications such as AUV docking and diver’s personal network [120]. Underwater optical communications also suffers from high attenuation due to scattering loss and absorption [122]. In addition, it requires line-of-sight and it is susceptible to water turbidity. Underwater optical waves can provide high data rate (i.e., 5 Mbps) over 100–200 m in water clarity with *e*-folding depth of 40 m [123]. On the other hand, acoustic waves in underwater environments can propagate over long distances but suffer from limited bandwidth, high propagation delay and multipath propagation [124]. The range in underwater **Acoustic Communications (ACOMMS)** is inversely proportional to the frequency i.e., a maximum communication range of 10 km can be achieved over a frequency band of 7 – 17 kHz with data rate of 6.9 kbps [34], whereas a maximum communication range of 300 m can be achieved over a frequency band of 120 – 180 kHz with data rate up to 62.5 kbps [22]. The following table 3.1 summarises some of the key characteristics of different underwater wireless communications technologies.

Type	Range	Data rate	Propagation speed (m/s)
RF electromagnetic	up to 50 m	8k - 300 bps	up to 4.30×10^6
Optical	up to 200 m	5 Mbps	3×10^8
Acoustic	300 m - 10 km	6.9 - 62.5 kbps	1500

Table 3.1: Underwater wireless communications technology

3.4.1 Physical Layer

Modulation, error correctness and channel equalisation are the main functionalities of the physical layer. Exploiting the limited bandwidth of the underwater acoustic channel is the main objective in designing a modulation technique [124]. **Frequency-Shift Keying (FSK)** based on energy detection is a reliable, simple and robust noncoherent modulation scheme and it copes well with the received signal multipath and Doppler in

applications of communicating low data rate required signals i.e., control and command [15]. An example of noncoherent modulation scheme of 0.6 kbps data rate and 2.9 km range with 10^{-3} bit error rate is presented in [125].

Although rapid phase variation characterises the underwater acoustic channel, phase-coherent modulation schemes such as [Phase-Shift Keying \(PSK\)](#) and [Quadrature Amplitude Modulation \(QAM\)](#) are considered to increase both the communication range and spectral efficiency but that was not possible without the improvements in phase tracking algorithms [124]. Phase-coherent modulation techniques are classified into purely phase-coherent and differentially coherent [124]. [Differential Phase-Shift Keying \(DPSK\)](#) is considered as an intermediate solution between purely coherent and noncoherent as it offers simple carrier recovery if it suffers from higher bit error rates compared to [PSK](#) [124]. An example of [DPSK](#) modulation is shown in [126] of 20 kbps data rate in 1 km range with 10^{-2} bit error rate. Multicarrier modulation techniques such as [Orthogonal Frequency-Division Multiplexing \(OFDM\)](#) is usually used to overcome the long delay spread [124]. [OFDM](#) is not commonly used in underwater [ACOMMS](#) due to the presence of large Doppler spread. However, [OFDM](#) schemes have been recently investigated for underwater [ACOMMS](#) [124]. Experimental results were shown in [127] where [Binary Phase-Shift Keying \(BPSK\)](#) modulation have been adopted in underwater acoustic communications modems. The authors in [127] achieved data telemetry bit rates of 500 bps at a distance of 10 km with bit error rate of 0 out of 4976 bits.

3.4.2 Medium Access Control Layer

Underwater acoustic communications are featured in limited bandwidth and high propagation delay [19]. Therefore, [MAC](#) layer should be carefully considered to achieve the maximum packet delivery ratio. Several multiple access techniques have been investigated for underwater acoustic sensor networks such as [TDMA](#), [Code Division Multiple Access \(CDMA\)](#) and [Carrier-Sense Multiple Access \(CSMA\)](#). [TDMA](#) and random access such as [ALOHA](#)-based access control [128] are the simplest to implement but they both provide limited channel utilisation [124]. [TDMA](#) is probably the most commonly adopted technique in underwater acoustic networks but it requires the networks to be synchronised and it does not provide maximum utilisation of the channel specially in large scale networks [129]. Each node in the network is allocated a certain time slot to transmit its packet in one [TDMA](#) frame; see Figure 3.6.

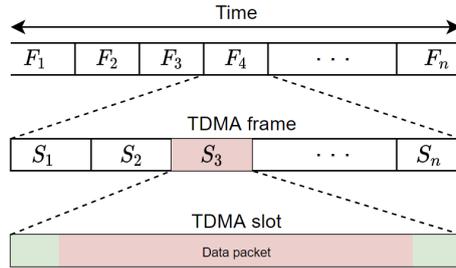


Figure 3.6: TDMA-based MAC layer where the time is divided into frames, each frame is divided into slots and each slot is assigned to a single node where it is allowed to transmit its data packet, guard time is added at both ends of each time slot to prevent packets collision.

However, there have recently been further investigation in utilising TDMA scheme without clock synchronisation [130]. In contrast, a node in CSMA scheme is required to listen to the medium for a short period of time and transmits its packet if the medium is idle [131]. However, underwater acoustic channel may be sensed idle when it is not due to the high propagation delay [124].

A narrowband signal can be transmitted over a wideband signal by multiplying each symbol by a spreading code of certain length in the physical layer and each node is assigned with a unique spreading code that can resist interference from multiple nodes, this can provide a resilient MAC technique i.e., direct-sequence CDMA [132]. Frequency-division multiple access (FDMA) is not commonly used in underwater acoustics due to the limited bandwidth. However, recent research and field experiments have showed that single carrier FDMA is a promising MAC scheme for underwater ACOMMS [133], [134].

3.5 Conclusion

In summary, the complicated nature and wide-area coverage requirements of underwater missions such as exploration, environmental monitoring, geological and ecological research have drawn special attention to multi-agent cooperative systems. Selected underwater localisation algorithms suitable for UWSN have been reviewed including least-squares trilateration. Monte-Carlo simulations were conducted to compare different least-squares multilateration solvers i.e., Gauss-Newton, PSO and analytical closed-form approach [103]. There is an improvement of 74% in mean trilateration error when Gauss-Newton optimiser is compared to the analytical approach presented in [103] and

an improvement of 94% when PSO is compared to Gauss-Newton. Therefore, PSO is adopted in subsequent chapters to solve the least-squares trilateration/multilateration problem. Inertial navigation system that is the backbone of any mobile underwater autonomous system has been explained and some of the commercially available ToF-based acoustic navigation aids have been briefly discussed.

Although underwater cooperative multi-robot systems have been fairly investigated in literature, we lay greater emphasis on underwater cooperative swarm systems. The key challenges underlying underwater cooperative swarm systems are localisation and navigation. On top of the challenges that are associated with any swarm system such as troubleshooting and parameters tuning, the offshore nature adds some complicated environment-related challenges such as underwater communications. Therefore, underwater communications technologies have been reviewed and their limitations have been highlighted.

Chapter 4

Underwater Robotics Simulation Platform

A physics-based robotic co-simulation platform that considers the underwater acoustic communications characteristics is developed and explained in this chapter to validate and optimise the proposed localisation algorithms in this thesis for cooperative underwater robotic swarms. Underwater robotic swarm deployment is particularly challenging due to the high cost of maritime assets including a swarm of AUVs and a deployment vessel. Simulation tools provide efficient and cost-effective alternatives to progress towards cutting edge research in underwater robotics. In addition, physics-based simulation plays a pivotal role in parameters optimisation and the scalability testing of newly-developed algorithms. Underwater robotic swarm simulation e.g., [135] is able to provide a cost-effective way to evaluate the system performance (e.g., localisation accuracy and scalability) of large underwater robotic swarms.

There are many physics-based robotic simulators available for research and industry communities such as Webots [136], EASY-ROB [137], Gazebo [138] and CoppeliaSim [139]. Webots robotic simulator is employed in this thesis for robot physics and environmental hydrodynamics simulation. Webots is a 3D simulation environment that can model, program and simulate mobile robots [136]. Webots has many features that makes it one of the best robotic simulators. Webots has the capability of simulating a swarm of robots in a shared environment with local or global communication and provides a supervisory control and monitoring of any agent. Webots simulator has a wide variety of sensors and actuators that are commonly utilised in robotics systems. It provides

[Application Programming Interface \(API\)](#) for many programming languages such as C, C++, Python, Java and MATLAB; it can also be made compatible to any third party software through TCP/IP socket programming. Webots utilises the [Open Dynamics Engine \(ODE\)](#) library [140] for high performance rigid body dynamics simulation. Common robotic simulators such as Gazebo, CoppeliaSim and Webots do not have the capability of simulating underwater acoustic channels but they have the capability of simulating radio and infra-red links [141]. However, underwater acoustic communications properties cannot be simulated within any of them. Therefore, the Underwater Network Project for acoustic simulation (i.e., UnetStack) [142] is employed for underwater acoustic communications simulation. In this chapter, the Underwater Network Project for acoustic simulation (i.e., UnetStack) is briefly explored and validated throughout field experiments in section 4.1. Section 4.2 presents the simulated [AUV](#) and the implemented simulation platform. Mutual simulation settings in both Chapter 5 and 7 are provided in section 4.3. Section 4.4 summarises this chapter.

4.1 Underwater Acoustic Communications

Intra-swarm communication is the backbone of cooperative navigation algorithms, therefore a realistic underwater acoustic simulation must be considered in underwater robotic swarms or [UWSNs](#). To overcome the limitation of robotic simulators platforms in simulating underwater acoustic channel, a dedicated acoustic simulator is employed i.e., the Underwater Networks Project UnetStack [142]. UnetStack is an agent-based network stack and simulator developed to support highly optimised protocols for underwater acoustic sensor networks [143]. It allows easy network configurations and management and allows protocols to be simulated in realistic channel conditions. The nature of underwater acoustic communications requires significant cross-layer information sharing [144]. However most, if not all, network simulators are not designed for cross-layer sharing. The popular network simulator ns2 [145] as an example, has not been originally implemented for cross-layer sharing but an extension was released called *miracle* [146] to allow cross-layer information sharing. Underwater acoustics simulators such as DESERT [147] and SUNSET [148] are ns2-based and of course they made use of *miracle* plugin [146]. UnetStack however, adopted service-oriented agent architecture that fundamentally enables cross-layer interaction. UnetStack supports both discrete-event simulation mode and real-time simulation mode. Therefore, the developed protocol can be deployed to any compatible underwater communications modem such as Evologics

[22] for field experiments directly without the need for recompilation. Ns2 however, is primarily a discrete-event simulator, efforts are needed to make significant changes to run in real-time mode for field experiments. Nevertheless, moving a simulated protocol to real-world deployments requires additional non-trivial steps such as cross-compilation. UnetStack [142] is therefore selected to be integrated with Webots simulator for underwater ACOMMS simulation to validate the proposed cooperative underwater localisation algorithms.

In this section UnetStack simulator is validated by field experiments. We compare the packet loss in the underwater communication channel in both UnetStack simulation and field experiments when deploying the same acoustic modems and same network topology.

4.1.1 Acoustic Modem and Channel Models

A typical UnetStack simulation is composed of several agents. Each agent provides a set of services and functionalities required to build underwater network consists of several nodes to exchange data via acoustic links. A Unet agent called *modem model* (i.e., half duplex modem) simulates the physical services needed to emulate acoustic modem behaviours. The properties (i.e., data rate, frame length, and power level) of the control channel, data channel and frame type (i.e., JANUS [149]) can be specified in Unet modem model. Several other properties such that transmission delay when switching from receiving to transmitting modes and frame header length that control the modem's behaviour are also modeled in Unet modem model. JANUS is an underwater acoustic communication protocol that has been recognised as NATO standard [149].

The Unet *channel model* simulates the communication channel specifics i.e., bandwidth, carrier frequency, salinity and temperature. The Unet simulator provides different channel models namely protocol channel model and acoustic channel model. The user can modify the channel properties in each of the models but in each model only certain properties can be parameterised.

The protocol channel model is a simple channel model and can be parameterised by communication range, interference range, detection range, sound speed, probability of detection \mathbf{P}_D and probability of decoding \mathbf{P}_C . Despite its simplicity, it models key effects such as collisions, interference range, limited communication range and propagation delay. Successful communication is possible at ranges within the communication range with a probability $\mathbf{P}_D \times \mathbf{P}_C$. A packet is possibly detected with a probability of \mathbf{P}_D at ranges between the communication range and detection range but it will not be

decoded. Data packets are lost at any range less than the interference range if they are being received at the same time.

On the other hand, the acoustic channel model is comprised of two models namely Urick acoustic model [150] and BPSK fading communication model [124]. The Urick acoustic model is parameterised by water depth, temperature and salinity, bandwidth, carrier frequency, spreading loss factor and noise power spectral density level. Sound speed in this model is computed based upon the nine terms equation proposed by Mackenzie in [89] and transmission loss is computed as in [150].

Given the transmission source level \mathbf{S}_L in dB re μPa @1 m, the computed transmission loss \mathbf{T}_L and the predefined noise level \mathbf{N}_L , the Signal-to-Noise Ratio (SNR) would be $\mathbf{S}_L - \mathbf{T}_L - \mathbf{N}_L$ [143]. The BPSK fading model uses Urick acoustic model's SNR to simulate a frame's detection and successful decoding. The BPSK fading model is parameterised by Rician fading parameter, fast or slow fading, acceptable probability of false alarm during detection and processing gain [124]. In order to compute the effective SNR in the channel, $10 \log_{10}(\mathbf{B}/\mathbf{D})$ is added to the SNR for BPSK communication signal with data rate \mathbf{D} bps and bandwidth \mathbf{B} . Rician fading and Gaussian noise are assumed to simulate bit errors. If fast fading is enabled, the error in each bit is generated in the simulation independently from Rician fading model with Gaussian noise. In contrast, a single realisation of Rician fading model with Gaussian noise is added to the entire frame when slow fading is enabled instead. The frame is successfully received and decoded if all bits are successful and it is lost if any bit is in error.

The following Figure shows the acoustic modem and the channel models that can be simulation on UnetStack. The acoustic channel model with Urick acoustic and BPSK fading models is utilised in our UnetStack simulation as it is a more realistic representation of the underwater ACOMMS than the protocol channel model that relies on predefined high-level parameters i.e., communication range, probability of detection and probability of decoding.

4.1.2 Experiments and UnetStack Simulation

Field experiments have been carried out in Loch Earn in Lochearnhead, Scotland on the 19th of March 2019 in collaboration with the stakeholders of smart dust for large scale underwater wireless sensing project (USMART [151]). Seven static sensor nodes have been deployed arbitrarily at different locations and depths. Each sensor node is equipped with low power, bio-friendly acoustic modem with maximum communication

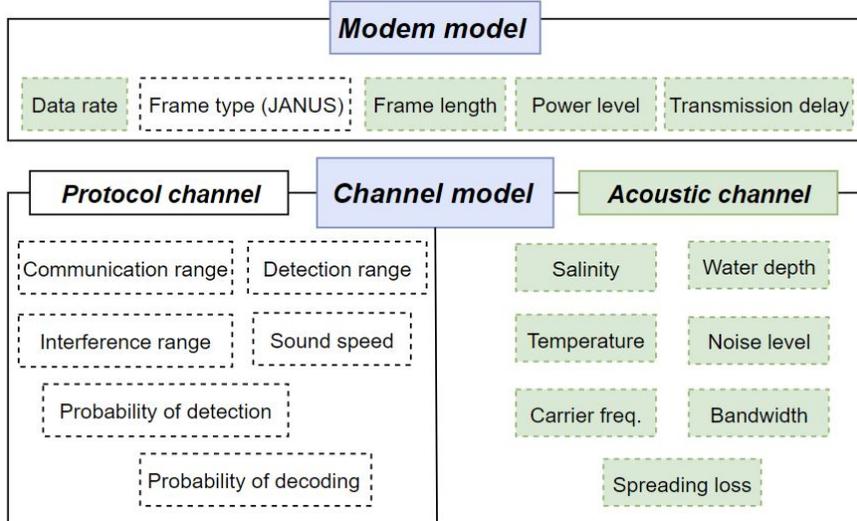


Figure 4.1: UnetStack acoustic modem and channel models and their parameters. The green highlighted models/parameters are utilised in our UnetStack simulation.

range of 2000 m, communication frequency of 24 – 28 kHz and acoustic data rate of 40-160 bps [152, 153]. Five of the sensor nodes are always on the receiving mode, one of the sensor nodes (gateway) is always on the transmitting mode and one acts as a relay, it re-broadcasts whatever it receives from the gateway. Figure 4.2 shows the location of each node on google map and their addresses in Loch Earn, the same network topologies are simulated on UnetStack for packet loss analysis. The only difference between the two network topologies is that one of the sensor nodes in one topology communicates with the gateway directly and through the relay node in another topology shown by the dashed lines in Figure 4.2. Both experiments (i.e., different network topologies) have been conducted separately. Each acoustic sensor node is deployed at a fixed location with a depth range of 10 – 22 m. Two experiments with different network topology have been conducted. In both experiments, the gateway broadcasts its address in 4 bytes every 10 seconds and the relay node (address 171) re-broadcasts the gateway message 5 seconds after it is received. The same network topology, configurations and communication modem specifications have been simulated on UnetStack. Table 4.1 lists the parameters of the communication modem and channel on UnetStack simulation that emulate Loch Earn experiments shown in Figure 4.2. The channel average packet loss of the field experiments was around 6.2% which was unexpectedly low, this could be due to **i**) the idleness of water as the experiments were carried out in March when there was not any activities in the lake whatsoever and **ii**) the long allocated time-slot for each



Figure 4.2: Network topology - top view; sensor nodes positions, MAC addresses and the acoustic links (arrows) marked on Google map of Loch Earn field experiments. The depth of each sensor node is shown in red. The two dashed lines show two different topologies.

Parameter	Value
Communication modem Freq band	25 kHz
Communication data rate	160 bit/s
Data packet length and duration	4 bytes; 200 ms
Data packet allocated TDMA time-slot length	5 s
Noise level	100 dB
Water salinity	1 ppt
Water temperature	10 °C
Rician fading parameter	10
Fast fading	enabled

Table 4.1: UnetStack simulation settings and parameters of the communication channel and modem

data packet to be transmitted (i.e., 5 seconds) in comparison to its duration of 200 ms. It is worth mentioning that the packet loss achieved in this field experiments are only valid in its environment with its certain experimental setup. The average packet loss in an underwater acoustic communication channel as reported in [114] is 15-40% and it depends on many factors such as the noise level and the guard time of each broadcast message. On the other hand, the simulated underwater acoustic communication channel packet loss was equal to 14.4% given that the noise level was set to 100 dB. The channel average packet loss in simulation is 56.9% higher than the field experiments and therefore

noise level for acoustic channel simulation in subsequent chapters is reduced to 60 dB. More field experiments are detailed in [143] to validate UnetStack simulator.

4.2 Simulation Platform Architecture

The developed co-simulation platform consists of three interacting components, namely physics-based robot simulator, underwater acoustic communications simulator and MATLAB Navigation toolbox for IMU modelling. Realistic simulation can be provided by the physics-based robot simulator by considering the mechanics of the AUVs and the hydrodynamic effects. The motion behaviours of AUVs are under significant effect of the geometry, dynamics of the AUVs themselves, as well as the hydrodynamic effect in underwater environment. Neglecting dynamics and hydrodynamics in simulation would compromise the developed algorithm's performance when it is deployed for field experiments. A swarm of identical AUVs and static deployment vessel are simulated on Webots; each AUV is subject to the static force (i.e., Archimedes' thrust) and the dynamic force (i.e., drag force) exerted by the simulated fluid properties on Webots such as density, viscosity and stream velocity [136]. The modelled AUV has three thrusters, two of them are horizontally placed on the left and the right side of the AUV to generate the surge velocity and to control the rotation around the vertical axis i.e., yaw angle. The third thruster is placed vertically to control the rotation around the transverse axis i.e., pitch angle. Figure 4.3 shows the front, side and top views of the modelled AUV. It can be noticed that the design of this AUV does not have an active actuation around the longitudinal axis (i.e., roll angle) but the rudders have positive dihedral which improves the stability around the longitudinal axis. Each AUV in the

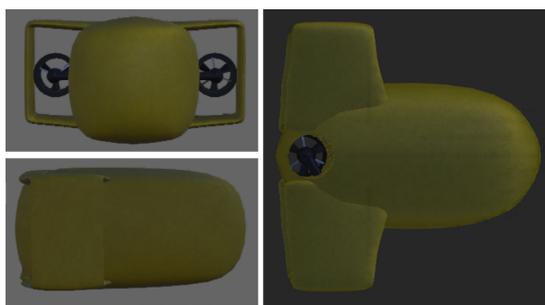


Figure 4.3: The front, side and top views of the modelled AUV shows two horizontally placed thruster and a vertical one. Autonomous Robotics Limited All rights reserved.

swarm has its own controller that is implemented using Webots Matlab APIs. Figure 4.4 shows a Webots scene example of AUVs swarm deployment in shallow water and the deployment vessel. A Webots supervisor-enabled node (Sniffer) is allocated to monitor



Figure 4.4: Underwater Webots simulation scene of 50 AUVs deployment, the USBL transceiver is hull-mounted on the deployment vessel.

each AUV in the swarm. If an AUV broadcasts a navigation aid, the Sniffer reports the network topology to UnetStack, runs the simulation and reports back the results to each AUV in the swarm in terms of the received message content (if any) and time of arrival. It is worth mentioning that the USBL system is not acoustically simulated as the error characteristics of the assumed USBL localisation system accounts for the localisation error due to non-ideal acoustic channel i.e., 10 AUVs can be localised with localisation accuracy of 2.7 m ($1-\sigma$ error) in 1000 m water depth in one TDMA frame of 1 second [66]. In addition, the characteristics of commercially available USBL system is proprietary which makes it hard to simulate the low-level details of the USBL. However, the centralisation of the USBL system and its TDMA frame and slots are simulated by a Webots supervisor-enabled node as the proposed algorithm requires the USBL to prioritise the Navigation Beacon (NB) AUVs over the rest when localisation requests received by the USBL is more than it can aid in a single TDMA frame. NB AUVs are randomly selected AUVs in the swarm and allowed to broadcast localisation aids based on some predetermined localisation performance criteria.

Figure 4.5 shows an example simulation scenario of four AUVs navigate to their destinations and only three of them broadcast localisation messages, the “Sniffer” reports the network topology to UnetStack simulator and reports back the results of localisation aids delivery and time of arrival to each AUV in the swarm.

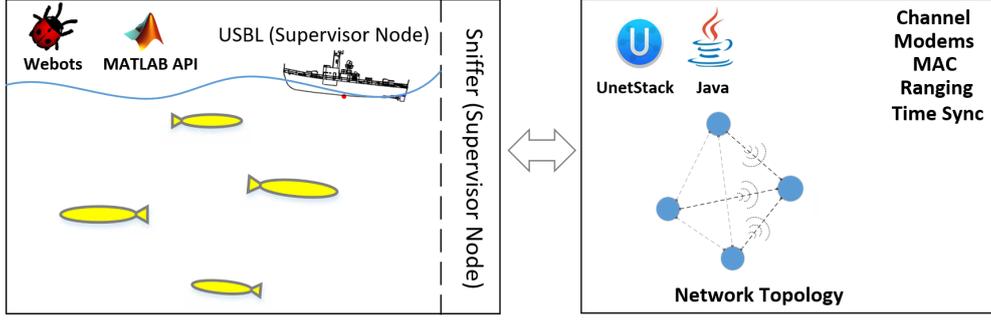


Figure 4.5: The implemented simulation platform used to validate the proposed navigation algorithm. Webots robotic simulator is employed for physics simulation and UnetStack is employed for simulating the underwater acoustic communication properties.

If an **AUV** that needs to be localised (i.e., based on some localisation performance criteria) receives localisation aids within a predefined time window from **NB AUVs**, it incorporates the new location estimate within its current location estimate using for example the **EKF**.

The network's mobility is neglected if all localisation aids are received within a time window of 200 milliseconds as the simulated **AUV** average velocity is about 0.5 m/s. The maximum unaccounted displacement due to network mobility is thus 10 cm, which is insignificant compared to the typical mean localisation error obtained i.e., 0.4%. However, the **AUV's** velocity impact on the trilaterated navigation aid due to time delay imposed by UnetStack simulation is acknowledged. When the network topology is reported to UnetStack (in the case of some **NB AUVs** are transmitting localisation aids), Webots simulation is frozen and UnetStack simulation is run on discrete-event mode. Once the results of UnetStack simulation is reported back to the **AUVs'** controllers, Webots simulation is run by the duration of UnetStack simulation time ΔT_{Unet} and only then an **AUV** has access to its received localisation aids. If an **AUV** a^i has successfully received three or more localisation aids and performed trilateration, the trilaterated position due to the **AUV's** mobility during UnetStack simulation is linked to its velocity v_t^i by the following

$$p_{LAT,t_2}^i = p_{LAT,t_1}^i + \int_{t_1}^{t_2} v_t^i dt \quad (4.1)$$

where $t_2 - t_1$ represents the time that UnetStack took to run the simulation ΔT_{Unet} or in other words it is the timestamp of the last localisation aid being received in the entire swarm, p_{LAT,t_1}^i is the **AUV's** trilaterated position when navigation aids are received at

time t_1 and p_{LAT,t_2}^i is the position that AUV a^i is aided by at time t_2 . Figure. 4.6 shows three NB AUVs broadcast localisation aids at time t_1 ; the localisation aids are received by AUV a^4 . The trilaterated p_{LAT,t_1}^4 solution corresponds to a^4 position when it was at $p_{t_1}^4$. AUV a^4 is aided by p_{LAT,t_2}^4 which takes into consideration the mobility of a^4 .

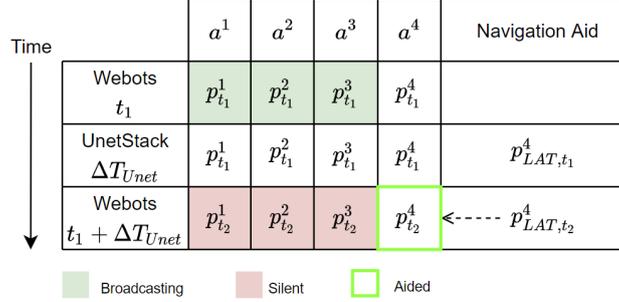


Figure 4.6: Network mobility considerations when trilateration is performed.

4.3 Simulation Settings and Scenario

A simple finite state machine has been designed to guide an AUV to its destination in a straight line trajectory. Figure 4.7 shows the implemented finite state machine with six states. Each AUV attempts to follow the shortest path to reach its destination on the seabed so that the swarm is deployed in a mesh-like geometry on the seabed for seismic imaging. Figure 4.8 shows the ground-truth trajectories of 5 AUVs from their home position on the sea surface to their seabed destinations. The IMU parameters and the destination coordinates are being assigned in the initialisation state. The AUV's target yaw and pitch are then computed based on its current position (i.e., deployment position) and its assigned destination. The required rotations are then executed by applying different speeds on the thrusters till $\Delta\Theta$ and $\Delta\Phi$ are within a predefined tolerance using a simple proportional-derivative controller where $\Delta\Theta$ and $\Delta\Phi$ are the errors in the AUV's yaw and pitch headings respectively. Once $\Delta\Theta$ and $\Delta\Phi$ are within the predefined tolerance, the AUV surges to its destination and updates its target yaw and pitch whenever it receives an external navigation aid i.e., USBL or NB localisation aid.

A typical industrial grade IMU is considered in our simulation [53] for DR navigation. Webots robotic simulator provides a 9-axis IMU sensor model to return the AUV's roll, pitch and yaw angles with respect to the world coordinate, the AUV's acceleration

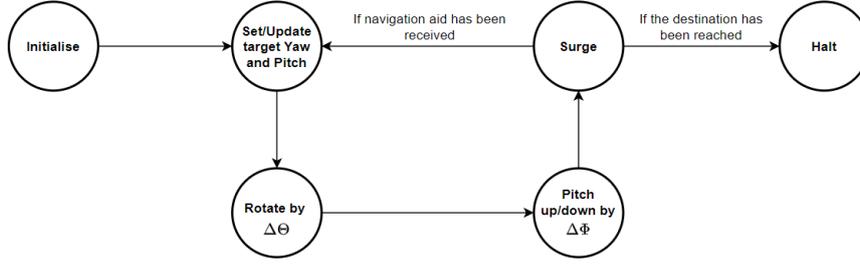


Figure 4.7: Finite state machine to guide the AUV to its pre-assigned destination where $\Delta\Theta$ and $\Delta\Phi$ are the errors in the AUV’s yaw and pitch respectively.

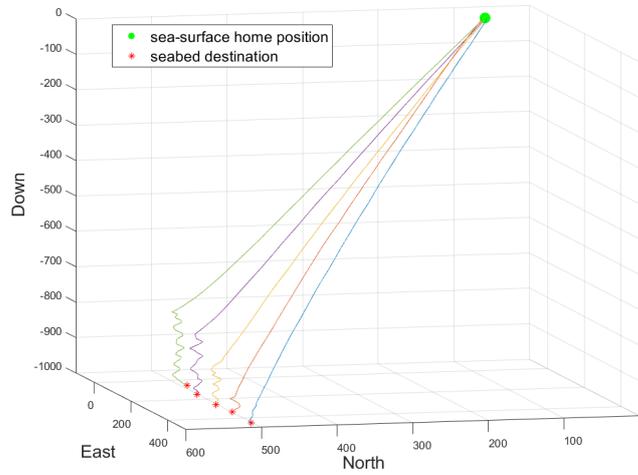


Figure 4.8: An example deployment scenario of 5 AUVs’ ground-truth trajectories.

and angular velocity can also be obtained [136]. Webots utilises a lookup table to add error models to the IMU measurements or to match Webots IMU’s output with device specific output; this look-up table can be very hard to generate given the IMU’s model complexity. However, MATLAB Navigation toolbox seamlessly models all the intricacies of a 9-axis IMU with a predefined properties such as velocity and angle random walks, bias instability, axis misalignment and constant bias. We therefore generate the angular velocity ${}^i\omega_{b,t}$, the acceleration ${}^b f_t$ and the local earth magnetic field ${}^n m$ of Webots inertial unit sensor with no noise or biases added and feed them into MATLAB Navigation toolbox for 9-axis IMU sensor modelling. Figure 4.9 shows a diagram of the procedure we followed to model a 9-axis IMU. Table 4.2 summarises the properties of the modelled IMU i.e., the predefined IMU properties and noise parameters.

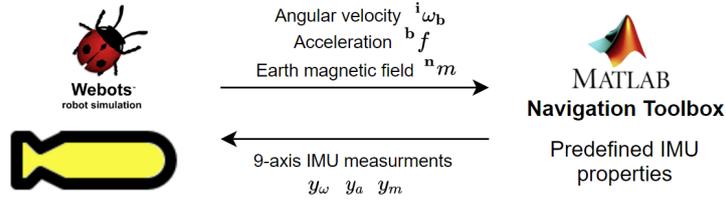


Figure 4.9: Angular velocity, acceleration and local magnetic field of the earth are generated as ground truth readings at each time instant of each IMU that is modeled in an underwater environment in Webots simulator. Given the ground truth readings and the IMU properties, a realistic 9-axis IMU is modelled on MATLAB navigation toolbox.

Parameter	Value
Accelerometer Resolution	60.958 μg
Accelerometer Constant Bias	14 μg
Accelerometer Noise Density	57 $\mu\text{g}/\sqrt{\text{Hz}}$
Gyroscope Resolution	0.0625 $^\circ$
Gyroscope Constant Bias	7 $^\circ/\text{hour}$
Gyroscope Noise Density	0.15 $^\circ/\sqrt{\text{hour}}$
Magnetometer Resolution	1 mGauss
Magnetometer Constant Bias	1.5 mGauss
Magnetometer Noise Density	3 mGauss

Table 4.2: Ellipse 2 micro IMU properties [53]

4.4 Summary

In summary, underwater swarm robotics research can be developed and validated on a realistic simulation platform due to the high cost of maritime assets. A high fidelity simulation platform that considers AUVs sensors, environment hydrodynamics and acoustic communications characteristics is implemented. Webots robotic simulator is employed for AUVs physics and fluid simulation for its design agility in swarm robotics and an industrial grade 9-axis IMU is modelled using MATLAB Navigation toolbox. The Underwater Networks Project UnetStack is utilised for underwater acoustic communications simulation for its easiness in network configurations and management and cross-layer sharing feature. Field experiments were conducted to validate UnetStack for packet loss in an underwater ACOMMS channel. The simulated and real world underwater acoustic communications packet loss were equal to 14.4% and 6.2% respectively due to the high simulated noise level and lake idleness where the experiments took place.

Chapter 5

A Fuzzy Localisation Framework for Underwater Robotic Swarms

This chapter proposes a novel underwater localisation method to dynamically fuse multiple position estimates of an [AUV](#) for better localisation accuracy along the whole trajectory using fuzzy decision support system. A simple proof-of-concept simulation and high fidelity physics-based simulation have been conducted to validate the proposed fuzzy-based algorithm. The results of the proposed algorithm are discussed and compared to other underwater localisation methods in the literature. The proposed fuzzy-based algorithm and its simulation results are published in [\[26, 27\]](#).

5.1 Introduction

Underwater localisation has recently attracted researchers interest, due to the wide variety of offshore applications that require this technology. A large number of underwater localisation algorithms have been proposed. Localisation algorithms can be classified into three categories based on operation depth, namely near surface, mid-water, and near seabed localisation algorithms. Underwater vehicles are either umbilically connected to sea surface vehicle [\[43\]](#) or periodically rise and dive [\[91\]](#) so that [AUVs](#) location can be obtained by means of GPS when they are close to sea surface. [Simultaneous Localisation and Mapping \(SLAM\)](#) based on seafloor landscape features [\[154\]](#) and [DVL](#) bottom track based on seafloor relative velocity of underwater vehicle [\[18\]](#) are commonly employed to

achieve near seafloor **AUVs** localisation. **DVL** water track or Acoustic Doppler Current Profiler can be utilised as an aiding sensor in **IMU**-based **DR** navigation in mid-water column [21]. **ToF** acoustic localisation such as **LBL**, **SBL**, and **USBL** are commonly used as navigation aids in **IMU**-based **DR** navigation. **USBL** does not require any artificial landmarks on the seabed and a single **Unmanned Surface Vehicle (USV)** or ship is deployed for operation. **USBL** is more flexible and has less limitations than those in **LBL** and **SBL**; hence it is the most commonly adopted method in the industry. A commercially available **USBL** system can achieve localisation accuracy of around 0.13-0.27% of slant range and can support tracking of four to ten underwater targets in a single **TDMA** frame of 1 second [66]. The localisation of each individual sensor node in a mobile **UWSN** (i.e., **AUVs** swarm) relies predominately on its proprioceptive sensors (i.e., **IMU**) but **IMU**-based navigation is prone to drift. Therefore, external navigation aids that are made available by exteroceptive sensors (e.g., **USBL** transponder, **DVL**) are usually required. Each localisation method such that **USBL** or **DVL** has its own merits and limitations which make each method suits certain operating conditions.

In this chapter, an underwater localisation method to dynamically fuse multiple localisation estimates of an **AUV** using fuzzy decision support system is proposed. We suffice, at first, with a simple proof-of-concept simulation and localisation methods' error characteristics to validate our approach. Thereafter, sophisticated simulation is implemented in which three underwater localisation methods are considered to validate our approach, namely **IMU**-based **DR**, trilateration/multilateration [155] and **USBL**. An industrial grade **IMU** is modelled, trilateration localisation methods are implemented and the co-simulation platform explained in Chapter 4 are employed to validate the proposed fuzzy-based localisation approach. The proposed cooperative navigation framework organises the cooperation among swarm nodes as exteroceptive sensors navigation aids of each **AUV** are controlled by the implemented fuzzy rules. The proposed localisation framework utilises fuzzy logic for information fusion which has inherent advantages over the common **EKF**-based fusion such as design simplicity and flexibility. It is straightforward to capture human expert knowledge in characteristics of localisation methods involved using fuzzy logic. In addition, new knowledge can be acquired and represented in additional fuzzy rules or modifying rules in the proposed localisation framework. In contrast, **EKF**-based fusion requires dynamic motion models, Gaussian error models and major changes should be made to accommodate changes in the motion models in case of integrating additional sensory information. Computational efficiency is another feature that can be gained when adopting fuzzy logic for information fusion over **EKF**-based fusion. Matrix operations in the **EKF** involve matrix inverse and multiplication that is

computationally expensive for small and dense matrices (i.e., computational complexity of $O(n^3)$ for $n \times n$ matrices) but fuzzy inference, on the other hand, is easy to parallelise in rule evaluation [156]. In addition, fuzzy logic chips for embedded hardware are available for optimised memory demand and computation speed of fuzzy controllers [157]. The proposed method can be easily extended to accommodate some other newly developed localisation methods by expanding the fuzzy rule base and thus better scalability is obtained with increasing swarm size. The proposed method's localisation performance is compared to USBL-aided DR navigation [63] with round-robin scheduling [158] in extensive simulation with swarm size of 150.

The remainder of this chapter is organised as follows. Fuzzy logic in the navigation context and the proposed localisation framework are illustrated in section 5.2. The effectiveness of the proposed fuzzy-based method is demonstrated by simulating a common deployment scenario of AUVs in a simple proof-of-concept simulation in section 5.3 as well as in a high fidelity physics-based simulation in section 5.4. Finally, section 5.5 concludes this chapter.

5.2 Fuzzy-based localisation

Fuzzy Logic [159] is efficient at information fusion, especially for uncertain or conflicting information. On the contrary, the family of probabilistic filters [23] such as EKF or Unscented Kalman Filter (UKF) requires Gaussian modelling of error uncertainty. Gaussian modelling of error uncertainty requires extensive experiments to estimate the parameters and shape of probability distribution of error and the error may not follow Gaussian distribution in reality. Information fusion problem in fuzzy logic is transformed into simple input-output mapping. Many other systems such as neural networks, differential equations and lookup tables can be used for input-output mapping but fuzzy logic is still the most convenient method [156, 160]. Fuzzy logic is adopted in robotic swarms literature in order to mitigate various challenges due to its simplicity. The authors in [135] adopted a fuzzy logic-based voter to decide on an AUV's health (i.e., the AUV is either fit for the task or must go back to the base) in swarm behaviour simulation of multiple AUVs. Fuzzy logic can be seen as a method for computing with words instead of numbers and thus it is convenient to use as it imitates human reasoning processes. It is the codification of common sense and therefore a lot closer to human intuition than any other input-output mapping method.

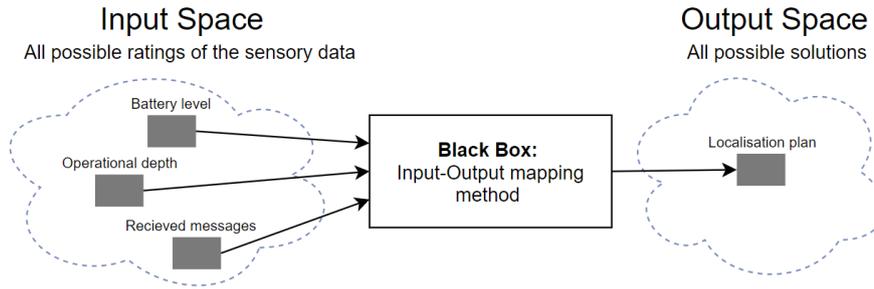


Figure 5.1: An example of input-output mapping for the localisation problem: “Given three variable inputs what the localisation plan should be?”

Figure 5.1 shows an example of input-output mapping for the underwater localisation problem based on three linguistic variable inputs i.e., the operational depth, battery level and received messages. Each variable input is associated with several ordinal or categorical linguistic concepts with vague boundaries. For example, the operational depth can be described by different adjectives, like shallow, deep and very-deep. These adjectives are the labels of linguistic concepts modelled by *fuzzy sets* [159]. In contrast, the classical classification in *crisp sets* would either include an input in one set or exclude it. A *membership function* has to be associated with each fuzzy set to map an input of the entire input space, *universe of discourse*, to its membership value between 0 and 1 [159]. If-then rules are the main constituent of fuzzy logic systems, they combine the antecedent (if-part) with the consequent (then-part) [156]. All variable inputs in the antecedent are resolved to a graded membership between 0 and 1 (*fuzzification*) but if the antecedent has multiple parts, fuzzy logic operators such as AND (t-norm) and OR (t-conorm) are applied to resolve the antecedent to a single number between 0 and 1 which is the degree of support for the rule (*application*) [160]. The resultant degree of support of the rule is then used to shape the output fuzzy set (*implication*) [160]. All if-then rules are being processed in parallel so that the order of the rules does not make a difference. Each if-then rule results in a fuzzy set, all consequent fuzzy sets are combined to give a single resultant fuzzy set by taking the maximum or any other methods such as sum of the rule output sets (*aggregation*) [159]. Given the aggregation process output fuzzy set and the output space of the output fuzzy set, the resultant aggregate output fuzzy set is converted into a single number (*defuzzification*) [159].

To summarise, fuzzy inference process comprises five steps:

- Fuzzification of the input variables.

- Application of the fuzzy operator.
- Implication from the antecedent to the consequent.
- Aggregation of the output fuzzy set of each rule across all the rules.
- Defuzzification the aggregated output fuzzy set.

One example of an IF-THEN fuzzy linguistic rule considering the localisation problem:

IF Operational Depth is Shallow AND Battery Level is High THEN
Localisation Method is L.

Where Operational Depth and Battery Level are two variable inputs, Shallow and High are two fuzzy sets to describe the inputs, both parts of the antecedent are combined by AND logical operator and Localisation Method is the variable output where L is its fuzzy set that refers to a particular localisation method and to be reshaped based on the rule degree of support.

Assume a swarm of AUVs is launched from known positions on the sea surface and a USBL system, which can navigationally aid only a limited number of AUVs in each of its TDMA frame. Each AUV is equipped with long/medium ACOMMS modem (USBL transponder), short range ACOMMS modem for intra-swarm communication, 9-axis IMU and depth sensor. Assume we have n underwater localisation methods. Each method can localise an underwater mobile sensor node with best accuracy under certain operational conditions e.g., AUV's operational depth and the reception of acoustic messages. This approach allows each AUV to either select a single localisation method or to fuse two or more localisation methods' estimates to improve localisation accuracy by increasing localisation aid updates along the whole trajectory based on fuzzy inference system. Mamdani fuzzy logic [161] is adopted for the localisation problem due to its intuition and well-suitability to human input which can be easily captured using if-then rule construct. Moreover, the impreciseness of human expert knowledge can be modelled and processed by fuzzy inference.

Figure 5.2 illustrates four examples of decision support elements (input variables) for this approach, namely operation depth \mathcal{D} , USBL availability \mathcal{U} , AUV's battery level \mathcal{B} and number of localisation aids received within a predefined time window from neighbouring AUVs \mathcal{G} . Input fuzzy sets are determined intuitively based on the features of the variable inputs e.g., a USBL localisation update or fix is either received by the AUV's transponder or not received. The final location estimate could be the output of either a single location

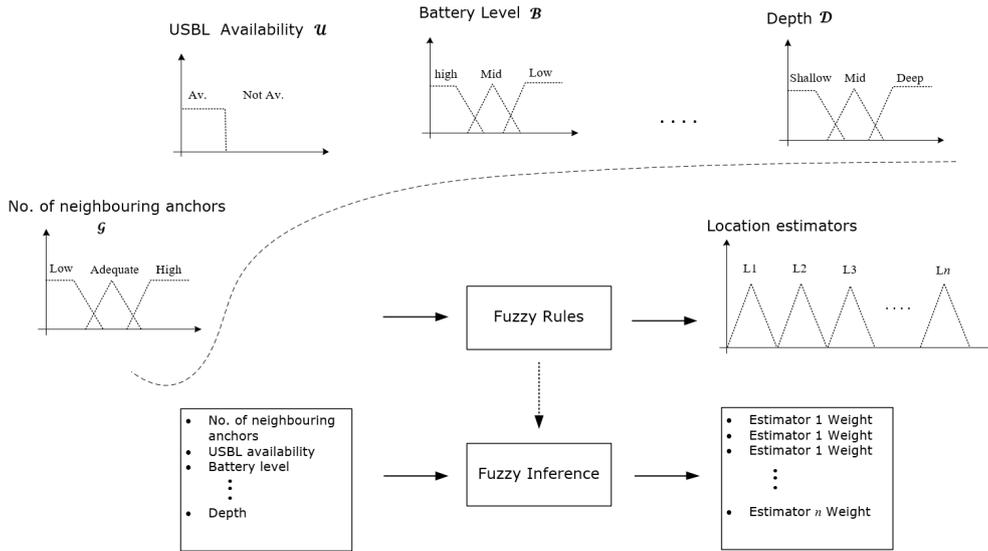


Figure 5.2: Best Suitable Localisation Algorithm approach (Decision-making)

estimator or a weighted combination between two or more location estimators based on the normalised sum of the firing strength in each of the output fuzzy set. Each of the considered localisation method is represented by a disjoint triangular fuzzy set over the universe of discourse so that each localisation method contribution in the final localisation plan can be easily obtained by its firing strength. It is worth mentioning that Chame et al. in [162, 163] have adopted our approach in sensor fusion in the context of underwater navigation. A generic policy that evaluates redundant navigation information of different estimators to obtain a navigation fusion plan was developed in [162, 163]. The fusion process in [162, 163] was handled by utilising a neural network.

5.3 Proof Of Concept Simulation

A simplified proof-of-concept numerical simulation based on localisation error models of four different underwater location estimator approaches was first conducted to show the proposed method is feasible. Four different underwater location estimators are assumed to validate our approach in this section and they are: USBL [66], LLNP [79], SLMP [80] and Inertial Navigation System aided by Doppler Velocity Log (INS/DVL) [17].

5.3.1 Localisation Error Models

Due to the lack of technical details of the sensor involved in all localisation methods and the operating environment characteristics, stochastic localisation error models are constructed based on literature for initial validation of the proposed **Best Suitable Localisation Algorithm (BSLA)** scheme. These error models are applied to emulate the localisation error generated by the corresponding localisation methods when sensor nodes are traversing underwater. Advanced **USBL** localisation system can localise up to 10 underwater targets in a single **TDMA** frame of one second with accuracy of 0.27% 1 Drms of slant range [66]. Figure 5.3 shows the relationship between the total error in meter 1 Drms of an accurate **USBL** system and the depth of an underwater target (blue curve). Figure 5.3 shows that in 1000 m depth 63% (1Drms) of **USBL** localisation total errors are within 2.7 m radius. Localisation accuracy of another assumed **USBL** system is shown in Figure 5.3 (red curve) which follows the same profile of Ranger **USBL** but with offset error of 7.3 m. The red curve suggests 63% of **USBL** fixes are within 10 m radius in 1000 m depth. We assume that the localisation error of a **USBL** system follows Gaussian distribution. The localisation error in **USBL** $E_U \sim \mathcal{N}(\mu, \sigma^2)$ where $\mu = 2.7$ m and 10 m for Ranger **USBL** and ordinary **USBL**, respectively and σ is fitted to the curves shown in Figure 5.3.

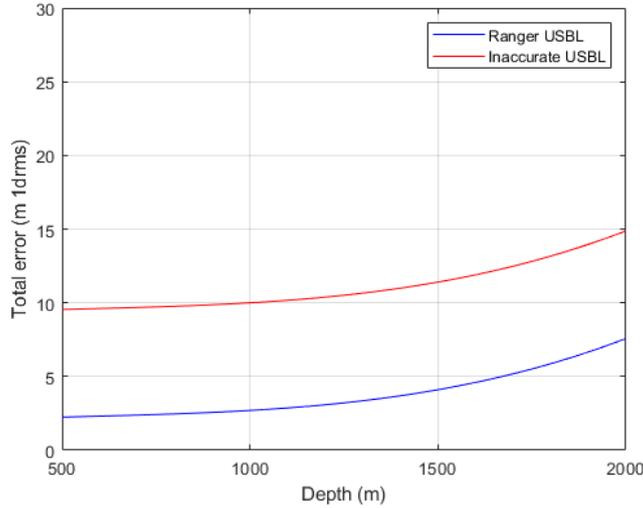


Figure 5.3: The localisation error of Ranger **USBL** and ordinary **USBL** with water depth

We assume that error characteristics of **LLNP** and **SLMP** (discussed in section 3.3) are normally distributed over the aforementioned error data points of **LLNP** and **SLMP**

in Figure 3.3 where $E_L \sim \mathcal{N}(\mu, \sigma^2)$: μ is the localisation error depicted in Figure 3.3 and $\sigma = 0.02\mu$. This assumption has been made based on existing underwater distance measurement technologies [164].

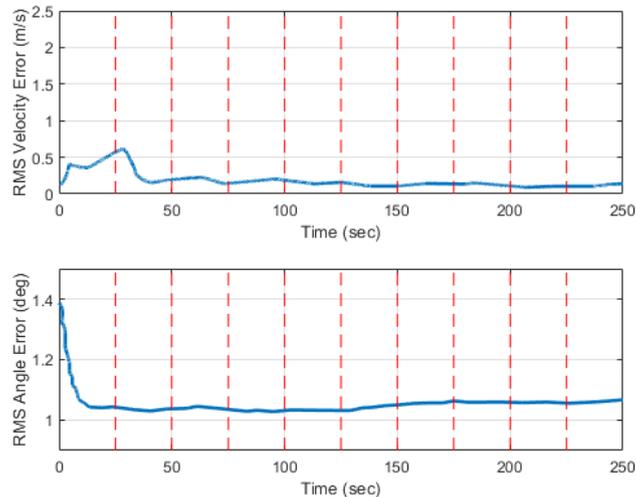


Figure 5.4: Root mean square of the velocity and the attitude errors using tightly coupled DVL/INS [17].

Authors in [17] presented simulation results of an AUV navigation performance obtained using different data fusion methods of INS aided by DVL. Figure 5.4 presents the root mean square errors of the velocity vector and the attitude error of an AUV navigates with a velocity of 2 m/s along a classic lawn mower trajectory using a tightly coupled INS/DVL [17]. It can be noticed that the velocity and attitude errors in Figure 5.4 are not consistent over time so that we assume that the velocity and attitude errors follow Gaussian distribution over a sliding time window of 25 seconds.

5.3.2 Simulation Settings

Suppose there are 50 mobile sensor nodes each of them equipped with a depth sensor, a 9-axis IMU, a USBL transponder, and a 300 kHz DVL (has a range of around 200 m). Their home position is somewhere close to sea surface and need to be deployed on pre-determined seafloor positions at a depth of around 3 km. Assume we have a USBL localisation system, which can only track one underwater node in its TDMA frame of one second with low-accuracy, hull mounted on a surface vessel. Assume that $n = 4$ (i.e., four localisation methods are considered) where L_1 represents USBL localisation system

with total localisation error suggested in the previous section in Figure 5.3 (ordinary USBL), L_2 represents LLNP [79], L_3 represents SLMP [80] and L_4 represents DVL/INS [17].

The following Figure 5.5 shows the implemented fuzzy antecedents, their types and limits over the universe of discourse. Each localisation algorithm is represented by a disjoint triangular fuzzy set as shown in Figure 5.5. This output fuzzy sets representation allows us to determine the contribution (weight) of each location estimate in the final localisation plan. Figure 5.5 shows equal contributions of L_1 and L_3 in the final localisation plan so that the final location estimate would be $0.5L_1 + 0.5L_3$. The fuzzy rule base we employed in our simulation is shown in Appendix A.1.

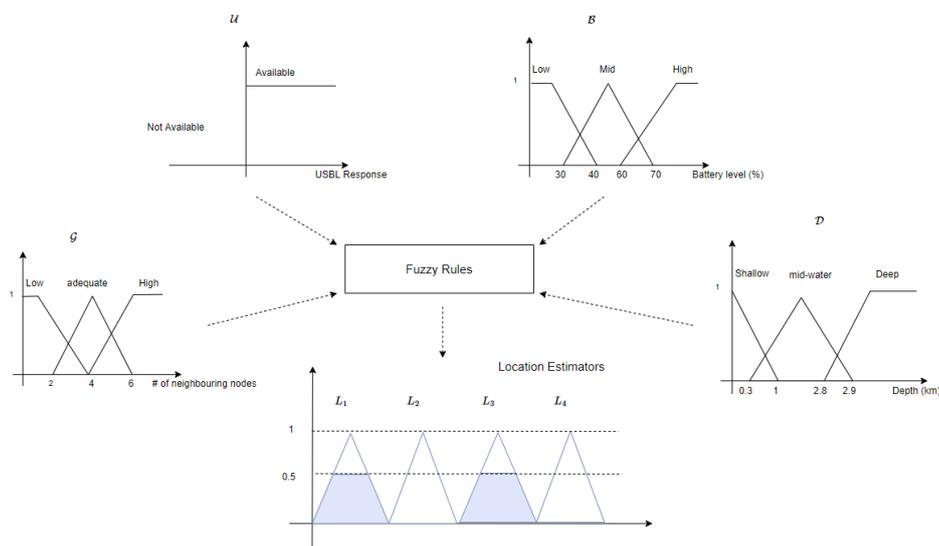


Figure 5.5: The adopted fuzzy antecedents, their types and limits over the universe of discourse in our poof-of-concept simulation. The aggregated fuzzy output shows an example of a final localisation plan of $(0.5L_1 + 0.5L_3)$

A simple two-dimensional (i.e., North-Down) dynamic model has been assumed to govern the AUV's mobility. Its governing equations are:

$$\dot{x} = \nu \sin(\Theta) \quad (5.1)$$

$$\dot{y} = \nu \cos(\Theta) \quad (5.2)$$

$$\dot{\Theta} = \nu \mathcal{K} \quad (5.3)$$

where $(\mathcal{X}, \mathcal{Y})$ are AUV's position coordinates, Θ is AUV's heading, \mathcal{V} is the commanded forward speed and \mathcal{K} is the commanded turn curvature. Table 5.1 summarises simulation and navigation parameters used to produce the results in this section.

Parameter	Value
Time step	1 s
AUV velocity	5 m/s
DVL range	200 m
Seafloor depth	3000 m
AUV's communication range	20 m
Anchor nodes density	50/100 m ²
USBL update rate	1 s

Table 5.1: Simulation parameters

Anchor nodes are randomly deployed with a density of 50 nodes per 100 m² and assumed to be perfectly localised. Anchor node density and AUV's communication range parameters are intended to be identical to those assumed in both LLNP and SLMP so that the use of the presented localisation error of LLNP and SLMP in Figure 3.3 can be justified. Figure 5.6 shows an example of a predefined path of a mobile sensor node descending from its home position (40 m below sea surface) to its destination on the seafloor. All AUVs are launched from the same home position and passing through the same Way Point 1 shown in Figure 5.6. Way Point 1 is at a depth where the bottom track DVL can successfully work.

5.3.3 Results and Analysis

Localisation performance of each method including the proposed fuzzy-based BSLA is obtained through five trial trajectories. Each of them has a different destination and is presented in terms of mean errors and standard deviation of the estimated positions along the trajectory. The unavailability of each localisation method along the trajectory is also evaluated as it represents the localisation update an AUV receives when relying on a certain localisation method. The performance of each localisation method in localising a single underwater node is shown in Figure 5.7. It is observed from Figure 5.7(a) and (b) that LLNP and SLMP have the most accurate localisation estimates but less than 30% of AUV's locations were estimated using either LLNP or SLMP, as shown in

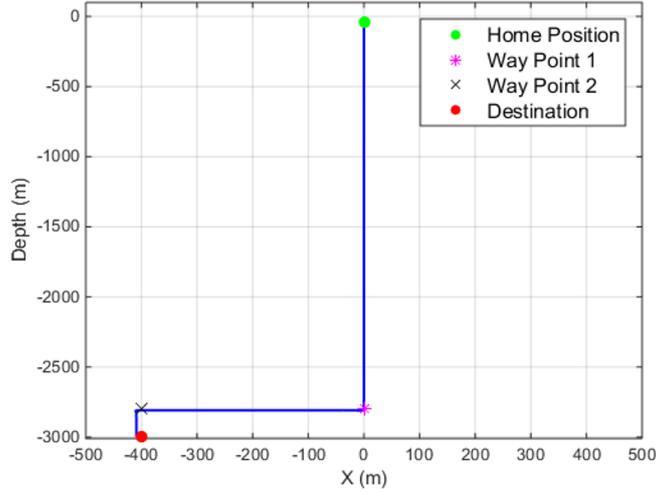


Figure 5.6: An example of a descending mobile sensor node trajectory

Figure 5.7(c). As it was expected, the USBL was able to estimate node’s position in high update rate with high localisation error. This is due to the high localisation error in the simulated USBL as a high offset error is added to commercially available USBL system [66]. On the other hand, node’s positions were estimated using BSLA approach have lower localisation error than that in the USBL and higher than that in LLNP and SLMP but it was available along the whole navigated trajectory.

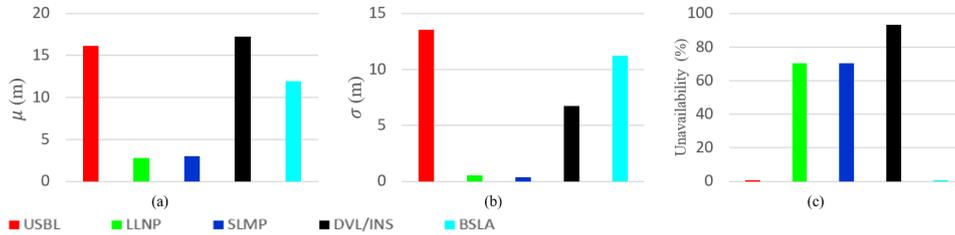


Figure 5.7: Performance of five localisation methods in localising a single underwater node (a) mean error (b) error standard deviation (c) localisation approach unavailability along the whole predefined trajectory

Figure 5.8 shows a comparison among all localisation methods. Each performance element is presented on a radius line and normalised to its highest value so that the least accurate and the most unavailable localisation approach would be reflected as an equilateral circumscribed triangle. A perfect localisation method (i.e., zero mean error, zero standard deviation of error and always available) would be reflected as a point at the

centre in Figure 5.8. It is clear that LLNP and SLMP have almost identical performance whereas DVL/INS has the highest mean error and the highest unavailability which was expected since the DVL does not work unless the node (AUV) is close to the seafloor.

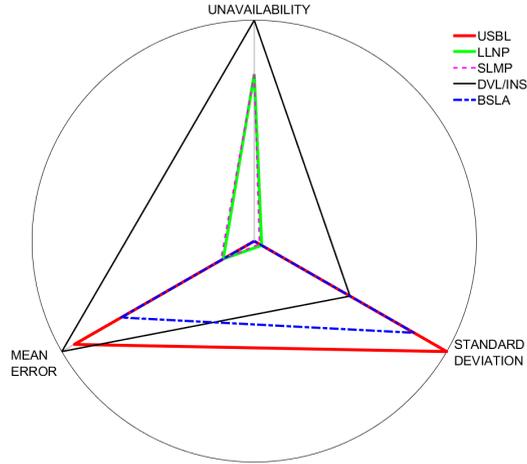


Figure 5.8: Normalised performance of five localisation approaches in localising a single underwater node

In the second set of simulations, positions of three identically equipped underwater nodes are estimated using the five localisation methods. The localisation performance in terms of mean errors, standard deviation and unavailability are depicted in Figure 5.9. Figure 5.10 compares the three performance elements of the five localisation methods. The proposed fuzzy-based approach - i.e. BSLA - notably improved localisation accuracy of around 23-30% compared to the USBL and DVL/INS. It is discernible that BSLA has improved localisation accuracy and was the best approach in term of availability.

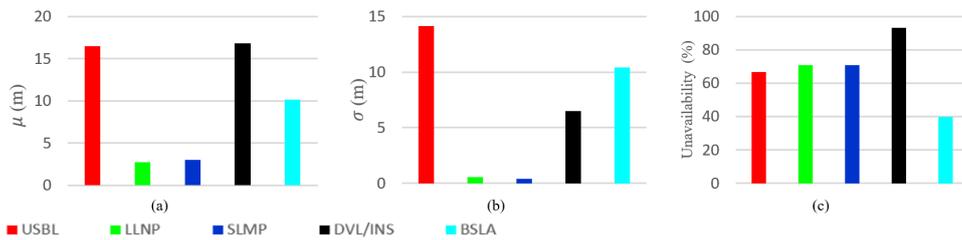


Figure 5.9: Performance of five localisation approaches in simultaneously localising three underwater nodes (a) mean error (b) error standard deviation (c) localisation approach unavailability along the whole navigated trajectory

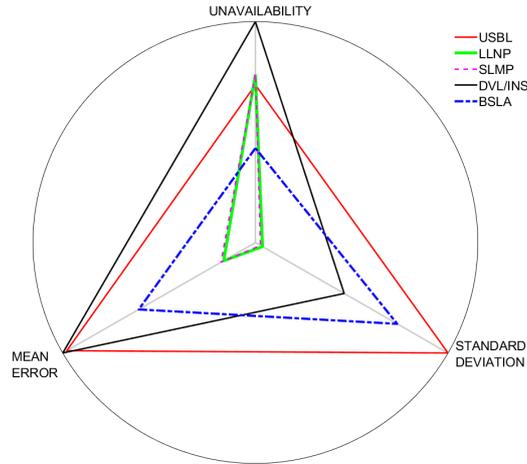


Figure 5.10: Normalised performance of five localisation approaches in simultaneously localising three underwater nodes.

The proposed fuzzy-based approach for underwater mobile sensor nodes localisation improved the localisation performance elements by dynamically fusing multiple location estimates of available localisation methods based on fuzzy decision support system. Results show that the presented approach improves both the localisation accuracy of around 23-40% and the availability of around 70% when three underwater nodes are simultaneously localised.

In this simple numerical simulation, error characteristics of different localisation approaches were considered instead of implementing them, precisely localised nodes have been considered with high density for LLNP and SLMP localisation methods and the underwater acoustic channel packet loss has not been considered.

5.4 Physics-based Simulation

In this section, a sophisticated simulation platform is implemented in which three underwater localisation methods are considered to validate our approach, namely IMU-based DR, trilateration/multilateration [155] and USBL. An industrial grade IMU is modelled and trilateration localisation methods are implemented. The physics-based co-simulation platform implemented in Chapter 4 that considers underwater environments hydrodynamics and underwater acoustic communications characteristics is employed to validate

the proposed localisation framework.

A physics-based high fidelity robotic simulator Webots [136] is employed to simulate AUV dynamics and hydrodynamic properties of underwater environments i.e., density, viscosity and stream velocity to generate external static and dynamic forces. The static and dynamic forces are then applied on the AUVs' body and corresponding thrust power is generated to guide each AUV to its destination. The Underwater Network Project for acoustic simulation (i.e., UnetStack) [142] is employed for underwater acoustic communications simulation in which the characteristics of the employed channel, modems, MAC protocol and time synchronisation are all considered. The proposed cooperative navigation framework organises the cooperation among swarm nodes as exteroceptive sensors navigation aids of each AUV are regulated by the implemented fuzzy rules. The proposed localisation framework utilises fuzzy logic for information fusion which has inherent advantages over EKF-based fusion such as design simplicity and flexibility. It is straightforward to capture human expert knowledge in characteristics of localisation methods involved using fuzzy logic. In addition, new knowledge can be acquired and represented in additional fuzzy rules or modifying rules in the proposed localisation framework. In contrast, EKF-based fusion requires dynamic motion models, Gaussian error models and major changes should be made to accommodate changes in the motion models in case of integrating additional sensory information. The proposed method can be easily extended to accommodate some other newly developed localisation methods by expanding the fuzzy rule base and thus better scalability is obtained with increasing swarm size. The proposed method's localisation performance is compared to USBL-aided DR navigation [63] with round-robin scheduling [158] in extensive simulation with swarm size of 150.

5.4.1 Implementation

Assume each AUV is equipped with long/medium ACOMMS modem (USBL transponder) working at medium frequency band i.e., 20-40 kHz, short range ACOMMS modem working at high frequency band (i.e., 100-180 kHz) for intra-swarm communication, 9-axis IMU, depth sensor and Chip Scale Atomic Clock (CSAC) for clock-synchronisation of the AUVs [165]. We assume that either FDMA MAC protocol [132] is utilised to separate intra-swarm communication from USBL communication and each channel has its own TDMA schedule to broadcast acoustic messages. Range measurements are acquired by means of One-Way Travel Time (OWTT) [166]. Timestamps for range measurements

are subject to Gaussian additive noise of zero mean and standard deviation of 1.2 ms [167] which corresponds to error standard deviation of 1.8 m in range measurements given that the average speed of underwater acoustic waves is 1500 m/s. The three-dimensional localisation problem can be converted into its 2D counterpart via orthogonal projection [94] as all AUVs are equipped with pressure sensors for accurate depth estimation. Due to the severely limited bandwidth and high latency in ACOMMS, only a subset of the swarm (i.e., NB) is made capable of broadcasting navigation aids within a short period of time once any reliable location estimator (e.g., USBL or trilateration) is dominating the final localisation plan. NB AUVs are considered as references for neighbouring AUVs localisation. Five variable inputs have been considered to determine the weights of the different underwater location estimators in the final localisation plan and they are: USBL availability \mathcal{U} , the number of localisation aids received within a predefined time window from NBs \mathcal{G} , battery level \mathcal{B} , operational depth \mathcal{D} and DR time \mathcal{R} . The USBL availability determines whether a USBL fix from the USBL transceivers (on the sea-surface) has been received or not. DR time represents the time period that the AUV has been relying on IMU-based DR for navigation and it resets to zero once a USBL fix or a trilateration/multilateration estimate has been fused. Inputs fuzzy sets have been determined intuitively by human expert based on its features e.g., a USBL fix is either received by the AUV's transponder or not received and thus the USBL availability \mathcal{U} is represented by crisp sets over the universe of discourse i.e., available or unavailable. In this implementation, the number of localisation aids received from NBs is either enough or not enough to perform trilateration i.e., three or more localisation aids received from recent USBL-localised NB AUVs. Although localisation by NB aids can be performed with less than three measurements (i.e., solutions are eliminated based on the AUV's direction of movement; see [116] for more details), we only consider three or more NB aids to reduce the localisation process's uncertainty.

Three localisation methods (i.e., location estimators) have been considered in our implementation, namely IMU-based DR, USBL and trilateration/multilateration and denoted by L_1 , L_2 and L_3 respectively. The final location estimate could be the output of either a single location estimator or a weighted combination between two or more location estimators based on the normalised sum of the firing strength in each of the output fuzzy set. Each of the localisation methods is represented by a disjoint triangular fuzzy set over the universe of discourse so that the firing strength of each localisation method can be easily dissociated and normalised in the final localisation plan. Gaussian error models are assumed in this implementation. However, fuzzy-based localisation can handle

non-Gaussian modelled noise but Gaussian error models have been assumed to make fair comparison between the proposed fuzzy-based localisation and EKF-based localisation which requires Gaussian modelled noise. Figure 5.11 shows fuzzy inference system's variable inputs based on each AUV's on-board sensors and Figure 5.12 shows the variable inputs and their fuzzy/crisp sets, their types and limits and the three underwater location estimators used in this implementation. An example of a final localisation plan of a combination between L_1 and L_3 is shown in Figure 5.12 where the final location estimate would be $(\frac{2}{3}L_1 + \frac{1}{3}L_3)$ as L_1 firing strength is as double as L_3 . Fuzzy set parameters were fine tuned in our simulation based on trial-and-error simulation but their initial values were estimated based on prior knowledge of each localisation method operating conditions e.g. USBL localisation error is low at shallow operational depths and battery consumption in trilateration localisation is relatively high. Therefore USBL localisation is most likely to be adopted (if available) at shallow operational depths and trilateration is most likely to be performed if battery level is high. The employed fuzzy rule base is shown in Appendix A.2. The proposed underwater swarm localisation al-

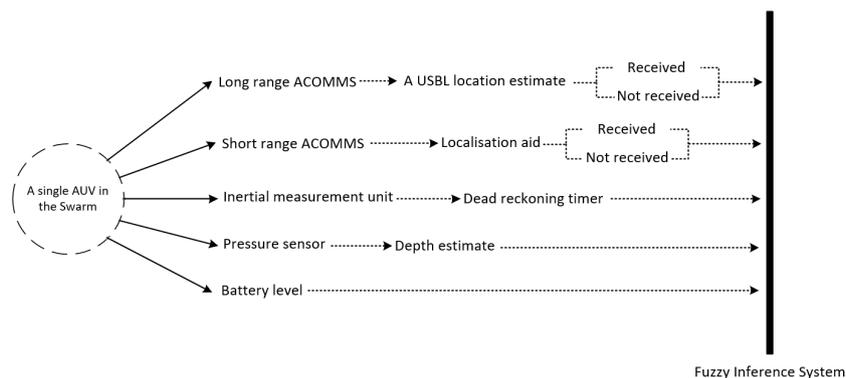


Figure 5.11: Fuzzy inference variable inputs of each AUV in a swarm of AUVs based on its on-board acoustic communication modems and sensors.

gorithm's performance is compared to USBL-aided DR navigation [63] with round-robin scheduling [158]. It is worth mentioning that in the proposed fuzzy-based localisation method if USBL transceivers receive localisation requests from more than the maximum that it can be aided in a single TDMA frame, the USBL employs round-robin scheduling [158] to respond to all AUV's localisation requests. A subset of the swarm (i.e., NBs) is configured to broadcast navigation aids within a predetermined period of time once any reliable location estimator (e.g., USBL or trilateration) is dominating the final localisation plan. The NB AUVs are a fixed set of AUVs and they are selected randomly in the

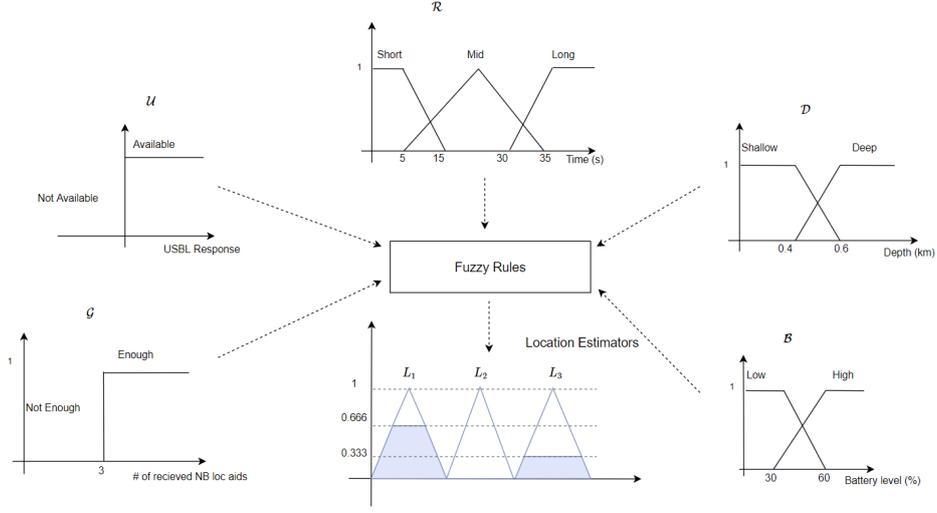


Figure 5.12: Fuzzy-based underwater localisation approach (Decision-making) with 5 variable inputs & their fuzzy/crisp sets and 3 underwater location estimators. The aggregated fuzzy output shows an example of a final localisation plan of $(\frac{2}{3}L_1 + \frac{1}{3}L_3)$.

swarm before deployment. Figure 5.13 shows an example of round-robin scheduling of a swarm of AUVs for USBL localisation where only 5 AUVs can be localised in a single TDMA frame ΔT . Each subset of 5 AUVs in Figure 5.13 can be navigationally re-aided by the USBL after the last node is aided e.g. the first subset of 5 AUVs is navigationally re-aided by the USBL every $k\Delta T$ where k is the swarm size divided by the maximum number of AUVs that can be aided in a single TDMA frame e.g., 5 AUVs. AUVs grouping for round-robin scheduling is performed based on first come, first served basis. In the proposed algorithm, when a NB receives a USBL fix, it broadcasts localisation aids of its North and East coordinates to its neighbouring AUVs. Similar to the MAC protocol adopted in GSM communications (i.e., a combination between both FDMA and TDMA [168]), we assume that either FDMA or CDMA MAC protocol [132] is utilised to separate intra-swarm communication from USBL communication and within both intra-swarm and USBL communications TDMA MAC protocol is adopted.

Figure 5.14 shows an example scenario of the proposed algorithm assuming that the USBL received localisation requests from all 11 AUVs shown in Figure 5.14. Given that the USBL can send navigation aids to only 5 AUVs in a single TDMA frame of ΔT , round-robin is then adopted for USBL navigation aids in three TDMA frames where at $time = t_o$ the first subset of 5 AUVs is aided by the USBL and at $time = t_o + \Delta T$ the second subset of 5 AUVs is aided. Subsequently the third subset of 1 AUV is aided. However in the proposed fuzzy-based localisation method while the second subset of

AUVs is aided by the **USBL** at $time = t_o + \Delta T$, **NBs** that have been aided by the **USBL** at $time = t_o$ broadcast localisation aid to their neighbouring **AUV** for trilateration.

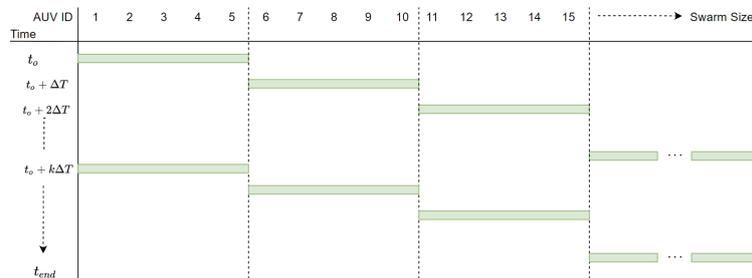


Figure 5.13: An Example of round-robin scheduling for **USBL** navigation aid (represented by the green bar) in a swarm of **AUVs**. The **USBL**, in this example, can only navigationally aid 5 **AUVs** in a single **TDMA** frame of ΔT .

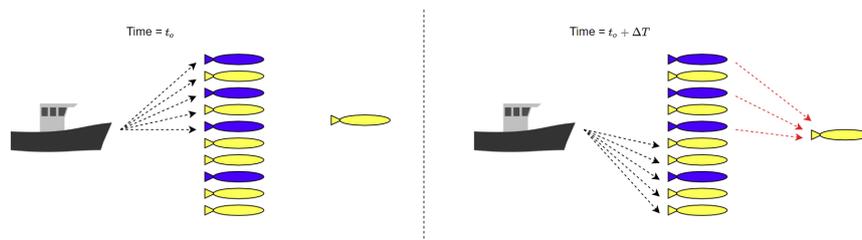


Figure 5.14: The proposed fuzzy-based localisation framework harnesses round-robin scheduling for **USBL** navigation aid assuming that all **AUVs** requested **USBL** navigation aid. The **USBL** can only navigationally aid 5 **AUVs** in a single **TDMA** frame of ΔT utilising low-frequency **ACOMMS** (in black arrows). **NBs** (in blue) broadcast localisation aid to their neighbouring **AUV** utilising high-frequency **ACOMMS** (in red arrows).

5.4.2 Simulation Scenario and Settings

A swarm of N **AUVs** is launched from known positions on the sea surface, each **AUV** has a unique ID associated with a specific seabed destination. We assume that an **AUV's** battery level follows a typical discharging profile of a lithium battery cell [169]. A hull-mounted **USBL** system on the sea surface can localise 10 **AUVs** in each of its **TDMA** frame [66].

A finite state machine with a proportional-derivative controller was designed to guide each **AUV** to its destination on the seabed through the shortest path i.e., straight line. The **AUV's** target yaw and pitch are updated once an external navigation aid is fused.

A swarm size of 50 to 150 AUVs has been simulated on Webots. Ten AUVs can be navigationally aided by the USBL in its TDMA frame of 1 second [66] so that a different subset of the swarm can be aided by the USBL every USBL TDMA frame. A fixed subset of the swarm (i.e., NBs) is configured to broadcast localisation aids to their neighbouring AUVs within a predefined period of time once they are externally aided by the USBL or NBs i.e., the weight of the USBL or trilateration estimators is greater than 0.8 in the final localisation plan of the proposed fuzzy-based localisation method. We consider a short period of time (i.e., 1 second), depends on DR accuracy, in which a NB can broadcast localisation aids once their final localisation plan is dominated by the USBL or trilateration. The same period of time (i.e., 1 second) is considered in the round-robin EKF-based localisation for the NBs to broadcast localisation aids once a USBL fix is received. Table 5.2 summarises the specifications of the modelled IMU. A subset of 10 AUVs can be navigationally aided by the USBL every 1 second (i.e., USBL TDMA frame length) but we instead delay the USBL update for 3 more seconds to account for larger swarm sizes (i.e., 3 times as large) in which the simulation becomes intractable using the available computing resources¹. Table 5.3 summarises simulation settings and parameters we have considered in our simulation and Table 5.4 lists the parameters of the simulated intra-swarm ACOMMS.

Parameter	Value
Accelerometer Resolution	60.958 μg
Accelerometer Constant Bias	14 μg
Accelerometer Noise Density	57 $\mu\text{g}/\sqrt{\text{Hz}}$
Gyroscope Resolution	0.0625 $^\circ$
Gyroscope Constant Bias	7 $^\circ/\text{hour}$
Gyroscope Noise Density	0.15 $^\circ/\sqrt{\text{hour}}$
Magnetometer Resolution	1 mGauss
Magnetometer Constant Bias	1.5 mGauss
Magnetometer Noise Density	3 mGauss

Table 5.2: Ellipse 2 micro IMU properties [53]

5.4.3 Results and Analysis

Swarm sizes of 50 to 150 AUVs are considered to validate and compare the proposed fuzzy-based localisation algorithm’s performance. The performance of each localisation

¹ Dedicated workstation with Intel Xeon Gold 6148 CPU @ 2.40 GHz 20 Cores, 192 GB RAM and Nvidia TITAN Xp 12 GB.

Parameter	Value
Swarm Size	50; 100; 150 AUVs
Simulation Time Step	100 ms
Clock-synchronisation error	1.2 ms $1-\sigma$
Seabed Depth	1000 m
Depth Sensor	2 Hz, 0.1 m $1-\sigma$ error
USBL Transponder Communication Range	6000 m
USBL Localisation Accuracy in 1000 m	2.7 m $1-\sigma$ error
Number of AUVs positioned by the USBL in a single TDMA frame	10 AUVs
USBL TDMA Frame length	1 s
USBL update rate	4 s
Number of NBs	10 AUVs
NBs broadcasting period	1 s

Table 5.3: Simulation parameters

Parameter	Value
Communication modem Frequency band	160 kHz
Communication data rate	50 kbit/s
Navigation aid length and duration	20 bytes; 3.2 ms
Navigation aid allocated TDMA time-slot length	20 ms
Noise level	60 dB
Water salinity	35 ppt
Water temperature	10 °C
Rician fading parameter	10
Fast fading	enabled

Table 5.4: Intra-swarm communication modem and channel parameters

method is compared based on mean localisation error and standard deviation of each AUV as well as of the entire swarm. Figure 5.15 shows the entire swarm mean localisation error and standard deviation in each simulation trial. Ten NBs are considered in both the proposed fuzzy-based localisation and EKF-based localisation. Round-robin EKF-based aiding at swarm size of 50 outperforms the proposed algorithm in the entire swarm mean localisation error by 29.8%. However the proposed fuzzy-based localisation algorithm outperforms Round-robin EKF-based aiding at swarm size of 100 and 150 AUVs by 13.25% and 16.53% respectively. The proposed fuzzy-based localisation algorithm greatly improves the entire swarm standard deviation by 35.17% at swarm size of 150 AUVs when compared to round-robin EKF-based method. One-tail two sample t -test is conducted to compare the localisation accuracy performance of each simulation trial in both round-robin EKF-based method and the proposed fuzzy-based method. Sufficient support of data is obtained at swarm size of 100 AUVs to reject the null hypothesis at

0.05 significance level with p-value of 0.025. Although the null hypothesis at swarm size of 150 AUVs is also rejected at 0.05 significance level with p-value that was expected to decrease, it has increased by insignificant margin of 0.007 to reach 0.032. It is believed that the null hypothesis will always be rejected at large swarm sizes (i.e., larger than 50 AUVs) and the p-value is expected to decrease when swarm size increases. Based on the one tail test, mean localisation error of the proposed fuzzy-based method is lower than that of the EKF-based method at significance level of 0.05. Therefore, mean localisation error of the proposed fuzzy-based method is lower than that of the EKF-based method when swarm size increases. Table A.1 in Appendix A.3 shows p-values, degree of freedom, t -statistics and critical values of each simulation trial to statistically compare the localisation accuracy of the proposed fuzzy-based method with the EKF-based method. Figure 5.15 shows that the break-even point for the proposed algorithm to outperform round-robin EKF-based aiding is at swarm size of around 80 AUVs.

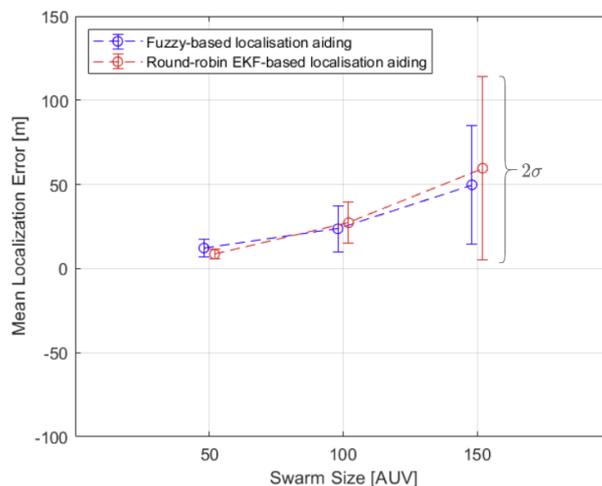


Figure 5.15: The entire swarm mean and standard deviation localisation error in both the proposed Fuzzy-based USBL/trilateration aided DR navigation (in blue) and round-robin EKF-based USBL/trilateration aided DR navigation (in red) at swarm size of 50, 100 and 150 AUVs. The error bar around the mean point represents 2σ standard deviation.

Figure 5.16 shows histograms of mean localisation error and standard deviation of each AUV in a swarm of 150 AUVs. The results in Figure 5.16 are depicted by computing the mean and standard deviation of each AUV along the followed trajectory from its home position on the sea surface to its destination on the seabed. It can be seen that the number of occurrences of standard deviation below 100 m is higher in the proposed

algorithm than in round-robin **EKF**-based localisation. There are 81 out of 150 **AUV**s achieved standard deviation below 100 m along their trajectories as opposed to 73 **AUV**s when round-robin **EKF**-based aiding is adopted. Localisation performance of an **AUV** in a swarm of 150 when the proposed fuzzy-based, round-robin-based and **DR**-only are shown in Figure 5.17. The **AUV**'s localisation performance in Figure 5.17 is aided by the **USBL** around the same time in both round-robin **EKF**-based and fuzzy-based aiding and that happened for two reasons. Firstly, the **AUV** has been exposed to the same environment settings when either of the localisation method is adopted. Secondly, the fuzzy rules are designed to prioritise a **USBL** fix whenever it is received. Figure 5.17

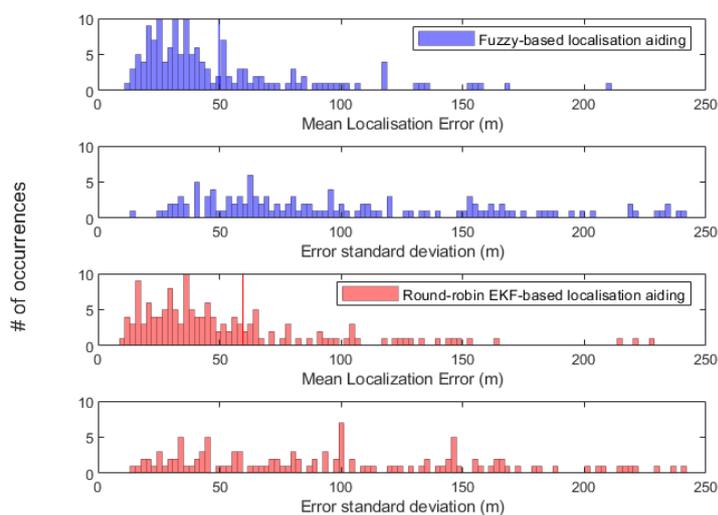


Figure 5.16: Histogram of the mean localisation error and standard deviation of each **AUV** of the entire swarm of 150 **AUV**s. The vertical lines represent the entire swarm mean localisation error which is 49.76 m and 59.62 m in fuzzy-based and round-robin **EKF**-based localisation respectively.

shows that the **AUV**'s localisation error quickly accumulates over 200 m in the first 100 seconds of the mission when there is no external navigation aid available as in **DR**-only (black curve). It can be observed that the **AUV** is being externally aided by the navigation beacons more frequently when the proposed fuzzy-based algorithm is adopted.

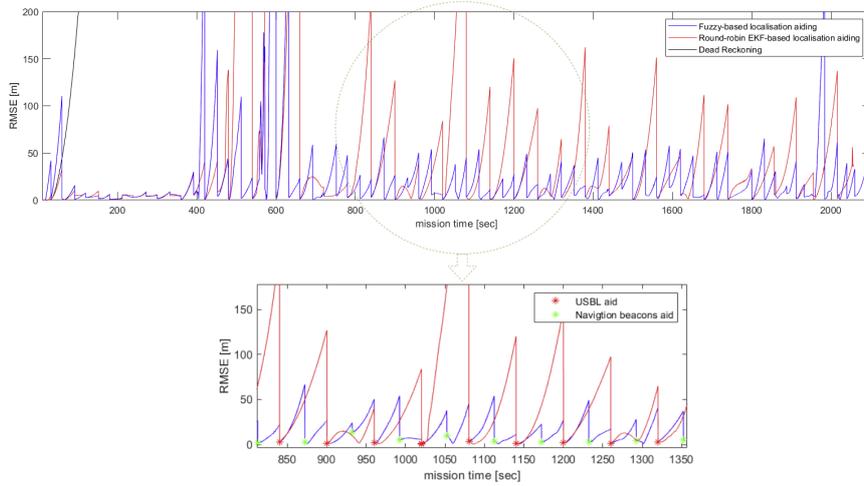


Figure 5.17: Instantaneous localisation performance of an AUV in a swarm of 150 when the proposed fuzzy-based, round-robin EKF-based and DR-only methods are adopted. A time window of 500 seconds shows the fused external navigation aid

5.5 Summary

In summary, design simplicity and flexibility are emphasised in the proposed localisation framework as new knowledge can be acquired and represented in additional fuzzy rules or modifying existent rules in the proposed localisation framework. In contrast, substantial efforts are needed when integrating new localisation methods in an existent EKF-based navigation for two reasons: Gaussian error modelling; and the entire filter re-implementation to expand the covariance matrix and the state vector. The proposed method can be easily extended to accommodate different localisation methods such as DVL aided navigation by expanding the fuzzy rule base and thus better scalability is obtained with increasing swarm size. The proposed fuzzy-based localisation method is computationally less expensive than the EKF-based localisation method, as matrix operations in the EKF that involve matrix inverse and multiplication are computationally expensive. Matrix operations in the EKF is not efficient to parallelise for small matrices (e.g., 6×6 matrices). However, fuzzy inference can be easily parallelised and fuzzy logic chips are available for optimised memory demand and computation speed implementation.

The proposed algorithm enhanced the overall localisation accuracy of the entire swarm by providing the AUVs with external navigation aids more frequently. The proposed

fuzzy-based aiding has improved the entire swarm mean localisation error and standard deviation by 16.53% and 35.17% respectively at swarm size of 150 AUVs when it was compared to round-robin EKF-based USBL/trilateration-aided DR navigation. The total number of AUVs that achieved standard deviation below 100 m along their trajectories has increased by around 10% when the proposed fuzzy-based aiding is adopted compared to round-robin EKF-based aiding. Simple fuzzy rules that capture human expert knowledge in underwater localisation methods by if-then rules are proposed and the impreciseness of expert knowledge is modelled and processed using fuzzy inference. The proposed algorithm performance is emphasised in large swarm sizes as it becomes nearly impossible to navigationally aid all AUVs by round-robin scheduling.

The proposed fuzzy-based localisation method does not provide information about localisation precision or uncertainty and it considers a fixed period of time for NB AUVs to broadcast navigation aids once they receive USBL localisation updates. Fuzzy set parameters and rules were fine tuned in our specific deployment scenario based on trial-and-error simulation but their initial values were estimated based on prior knowledge of each localisation method error characteristics. Tuning those parameters can be tedious as parameters need to be tuned if deployment scenarios are different. Moreover, it is hard to generate data sets for self-tuning the fuzzy rule base through experiments or simulation. In the following chapter an uncertainty indicator of the localisation process will be developed to prioritise the USBL navigation aids for some AUVs over the rest. This uncertainty indicator can be also used to control NBs broadcasting period e.g., when greater than a predefined threshold. A single localisation precision indicator (i.e., based on each localisation method error characteristics) is utilised in the following chapter to decide on which localisation method to be adopted in case of redundant navigation information.

Chapter 6

Confidence based Underwater Swarm Localisation and Optimisation

A confidence-based localisation algorithm is proposed in this chapter for improving localisation accuracy by promoting AUVs with high confidence of location estimates to references for their neighboring AUVs, and therefore increasing the external navigation aids update rate. Confidence update rules based on Bayes filters are proposed given localisation methods' error characteristics where expected localisation error is generated based on measurements such as operational depth and travelled distance. The proposed algorithm's key parameters are then optimised using the [Evolutionary Multi-objective Optimisation \(EMO\)](#) algorithm NSGA-II [170] for localisation error minimisation and localisation confidence maximisation. The proposed confidence-based localisation algorithm and its optimised parameters are published in [28] and [29].

6.1 Introduction

Over the past two decades, swarm robotics have become an attractive research area as it can provide relatively low-cost solutions for complicated tasks in many applications, especially when wide region coverage is required [9], [106] and [10]. Locations of each

individual node must be known and tracked during operation for location-aware applications e.g. seismic imaging and environmental monitoring. Nodes localisation can be performed in a *centralised way* in which information is gathered from all nodes including a set of anchors to be processed in centralised server for localisation i.e., decision is made based on complete information from the whole swarm and environment. Nodes localisation can alternatively be performed in a *cooperative way* in which anchors as well as a few nodes with known locations, at a certain instant, navigationally aid the others. Most efforts in this area, such as [107] and [108], assume an environment where information can be exchanged easily among team members. We rather consider a swarm of AUVs that utilises acoustic channels for communication. The high cost of AUVs, the severely limited bandwidth and long latency of the underwater acoustic communication limit the number of AUVs that can be deployed at once to collaboratively complete a mission [19] and [21]. Therefore, most terrestrial and aerial swarm localisation algorithms reported in the literature are not directly applicable to underwater swarm robotic systems. The USBL is the most commonly adopted localisation method in the industry due to its flexibility as it does not require artificial landmarks to be deployed on the seafloor and it only requires a single surface vessel for operation. However, the maximum number of underwater targets that can be localised in a single TDMA frame by the USBL is very limited (up to 10 using the most advanced technology) [66]. Different localisation methods including trilateration and dead reckoning are employed when USBL is not available in hierarchical localisation [80, 79, 171].

In this chapter, we lay down the proposed algorithm’s concepts in underwater cooperative localisation for swarm of AUVs. Underwater acoustic communication constraints (i.e., packet loss and limited bandwidth) are not considered in this chapter for proof-of-concept simulation but it will be considered in Chapter 7 using the high fidelity physics based co-simulation platform developed in Chapter 4.

A confidence-based underwater localisation scheme is introduced in which three common localisation methods, namely USBL localisation, trilateration and DR were adopted. The confidence threshold and node ¹ density are key parameters to the confidence-based localisation algorithm’s performance, so they are optimised for accuracy enhancement using an Evolutionary Multi-objective Optimisation algorithm through extensive simulation. Each AUV or node in the swarm is associated with a scalar confidence value which represents the localisation estimate precision using a belief function to update the role of an AUV by either promoting it from an ordinary node to a reference node or demoting it from a reference node to an ordinary node in the swarm. Confidence values

¹The term “node” and “AUV” are used interchangeably in this chapter

are updated and monitored through the proposed algorithm in which confidence update rules based on localisation error characteristics and Bayes filters are employed. Nodes with high confidence can be employed as references for neighboring ordinary nodes localisation using trilateration.

The remainder of this chapter is organised as follows. Section 6.2 provides a brief background of both hierarchical localisation algorithms, source of inspiration, and multi-objective optimisation. Section 6.3 explains the proposed algorithm and formulates the multi-objective optimisation problem for finding the optimal confidence threshold and AUV density. Section 6.4 shows how to employ localisation method's error characteristics in confidence update rules and multi-objective optimisation in localisation accuracy improvement. Moreover, the algorithm's performance is compared for both optimised and arbitrary non-optimised cases and a scalability test is carried out. The algorithm's performance is also compared to round-robin scheduling-based USBL aided DR navigation. Finally, section 6.5 summaries this chapter.

6.2 Background

6.2.1 Evolutionary Multi-objective Optimisation (EMO)

Multi-objective optimisation involves, as the name suggests, optimising more than one objective. The problem of Multi-objective optimisation becomes particularly challenging when the objectives are irreconcilable. Population based approaches such as Evolutionary Optimisation (EO) algorithms are very popular in EMO. EO algorithms are fundamentally different from classical optimisation methodologies as: (a) EO algorithms do not require the gradient of the objective functions (b) EO algorithms update more than one solution in each iteration (*population-based*) versus classical optimisation approaches update a single solution in each iteration (*point-based*) [172]. EO approaches are flexible and widely applicable as they do not require the gradient information of the objective functions which are hard to be obtained in most real-life applications. Moreover, the nature of population-based search allows EO algorithms in multi-objective optimisation problems to return a set of trade-off optimal solutions instead of a single solution that does not optimise all irreconcilable objectives [173]. For the aforementioned reasons, EO algorithms are superior choice for multi-objective optimisation [174].

EO algorithms begins with random initialisation of the population over the solutions' search space; each individual in the population represents a solution. The population

evolves iteratively through the selection, crossover and mutation procedure. The last generation would have the elitist population. Each individual in the population is evaluated according to the fitness function in the selection process (*tournament selection*) [172]; however obtaining a mathematical representation of the fitness function may not always be possible (e.g. topology optimisation) in such cases each member is evaluated on a simulated environment that captures the main characteristics of the optimisation problem (*simulation-based optimisation*) [175]. The new generation (i.e., offspring) is then created from the fittest members of the tournament selection by the crossover process. The new generation would have a mixture of their parents' characteristics (parents' genes). New characteristics or genes are introduced to a small portion of the offspring in the mutation process so they are not a mirror subset of their parents [172]. Mutation severity and portion size of the new generation to mutate is usually governed by a probabilistic distribution. The mutation process is crucial in EO so that the new generation does not stick in a local extrema [172].

EMO has increasingly attracted research and industry communities' attention. It is now an established field of research and applications. The Fast and Elitist Multi-objective Genetic Algorithm NSGA-II is a robust and efficient EMO algorithm introduced by Deb et al. to find the *Pareto-optimal* set based on *non-dominated sorting* and *crowding distance* [170]. Pareto-optimal solutions are the elitists population in the last generation in which choosing one solution over another requires sacrificing one objective and gaining another [173]. A solution is said to dominate another when it is not worse in all objectives and better in at least one objective. The crowding distance is simply a measure of how close a solution is to another. Longer distances are associated with higher scores, and thus the diversity is ensured in the Pareto-optimal set. The population of each generation in EMO algorithms is divided into non-dominated fronts. The solutions in the first non-dominated front are identified by comparing each solution with every other solution in the population to find if it is dominated [174]. To find the solution in the subsequent fronts, the solutions in the first front are excluded temporarily of the population and the same comparison procedure is repeated [174].

In NSGA-II, fast non-dominated sorting is carried out by calculating (i) the domination count which is the number of solutions that dominate a certain individual and (ii) the set of solutions that a certain individual dominates [170]. The domination count of each individual in the first front is equal to zero. The solutions that a first front individual dominates are visited and their domination counts are discounted by one. The solutions with a domination count of zero after the discount are the second front members. The same procedure is repeated on the solutions dominated by the second front individuals

to find the third front members.

Figure 6.1 shows an example of a Pareto-optimal solutions in three fronts in the objective space of f_1 and f_2 . Assume both objective function f_1 and f_2 to be minimised so individual **D** has a domination count of zero as it is not dominated by any other solution, therefore it belongs to front 1. However if we consider that individual **C** domination count is equal to two, the first front solution **D** dominates [**A**, **B**, **C**, ...] and the first front solution **E** dominates [**B**, **C**, ...]. Therefore solution **C** would then belong to the second front as its domination count is equal to zero after it has been discounted by one for every time individual **C** is found in a first front members' dominating list.

The second main sorting approach in NSGA-II after the fast non-dominated sorting is the crowding distance sorting. The population is first sorted according to the ascending order of each objective function value. Infinity distances are assigned to the boundary solutions and the distance for all other intermediate solutions is equal to the absolute normalised difference in the function values of two sandwich-neighbouring solutions. The crowding distance would be equal to the sum of the individual's distances in each objective. Members with high crowding-distance values are associated with high score (preferred) to ensure the diversity in the next population. The dashed rectangle in Figure 6.1 shows an example of a Front 3 individual's crowding distance which is equal to $(d_1 + d_2)$ as both d_1 and d_2 correspond to the absolute normalised difference of the adjacent neighbouring solution values in both objectives f_1 and f_2 .

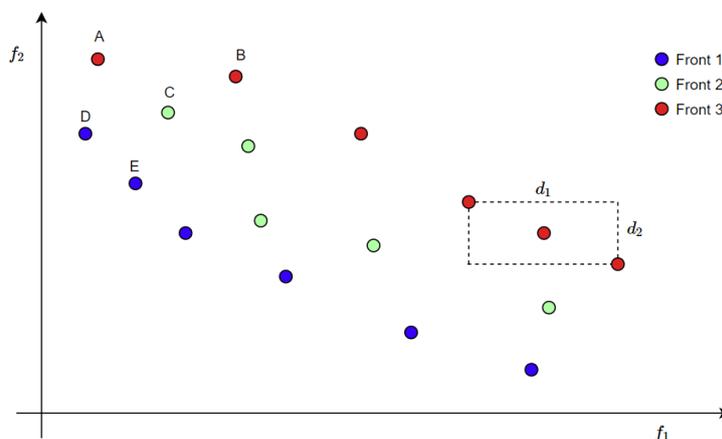


Figure 6.1: A Pareto-optimal solutions in three fronts in the objective space of f_1 and f_2 with an example of non-dominated sorting and crowding-distance sorting procedures in NSGA-II.

The Fast and Elitist Multi-objective Genetic Algorithm NSGA-II procedure begins with random parents population and each member is assigned a fitness equal to its non-domination rank. The usual selection, crossover and mutation operators are adopted to create offspring population. From the second iteration till the algorithm terminates, the parents P_t and offspring Q_t populations are combined to create a new population R_t as shown in Figure 6.2. The new population is then sorted into different fronts by the fast non-dominated sorting, members with lower rank fronts pass through to the next population (minimisation of the objective functions is assumed). The next parent population P_{t+1} has to be of the same size of the old parents population P_t . The crowding-distance approach is then adopted to either accept or reject members of the same front rank in the next parents population P_{t+1} . Interested readers are referred to [170] for the computation complexity of NSGA-II.

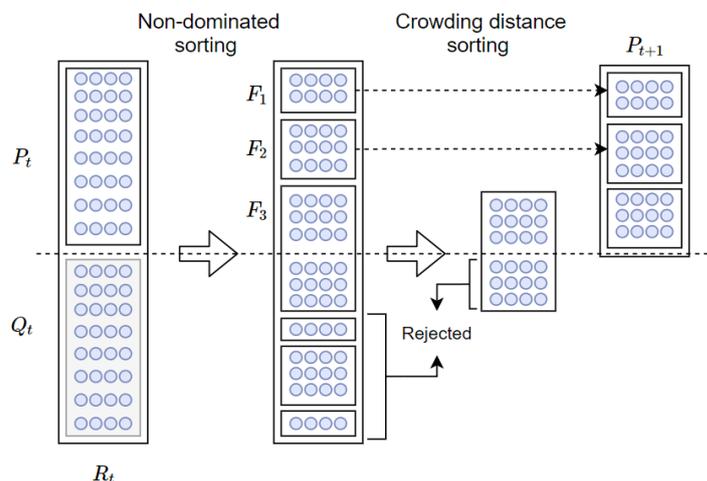


Figure 6.2: NSGA-II procedure where the population is sorted into different fronts (F_1, F_2, F_3, \dots) by fast non-dominated sorting and members of the same front are either accepted or rejected in the next parents population based on its crowding distance values.

6.2.2 Hierarchical Underwater Localisation

A large-scale hierarchical localisation approach has been investigated in [79] for stationary underwater sensor network. Zhou et al. in [80] extended the algorithm in [79] and introduced a hierarchical localisation approach for mobile underwater sensor network in which underwater sensors predict their mobility patterns. The details of both [79] and

[80] have been discussed in Chapter 3. The main concept behind a hierarchical localisation approach is that a successfully localised ordinary node with high precision can serve as a reference node for neighboring nodes localisation. Both [79] and [80] considered a simple approach to regulate the promotion of ordinary nodes to reference nodes. They introduced the concept of the confidence value and confidence threshold for promoting ordinary nodes to reference nodes and demoting reference nodes. Confidence values of localised ordinary nodes in [79] were solely dependent on the localisation error. The major drawback of this algorithm is that it is not always possible to measure localisation error in underwater missions. However, the confidence values in [80] were calculated by simply averaging the participating reference nodes' confidence values and considering the error in range measurements. Bhuvanewari et al. in [176] proposed a confidence discount rule based on the number of time steps since last external localisation update and a high but arbitrarily defined confidence threshold. A computationally expensive quality of trilateration-based localisation scheme in 2-dimensional space has been introduced in [177] where reference nodes are selected based on geometric relationship of their positions and ranging errors. The authors in [177] focused only on localisation by trilateration and considered a scenario in which a node has to select 3 reference nodes for localisation based on their quality-of-trilateration score.

6.3 Confidence-based localisation Algorithm

In this section, confidence-based localisation algorithm for a swarm of mobile underwater sensor nodes is proposed. The proposed algorithm aims at improving localisation coverage and localisation estimate accuracy by promoting high-precision localised ordinary nodes to reference nodes based on their confidence values. The confidence value of a node is dynamically updated by the proposed confidence update rules.

6.3.1 Confidence Update Rules

Consider a swarm of N AUVs denoted by a set $\mathcal{A} = \{a^1, a^2, \dots, a^N\}$. Define δ_t^i as a confidence value, which is between 0 and 1, associated with the AUV a^i at time t . It measures the uncertainty associated with the AUV's current location estimate using a belief function. The certainty of an AUV being at a certain position can be considered

as a belief (i.e., state of knowledge) and it can be represented as a conditional likelihood distribution [178]. A belief can be easily calculated by the Bayes filter algorithm [178]. If an AUV has a confidence value of 1, its current location estimate is certain. On the other hand, the current location estimate of an AUV is completely unreliable if its confidence value is 0. Initially it is set to 1 as all AUVs are deployed from a known position. The confidence value of AUV a^i (δ_t^i) is dynamically updated in each localisation step. Any localisation method can be integrated in the proposed algorithm by implementing confidence value update rules based on a localisation method's error characteristics i.e., mean localisation error and standard deviation of localisation error. Different update rules of the confidence value are implemented based on a designated localisation method's expected localisation error. In contrast to terrestrial localisation, localisation estimate error in the underwater environment cannot be measured unless a sophisticated localisation system is employed such as LBL which require artificial landmarks to be deployed on the seafloor in advance. If the confidence value of AUV a^i (δ_t^i) drops below a pre-defined confidence threshold λ_1 and the USBL is available, then AUV a^i will be localised by the USBL and its confidence value is updated (boosted) based on its previous confidence value δ_{t-1}^i that is related to p_{t-1}^i where p_{t-1}^i is the estimated position at time $t-1$ and measurements $\{z_t: \text{operational depth}\}$ which can be accurately acquired by a depth sensor. The AUV's confidence value (δ_t^i) is updated as follows

$$\delta_t^i := \text{bel}(p_t^i) \quad (6.1)$$

$$\text{bel}(p_t^i) = \mathcal{L}(m_t | p_t^i) \text{bel}(p_{t-1}^i) \quad (6.2)$$

where $\mathcal{L}(m_t | p_t^i)$ represents the likelihood of AUV a^i being at the estimated position p_t^i given some measurements m_t i.e., the operational depth z_t when an AUV is localised by the USBL. In other words, the likelihood of an estimated position being matched with an expected position is related to the expected error derived from a localisation method's error characteristics. If an AUV does not receive a USBL localisation aid (USBL is not available), then three conditions will be checked (refer to Algorithm 1) prior to performing ToA based trilateration [155] where J is the number of neighbouring AUVs and l_d is the minimum bounding box's dimensions formed by neighbouring AUVs $j = 1, 2, \dots, J$. ToA-based trilateration least squares problem is solved using PSO [101] as shown in subsection 3.3.2. Confidence value (δ_t^i) is updated, in this case, based on neighbouring AUVs confidence values (δ_t^j) and their estimated positions (p_t^j), the estimated position of AUV a^i (p_t^i) and range measurements (r_t^{ij}) between AUV a^i and its neighbouring AUVs ($a^j : j = 1, 2, \dots, J$):

$$\delta_t^i = \frac{\sum_{j=1}^J \delta_t^j \left(1 - \frac{\|p_t^j - p_t^i\| - r_t^{ij}}{\|p_t^j - p_t^i\|}\right)}{J} \quad (6.3)$$

Equation 6.3 is introduced based on empirical evaluation as it considers the undiscounted confidence value of a neighbour AUV a^j if the distance between AUV a^i and a^j through their estimated positions (p_t^i) and (p_t^j) perfectly matches the corresponding range measurement (r_t^{ij}). AUV a^i location will be tracked using dead reckoning when neither USBL nor trilateration method can be adopted. Confidence value (δ_t^i) is discounted based on its previous confidence value and current measurements (w_t : travelled distance since the last USBL or trilateration localisation) using Equations 6.1 and 6.2 with $m_t = w_t$. Algorithm 1 depicts the localisation process of AUV a^i in which USBL system (at most ten AUVs can be localised in a single TDMA frame of 1 second) [66], trilateration or dead reckoning localisation is selected for every localisation period ΔT based on its confidence value (δ_t^i).

Algorithm 1: Confidence-based localisation - AUV a^i localisation

Result: p_t^i, δ_t^i

initialisation: $p_{t=t_0}^i = p_{initial}^i, \delta_{t=t_0}^i = 1;$

for each time step ΔT do

if $\delta_t^i \leq \lambda_1$ then

 Request a USBL localisation aid

if request is granted then

 Adopt USBL

 Update $\delta_t^i \leftarrow (p_{t-1}^i, z_t)$ as in 6.2

end

if request is not granted && $\min_{j=1:J} \delta_t^j \geq \lambda_1$ && $J \geq 4$ && $\min_{d=1:3} l_d \geq 1$ then

 Adopt Trilateration

 Update $\delta_t^i \leftarrow (\delta_t^j, p_t^j, p_t^i, r_t^{ij})$ as in 6.3

else

 Adopt Dead reckoning

 Update $\delta_t^i \leftarrow (p_{t-1}^i, w_t)$ as in 6.2

end

else

 Adopt Dead reckoning

 Update $\delta_t^i \leftarrow (p_{t-1}^i, w_t)$ as in 6.2

end

end

6.3.2 Parameters Optimisation

In the proposed algorithm, a predefined confidence threshold (λ_1) is set to promote an ordinary high precision localised **AUV** to a reference **AUV**. However, determining a universal confidence threshold that suits different **AUV** deployment scenarios is laborious and nearly impossible. In addition, as far as **ToA**-based trilateration localisation method is concerned, a minimum **AUV** density (λ_2) in the swarm should also be carefully maintained i.e., at least 3 references should be in an **AUV**'s communication range to perform trilateration as explained in subsection 3.3.2. Random walkers motion model is a simple way to generate many possible spatial patterns of nodes in a node's neighbourhood. Therefore, it has been commonly assumed that the optimised parameters on random walkers may suit various deployment scenarios. We have assumed correlated and uncorrelated random walker models [179] to govern the mobility of nodes in a confined region. The impact of confidence threshold and **AUV** density on localisation performance have been investigated through extensive simulation. Four performance metrics were considered, namely mean localisation error, mean confidence value, **USBL** utilisation and **ToA**-based trilateration utilisation.

Our objectives are to minimise localisation error (i.e., $f_1(\lambda_1, \lambda_2)$) and **ToA**-based trilateration utilisation (i.e., $f_2(\lambda_1, \lambda_2)$) due to its high demand of on-board computational power while maximising mean confidence value (i.e., $f_3(\lambda_1, \lambda_2)$) and **USBL** utilisation (i.e., $f_4(\lambda_1, \lambda_2)$) as it is the most reliable localisation method. There is no single optimum solution in the parameter space that simultaneously optimises these four irreconcilable objectives in Equation 6.4. However, a set of optimal solutions that provides a trade-off among objectives seems ideal to this multi-objective optimisation problem:

$$\left\{ \begin{array}{l} \min f_1(\lambda_1, \lambda_2) \\ \min f_2(\lambda_1, \lambda_2) \\ \max f_3(\lambda_1, \lambda_2) \\ \max f_4(\lambda_1, \lambda_2) \end{array} \right. \text{ subject to } \left\{ \begin{array}{l} L_1 \leq \lambda_1 \leq U_1 \\ L_2 \leq \lambda_2 \leq U_2 \end{array} \right. \quad (6.4)$$

where λ_1 is confidence threshold, λ_2 is **AUV** density, L_i and U_i ($i = 1, 2$) are their lower and upper bounds respectively. **AUV** density (λ_2) is defined as the expected number of **AUV**s in an **AUV**'s neighbourhood and thus it can be varied by the **AUV**s' communication range.

The objective functions f_1, f_2, f_3 and f_4 are defined in the simulation as follows:

$$f(\lambda_1, \lambda_2) = \frac{\sum_{i=1}^N \left(\frac{\sum_{t=0}^T \Psi_t^i}{T} \right)}{N} \quad (6.5)$$

where N is the swarm size and T is the simulation maximum time step. Ψ_t^i in Equation 6.5 is defined as the localisation error for f_1 , true binary variable if trilateration is successful for f_2 , confidence value for f_3 and true binary variable if USBL aiding is successful for f_4 of AUV a^i at time instant t . The Fast and Elitist Multi-objective Genetic Algorithm NSGA-II [170] explained earlier in Section 6.2 is adopted to find the Pareto-optimal solutions of Equation 6.4.

6.4 Simulation

In this section, error characteristics of localisation methods are employed to generate a localisation method's expected error and thus, confidence values are updated as in Equations 6.2 and 6.3. The importance of optimising confidence threshold and node density is emphasised in this section by comparing the proposed algorithm's performance with optimised parameters and arbitrarily selected non-optimised parameters.

6.4.1 Error Characteristics for Confidence Update

When USBL localisation method is adopted, the expected error for localisation estimate can be generated based on its error characteristics. In 1000 m depth, 63% (1 Drms) of the total USBL localisation errors are within 2.7 m radius [66]. We assume that the localisation estimate error of a given USBL system follows a Gaussian distribution given by

$$\mathcal{E}_U \sim \mathcal{N}(\mu, \sigma^2) \quad (6.6)$$

where $\mu = 2.7$ m and $\sigma =$ total error (1Drms) depicted from the relationship in [66] and shown in Figure 5.3. The USBL localisation error can be predicted based on the operational depth. We calculate the likelihood $\mathcal{L}(z_t | p_t^i)$ in Equation 6.2 as follows

$$\mathcal{L}(z_t | p_t^i) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(\mathcal{E}_U)^2/2\sigma^2} + \tau \quad (6.7)$$

where \mathcal{E}_U is the **USBL** expected localisation error, τ is a damping factor. A damping factor (τ) is crucial for the likelihood stability, the higher the value of τ the less-likely the confidence value is to fluctuate. Equation 6.3 is used to calculate the confidence value of **AUV** a^i when **ToA**-based trilateration is adopted. Based on existing underwater range measurement technologies we assume that the range measurement between two arbitrary neighboring **AUVs** a^i and a^j (r_t^{ij}) follows a Gaussian distribution with mean equal to real measured range and standard deviation of 2% of the mean [164]. In case none of the available external localisation aids is adopted, an **AUV**'s location is tracked using **IMU**-based dead reckoning. Confidence value (δ_t^i) is then updated based on equation 6.2. We assume a low cost and low power consumption sensor suite consists of **INS** and pressure gauge employed in each node with a typical dead reckoning accuracy of 30% of travelled distance [180]. We calculate the expected error of dead reckoning localisation as follows

$$\mathcal{E}_D = w_t \Phi : \Phi \sim \text{uniform}(\alpha, \beta) \quad (6.8)$$

$$\mathcal{L}(w_t | p_t^i) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(\mathcal{E}_D)^2/2\sigma^2} + \tau \quad (6.9)$$

where \mathcal{E}_D is the dead reckoning expected localisation error, α is proportionally related to the number of dead reckoning navigation steps (resets to 0 when **USBL** or trilateration is adopted) and β is the maximum drift of dead reckoning navigation i.e., 30% of travelled distance. Thus, the width of the probability density function of Φ is decreasing when time progresses.

6.4.2 Simulation Settings

Suppose 100 identical mobile nodes are randomly deployed on a surface of a confined region of 100 m³. Each node is equipped with a depth sensor with accuracy of 0.01% [181], **IMU** with a typical **INS** dead reckoning accuracy of 30% [180] of the travelled distance, a **USBL** transponder and a short-range communications modem. Assume a **USBL** localisation system, hull mounted on a surface vessel, capable of localising 10 nodes in a single **TDMA** frame of 1 second [66]. Correlated and uncorrelated random walker models [179] are employed to govern the mobility of the nodes. Table 6.1 summarises the key parameters of the simulation and Evolutionary Multi-objective Optimisation NSGA-II [170] used in confidence-based localisation algorithm optimisation. Notice that we consider measuring distances in the objectives space (Pareto Front) instead of variables space for crowding-distance as the computed distances of solutions in variables space

Parameter	Value
Endurance time	1000-time step
Time step	1 s
Swarm size	100-node
Initial confidence value	1
Max number of USBL localised nodes in a single TDMA frame	10-node
USBL TDMA frame length	1 s
Max dead reckoning drift	30%
Intra-swarm node's communication range	[5, 55] m
Confidence threshold	[0, 1]
NSGA-II population size	1000
NSGA-II max generation No.	500
NSGA-II non-dominated fraction	0.02

Table 6.1: Simulation parameters

might be very small although their corresponding Pareto Front distances are not.

6.4.3 Results and Analysis

The proposed algorithm performance with respect to the four aforementioned performance indicators has been investigated through more than 200 simulations in which the confidence threshold (λ_1) was varied from 0 to 1 with an increment of 0.05 and nodes' communication range were varied from 5 m to 55 m with an increment of 5 m. This represents nodes density (λ_2) ranging from 0 to almost 40 as shown in Figure 6.3. Figure 6.4 shows the impact of the confidence threshold and the AUVs density (varied by AUVs' communication range) on (a) mean localisation error, (b) mean confidence value, (c) USBL utilisation and (d) ToA-based trilateration utilisation in a swarm of 100 AUVs.

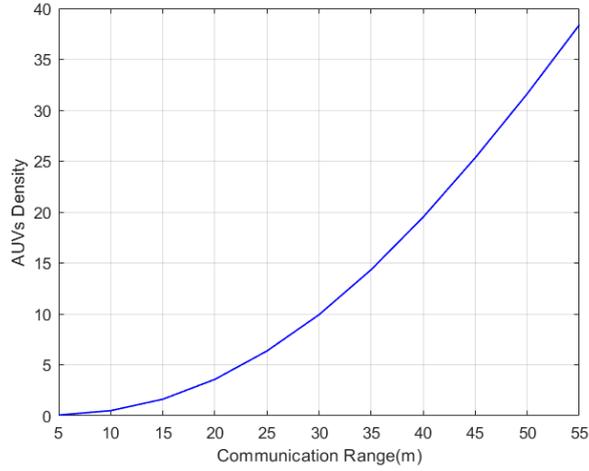
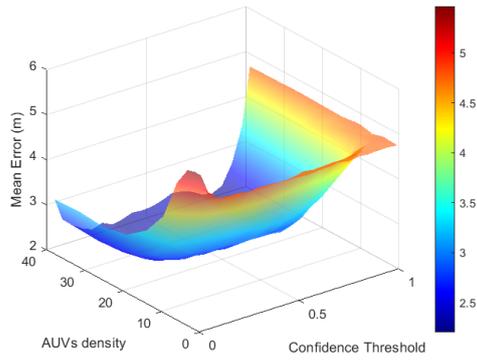
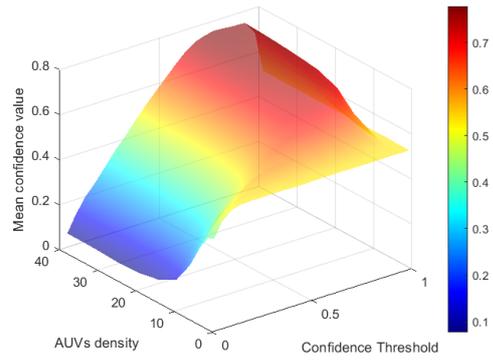


Figure 6.3: The relationship between the average number of the nodes in a node's neighborhood (AUVs density) and node's communication range.

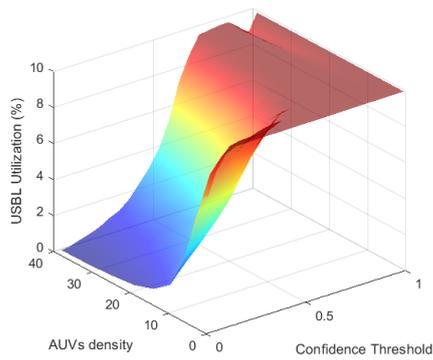
The fitness function of each objective has been built based on data fitting models of the objective function surfaces shown in Figure 6.4. The evolutionary multi-objective optimisation method NSGA-II is then employed to find the optimised confidence threshold $\lambda_1 : 0 \leq \lambda_1 \leq 1$ and node density $\lambda_2 : 0 \leq \lambda_2 \leq 40$. The upper bound of λ_2 (40) is equivalent to nodes' communication range of more than 50% (≈ 55 m) of a deployment region's dimension i.e., 100 m. Figure 6.5 reveals the Pareto Front (Pareto-optimal set score in objectives space) and Figure 6.6 shows the corresponding Pareto-optimal set. Figure 6.7 shows the score of the four objectives of four dominant optimal solutions in the proposed deployment scenario.



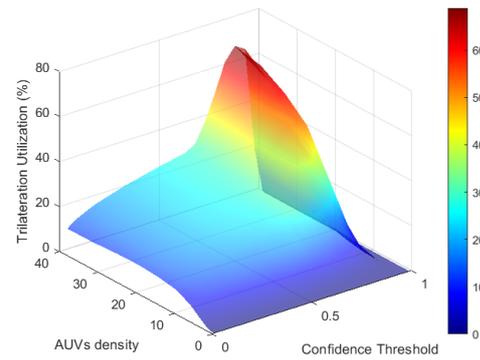
(a)



(b)



(c)



(d)

Figure 6.4: The impact of the confidence threshold and AUVs density on (a) mean error (b) mean confidence value (c) USBL utilisation and (d) ToA-based trilateration utilisation in a swarm of 100 AUVs over 1000 localisation period.

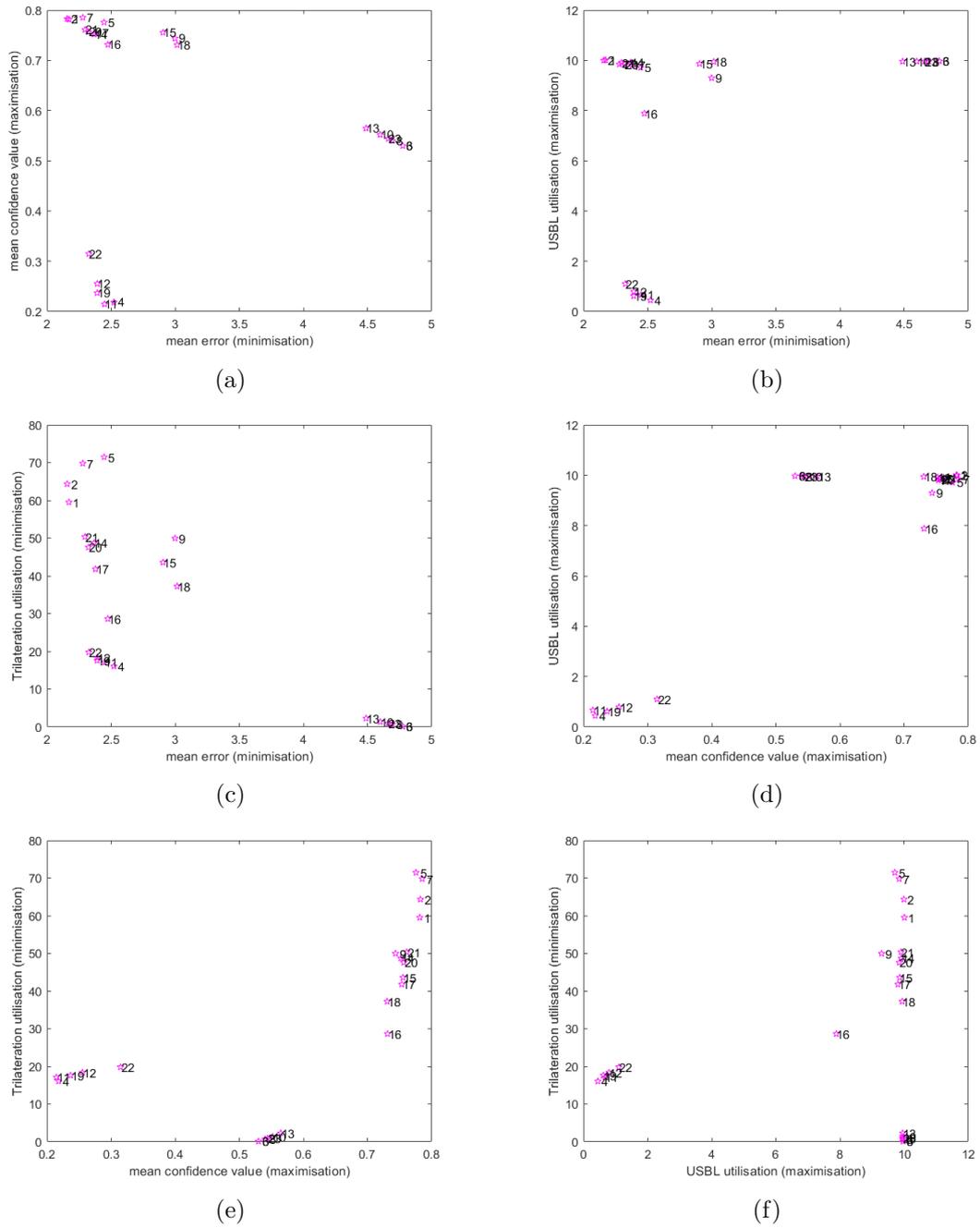


Figure 6.5: The score of Pareto-optimal set, Pareto Front, in (a) mean error and mean confidence value (b) mean error and USBL utilisation (c) mean error and ToA-based trilateration utilisation (d) mean confidence value and USBL utilisation (e) mean confidence value and ToA-based utilisation and (f) USBL utilisation and ToA-based utilisation. The solutions in Pareto front are numbered from 1 to 23.

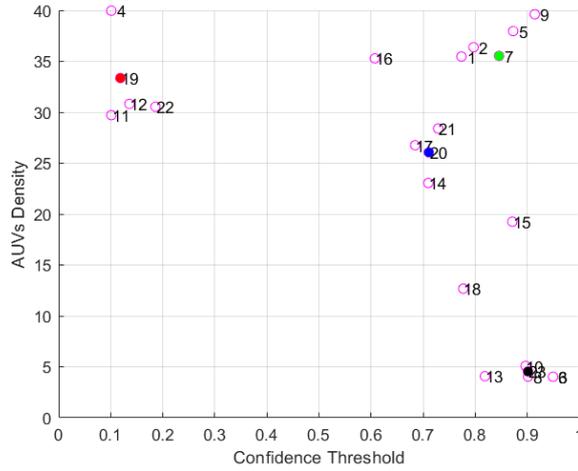


Figure 6.6: The corresponding Pareto-optimal set of Pareto Front (in Confidence Threshold and AUVs Density). Four optimal solutions are selected (filled coloured circles) to represent their typical clusters' parameter values in the objective space.

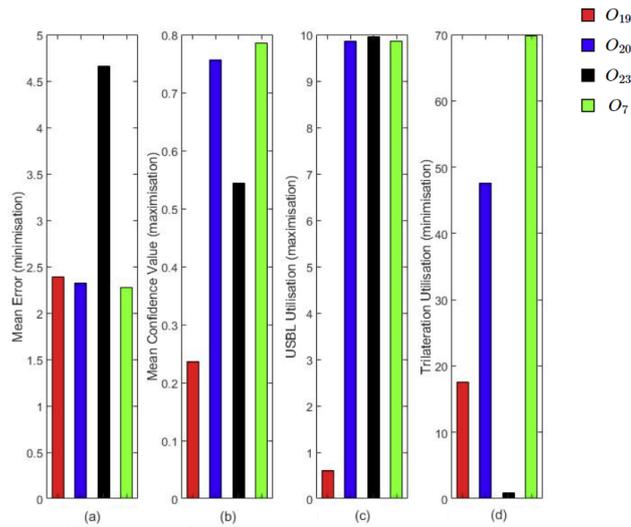


Figure 6.7: The score of four selected optimal solutions in the four objectives (a) mean error (b) mean confidence value (c) USBL utilisation and (d) ToA-based utilisation.

A decision maker now has the option to choose any of the solutions in the Pareto-optimal set in Figure 6.6 based on application requirements or objectives priorities. It can be noticed that the optimal solutions in Figure 6.6 can be grouped into 4 clusters. We therefore select a single solution in each cluster in the Pareto-optimal set to emphasise each cluster's score in the Pareto Front. The selected four optimal solutions (coloured)

are 19, 20, 7 and 23, as shown in Figure 6.6. Solution 7 (O_7) minimises the mean error in Figure 6.7 (a) while maximises mean confidence value in Figure 6.7 (b) and USBL utilisation in Figure 6.7 (c) but it does not minimise trilateration utilisation in Figure 6.7 (d). Although O_{23} minimises trilateration utilisation in Figure 6.7 (d), it maximises the mean error in Figure 6.7 (a). However, O_{19} outperforms O_{20} in minimising the trilateration utilisation in Figure 6.7 (d) by around 30%. O_{20} outperforms O_{19} in maximising both USBL utilisation and mean confidence value; hence O_{19} suggests mostly dead-reckoning localisation which may lead to severe error accumulation in different deployment scenarios. Therefore, we select the set of optimal parameters represented by O_{20} . It is worth mentioning that O_{19} can provide optimal parameters for our deployment scenario given the relatively small deployment region considered. From Figure 6.6, O_{20} suggests a confidence threshold of 0.7109 and AUV density of 26 (communication range of 45 m). Figure 6.8 below shows histograms of localisation estimate error and confidence value of arbitrarily selected AUV in a swarm of 100 AUVs over 1000 localisation period in an arbitrarily selected non-optimal case where confidence threshold (λ_1) is 0.9 and AUV density (λ_2) is 6.35 (25 m communication range) and in the selected optimal case (O_{20}). Figure 6.9 depicts the traces of localisation error, confidence value and the adopted localisation method in each localisation period of the same AUV presented in Figure 6.8 over a time window of 150 localisation period in both cases.

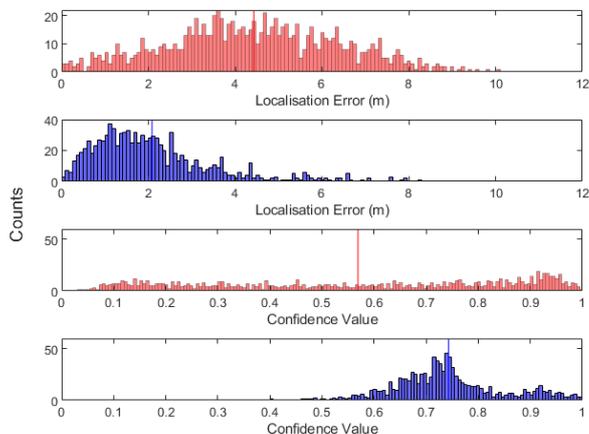


Figure 6.8: Histograms of localisation error and confidence value of a single AUV in both a non-optimal case (red) and the optimal case (blue) over 1000 localisation period with mean localisation error of 4.42 m and 2.08 m and mean confidence value of 0.56 and 0.74 in the non-optimal and the optimal cases respectively.

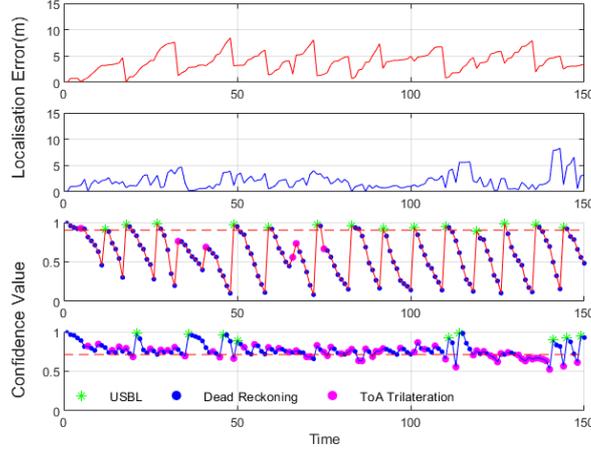


Figure 6.9: Traces of typical localisation errors and confidence values of a single AUV over the first 150 localisation period in the non-optimal case (red) and the optimal case (blue). The red dashed horizontal lines represent confidence thresholds.

When both Confidence Threshold (λ_1) and AUV Density (λ_2) are optimised, the AUV presented in Figure 6.9 (the optimal case) was considered as a reference node for 62.9% (629 localisation periods) of the total running time (1000 localisation periods). In contrast, when confidence threshold and AUV density were arbitrarily set to 0.9 and 6.35 respectively (a non-optimal case), the AUV presented in Figure 6.9 was considered as a reference AUV for only 18% (180 localisation period) of the total running time. Consequently, mean localisation error and mean confidence value have been improved by 52.94% and 32.14% respectively when confidence threshold and AUV density are optimised as shown in Figure 6.8. More AUVs can become reference AUVs for trilateration with sufficiently high confidence in the optimised case. In addition, standard deviations of both localisation estimate error and confidence value in Figure 6.8 have been improved by around 30.15% (from 1.99 to 1.39) and 65.5% (from 0.29 to 0.10) respectively. Figure 6.10 shows histograms of the localisation estimate error and the confidence value of the entire swarm (i.e., 100 AUVs) in the pre-mentioned non-optimal case and in the suggested optimal case (O_{20}).

Figure 6.10 reveals an improvement of 47.7% in localisation mean error, 27.3% in localisation error standard deviation and 33.92% in the mean confidence value in the swarm (10^5 localisation period) when algorithm's parameters (confidence threshold and AUV density) are optimised.

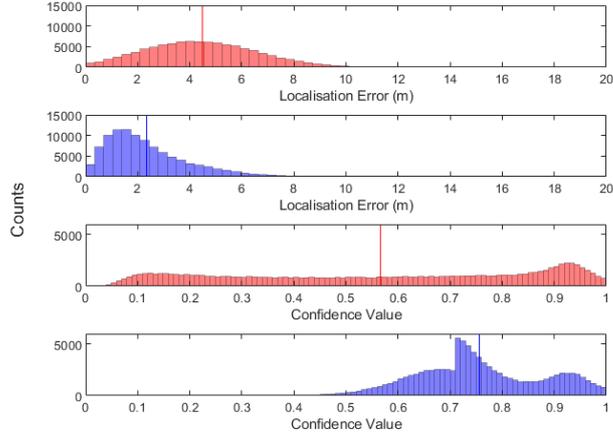


Figure 6.10: Histograms of localisation error and confidence value of 100 AUVs in both the non-optimal case (red) and the optimal case (blue) over 1000 localisation periods. The mean localisation error in 10^5 localisation periods is equal to 4.48 m and 2.34 m with mean confidence value equal to 0.56 and 0.75 in the non-optimal and the optimal cases respectively.

We have furthermore taken the advantage of the simulation platform simplicity to test the scalability of the proposed confidence-based localisation approach. An extensive simulation has been performed to test the scalability and to further compare the proposed localisation approach with round-robin scheduling-based **USBL** aided **DR** navigation. Figure 6.11 shows the mean localisation error of the entire swarm in both the proposed confidence-based **USBL**/trilateration aided **DR** navigation and round-robin scheduling-based **USBL** aided **DR** navigation and Figure 6.12 shows the associated standard deviation. Mean localisation error and standard deviation of the entire swarm shown in Figures 6.11 and 6.12 are depicted over a swarm size of 10 to 1000 **AUVs** with an increment of 10 **AUVs**.

It can be clearly noticed that the localisation mean error and standard deviation in round-robin scheduling-based **USBL** aided **DR** navigation increases linearly with increasing swarm size as the **USBL** update delay increases linearly in round-robin scheduling. Whereas mean localisation error slightly increases as shown in Figure 6.11 with increasing swarm size when the proposed confidence-based localisation algorithm is adopted. Linear regression has been performed on the data presented in Figure 6.11 to find the data model in both cases. The regression coefficient (slope) in round-robin scheduling-based **USBL** aided **DR** navigation is equal to 74×10^{-4} whereas it is equal to 8.27×10^{-4}

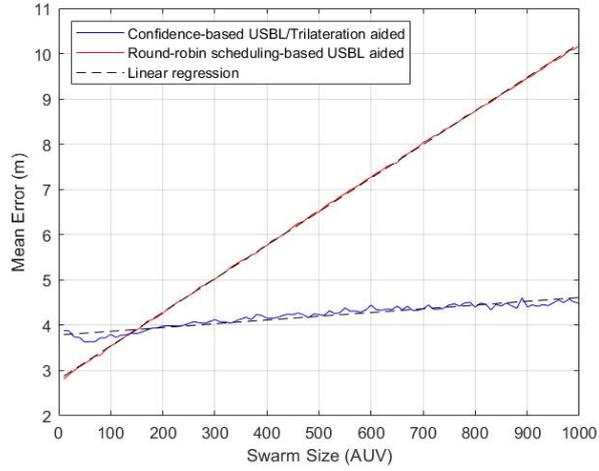


Figure 6.11: The performance of the proposed algorithm is compared through the mean error of the entire swarm with round-robin scheduling-based USBL aided dead reckoning navigation. The black dashed lines represent the data models in both cases found by linear regressions.

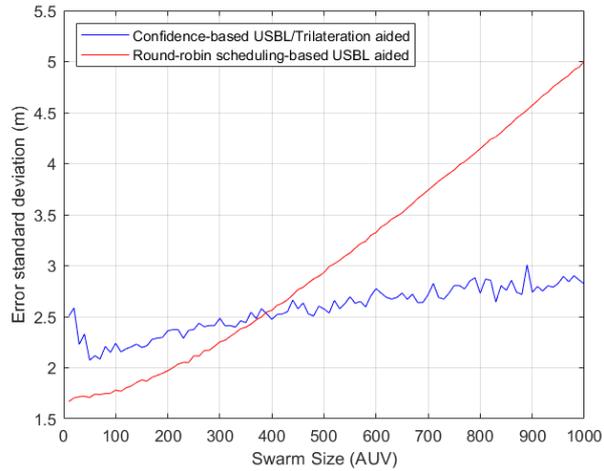


Figure 6.12: Localization error standard deviation of the whole swarm is compared when two different localization methods are employed.

in the proposed confidence-based USBL/trilateration aided DR navigation with coefficient of determination R^2 of 0.99 and 0.89 respectively.

Figure 6.11 shows that round-robin scheduling approach outperforms the proposed localisation approach in swarm sizes less than 130 AUVs. This depends on two factors, namely the AUV's on-board IMU accuracy and trilateration robustness. The AUV's

on-board **IMU** accuracy is essential to the time interval length between two consecutive external navigation aids i.e., **USBL** or trilateration.

In deployment scenarios of swarm sizes less than 130 **AUVs**, frequent **USBL** aids and relatively accurate **AUV**'s on-board **IMU**, either an outlier detection algorithm can be developed and integrated in the proposed algorithm to reject inaccurate trilateration results or further tuning of the proposed algorithm's parameters can be carried out to minimise trilateration utilisation. In the later case the proposed localisation algorithm performance would at least be similar to round-robin scheduling performance.

6.5 Summary

In this chapter, a confidence-based algorithm is proposed for underwater localisation in large-scale mobile swarm. Confidence threshold and **AUV** density are key parameters in the proposed algorithm. They are obtained and optimised through extensive simulation in which random walker models are applied so that the optimised parameters could suit various deployment scenarios. The presented results highlight the importance of optimising the proposed algorithm's parameters as mean localisation error is improved when the parameters are optimised. Mean localisation error and mean confidence value have been improved by 52.94% and 32.14% respectively when the proposed algorithm's parameters are optimised.

Different underwater localisation methods (i.e., **USBL**, **ToA**-based trilateration and **IMU**-based **DR**) have been considered in the proposed localisation scheme. We sufficed to stochastically model the **USBL** and **IMU**-based **DR** localisation methods based on their error characteristics to show the proposed method is feasible. Random walker models have been considered to govern the mobility of the **AUVs** and hence **IMU**-based **DR** navigation method has been stochastically modelled. The proposed localisation algorithm offers a holistic approach in which a location-confidence value is associated with each **AUV** in the swarm. Given that each **AUV**'s confidence value is being updated on-the-fly, limited external navigation aid resources (i.e., **USBL**) can be then optimally utilised. Furthermore, based on the same confidence value a decision on whether an **AUV** broadcasts navigation aids is made. The proposed algorithm's performance in term of the entire swarm mean localisation error and standard deviation is compared with round-robin scheduling-based **USBL** aided **DR** navigation. The presented results strengthens the idea of cooperative localisation in large scale underwater robotic swarms

as the mean error of the entire swarm remains almost constant or slightly increases with increasing swarm size.

The proposed fuzzy-based localisation algorithm of Chapter 5 considers a fixed period of time for reference nodes (i.e., NB AUVs) to broadcast localisation aids. On the contrary, reference nodes in the proposed confidence-based localisation algorithm relies on a single indicator and a predetermined threshold (i.e., confidence value and confidence threshold) to broadcast localisation aids. Therefore, more localisation aids can be broadcasted and maximum acoustic channel utilisation can be achieved. A fixed number of reference nodes (i.e., NB AUVs) have been considered in fuzzy-based localisation algorithm (i.e., 10 AUVs). However, the number of reference nodes in the proposed confidence-based algorithm has been optimised through node density. The proposed confidence-based algorithm successfully eliminates the need for constructing and tuning a fuzzy system but it introduces some other parameters that have to be tuned due to its dependence on localisation methods error characteristics in penalising the confidence value. The confidence value in the following chapter will be updated intuitively based on one generic rule. Ocean bottom seismic imaging deployment scenario is simulated on the co-simulation platform for localisation performance analysis. Confidence threshold and the number of navigation beacons are then optimised based on the same multi-objective optimisation procedure presented in this chapter.

Chapter 7

Confidence based Localisation for Cooperative Underwater Robotic Swarms using the Extended Kalman Filter

A cooperative confidence-based localisation scheme for underwater robotic swarms using the [EKF](#) is proposed. A subset of the robots in the swarm is localised using ultra-short baseline technology, and another subset is configured to broadcast their location estimate (i.e., once this reaches a given confidence threshold) in order to act as range-only [Navigation Beacon \(NB\)](#)s. The confidence value for localisation precision in a single measurement is proposed to control the cooperation dynamics in the swarm in term of [USBL](#) localisation and navigation beacons aiding. Given the error characteristics of a commercially available [USBL](#) system and the covariance matrix of a trilaterated underwater vehicle position, [EKF](#)-based [USBL](#) or range measurements-aided dead reckoning navigation is performed and controlled by the [AUV](#)'s confidence value. We compare the performance of the proposed confidence-based localisation algorithm with both round-robin [USBL](#)-aided [EKF](#)-based [DR](#) navigating and the fuzzy-based localisation algorithm proposed in Chapter 5.

7.1 Introduction

Cooperative localisation of underwater multi-agent robotic system (i.e., 3-4 AUVs) has been investigated in [57], [114] and [116]. In cooperative underwater navigation, where server-client or transmitter-receiver architecture is normally considered, MAC protocol must be carefully designed to coordinate multiple transmitted acoustic signals with collision-free and maximum channel utilisation. TDMA is the most adopted MAC protocol in the subsea domain [130] while other multiple access technologies such as FDMA [182] and CDMA [132] are rarely utilised for underwater acoustic communications [183]. TDMA scheme required all active participants of the network to be synchronised; see [129] for a survey on TDMA schemes. However when clock-synchronisation is not available, CDMA, FDMA or CSMA schemes can be adopted [124]. A centralised cooperative localisation algorithm for synchronous-clock acoustic navigation has been reported in [57] using a centralised EKF. Webster et al. in [114] designed a decentralised cooperative navigation scheme to improve the scalability of the localisation scheme reported in [57] where they employed the EIF to reduce the transmitted information amount among the vehicles. In both [57] and [114], the authors considered one surface vehicle and two AUVs for cooperative localisation where the navigation relies on range measurements from a single beacon with known position. Likewise, in [116], Bahr et al. studied the cooperative localisation of two AUVs based on range measurements from a single beacon. Despite the paucity of underwater robotic swarm localisation research, a number of UWSN localisation techniques have already been investigated for ocean monitoring, geological and ecological research, and samples collection [167]. A localisation scheme for large scale UWSN integrates a 3-dimensional Euclidean distance estimation method with a recursive location estimation method introduced in [79]. Further gain in localisation accuracy was shown in [80], by incorporating sensors' mobility prediction model into the algorithm reported in [79]. The localisation scheme in both [79] and [80] is hierarchical where a localised sensor node serves as a localisation reference for non-localised sensor nodes if its associated confidence value is higher than a predetermined threshold. In chapter 6, we proposed a confidence-based underwater localisation for AUVs swarm in which confidence update rules were introduced based on the adopted localisation method's error characteristics and a simulation-based optimisation was carried out to determine an optimal confidence threshold.

A distributed cooperative localisation algorithm for underwater robotic swarm is proposed in which we utilise a USBL system for a subset of the swarm localisation. A

NB broadcasts a localisation message once its confidence value is higher than a pre-determined optimised confidence threshold. If another AUV with low confidence value receives three or more NB localisation messages (i.e., range-only) within a time window, unscented trilateration/multilateration is performed. The EKF is incorporated for Multi Data Sensor Fusion (MDSF). This chapter extends the confidence-based localisation algorithm proposed in Chapter 6 with the following contributions:

1. Incorporating the EKF for MDSF so that the confidence value of an AUV location can be intuitively generated of the associated covariance matrix.
2. A comprehensive analysis of the proposed algorithm based on simulating real-world deployment scenarios for ocean bottom seismic imaging using the co-simulation platform discussed in Chapter 4.
3. A comparison of the proposed algorithm performance with other navigation frameworks such as round-robin EKF-based cooperative localisation and the proposed fuzzy-based localisation in Chapter 5.

The remainder of this chapter is organised as follows. Section 7.2 details the proposed localisation algorithm and its confidence update rules. Section 7.3 presents deployment scenarios and settings and reports performance evaluation of the proposed confidence-based localisation algorithm. Finally, this chapter is summarised in section 7.4.

7.2 Cooperative Swarm Localisation

In this section, the proposed underwater robotic swarm localisation algorithm is detailed. Two different localisation aids are considered in the proposed algorithm, namely USBL fixes and range-only measurements to NBs. Spherical Error Probability (SEP) [184] is employed as an indicator for the location estimate precision of an AUV.

7.2.1 Algorithm Overview

Consider a swarm of N AUVs denoted by a set $\mathcal{A} = \{a^1, a^2, \dots, a^N\}$ is on a mission. Consider we have a USBL system, which can localise only a limited number of AUVs in each of its TDMA frame. Assume each AUV is equipped with a medium range acoustic USBL transponder working at medium frequency band i.e., 20-40 kHz, a short range

acoustic modem working at high frequency band (i.e., 100-180 kHz) for intra-swarm communication, a 9-axis IMU, a depth sensor and a CSAC for clock-synchronisation of the AUVs. Range measurements are acquired by means of OWTT as all AUVs are accurately synchronised using CSAC.

The three-dimensional localisation problem can be converted into its 2D counterpart via projection onto the horizontal plane as all AUVs are equipped with a pressure sensor with accurate depth estimation [94]. A subset of the swarm \mathcal{A}_{NB} with a priori known IDs are dedicated NBs i.e., transmit localisation messages when they are precisely localised where $\mathcal{A}_{NB} \subseteq \mathcal{A}$. Localisation precision is determined by the SEP of the estimated position covariance matrix which we define as the AUV's confidence value. All AUVs are launched from known position $\hat{p}_{t_0}^i$ relative to the deployment vessel's position (i.e., independent of the GNSS accuracy) and hence their initial position's covariance matrix norm $\|\Sigma_{t_0}^i\|$ is close to zero.

Let δ_t^i be the confidence value of AUV a^i being at a certain position \hat{p}_t^i at time (t); δ_t^i is a scalar between 0 and 1. In the proposed algorithm, a localised navigation beacon (i.e., NB) with confidence value higher than a predefined confidence threshold λ_1 broadcasts a localisation aid to its neighboring AUVs. It keeps silent otherwise as it is considered as an unreliable navigation beacon i.e., need-to-be-aided. The AUVs are assumed to navigate towards their designated destinations using their on-board proprioceptive sensors based on their INS which is composed of the IMU hardware and navigation algorithms as explained in Section 3.2. Their confidence values are updated at each localisation step based on their estimated position's covariance matrix. Once the confidence value δ_t^i of a^i drops below a predefined confidence threshold λ_1 , a^i broadcasts a USBL localisation request in its USBL TDMA slot. The USBL can localise a limited number of AUVs in a single TDMA frame. Specifically, USBL transceiver broadcasts a localisation message to a^i in a time slot in its TDMA frame if the maximum number of the AUVs that can be localised in one TDMA frame is not reached. If the number of localisation requests received by the USBL transceiver is more than what it can be localised in one USBL TDMA frame, AUVs with the lowest confidence values are only considered with prioritising NBs i.e., $a^i \in \mathcal{A}_{NB}$. Alternatively, once the confidence value δ_t^i of a^i drops below λ_1 and if a^i receives three or more localisation messages from neighbouring NBs within a time window, a^i performs unscented multilateration, explained in subsection 7.2.3, for location and covariance estimation ($p_{LAT,t}^i, \Sigma_{LAT,t}^i$). Figure 7.1 depicts the localisation process of the proposed confidence-based localisation algorithm of each AUV a^i in the swarm in which the EKF is utilised for fusing and tracking the biases of the IMU's measurements in indirect feedback integration as explained in subsection 3.2.2. The

spherical linear interpolation utilised as a low pass filter to smooth the **AUV**'s orientation estimates by creating a sequence of quaternions that vary smoothly between every two estimates with a constant angular velocity [185]. A simple outlier detector, based on the travelled distance, is applied on both **USBL** and multilateration localisation aids before being fused.

It has been proven in Chapter 6 that optimising the confidence threshold and node density (varied by the communication range of each node) can enhance the proposed localisation algorithm's performance. We are in this chapter adopting a more generic and intuitive method to update the confidence value than that we proposed in Chapter 6. Moreover, practical implementations of the proposed confidence-based algorithm are considered in this chapter and hence utilises off-the-shelf acoustic communications modem for intra-swarm communication with a communication range that cannot be adjusted. Therefore, the confidence threshold (i.e., λ_1) in this chapter is optimised based on the proposed confidence update method and in the sense that the communication range cannot be adjusted, the number of **NBs** (i.e., λ_2) is optimised. Optimising the number of **NBs** in the swarm is essentially node density optimisation which is defined as the number of **NBs** within an **AUV**'s communication range. The same optimisation procedure presented in Chapter 6 are followed but the **USBL** utilisation is minimised and range measurements aiding (i.e., trilateration) utilisation is maximised. We acknowledge that the **USBL** utilisation should be maximised. However the optimisation is carried out on fixed swarm size of 50 **AUVs** and the resultant optimised parameters are going to be used for a scaled up swarm sizes i.e., 100 and 150 **AUVs**. Given that the **USBL** can navigationally aid only a limited number of **AUVs** in a short **TDMA** frame (i.e., 1 second), the **USBL** utilisation is minimised so that the average localisation error of the entire swarm does not dramatically increase with increasing swarm size. The parameters λ_1 and λ_2 are optimised on a fixed number of **AUVs** (i.e., swarm size) and then the performance of the proposed algorithm with the optimised parameters is evaluated on a scaled-up swarm sizes.

Our objectives are to minimise the entire swarm mean localisation error i.e., $f_1(\lambda_1, \lambda_2)$, the entire swarm standard deviation of the mean of each **AUV**'s instantaneous **Root Mean Square Error (RMSE)** ($1-\sigma$ error) i.e., $f_2(\lambda_1, \lambda_2)$, **USBL** utilisation i.e., $f_3(\lambda_1, \lambda_2)$ and intra-swarm communication packet loss i.e., $f_4(\lambda_1, \lambda_2)$ while maximising the entire swarm mean confidence value i.e., $f_5(\lambda_1, \lambda_2)$ and trilateration utilisation i.e., $f_6(\lambda_1, \lambda_2)$

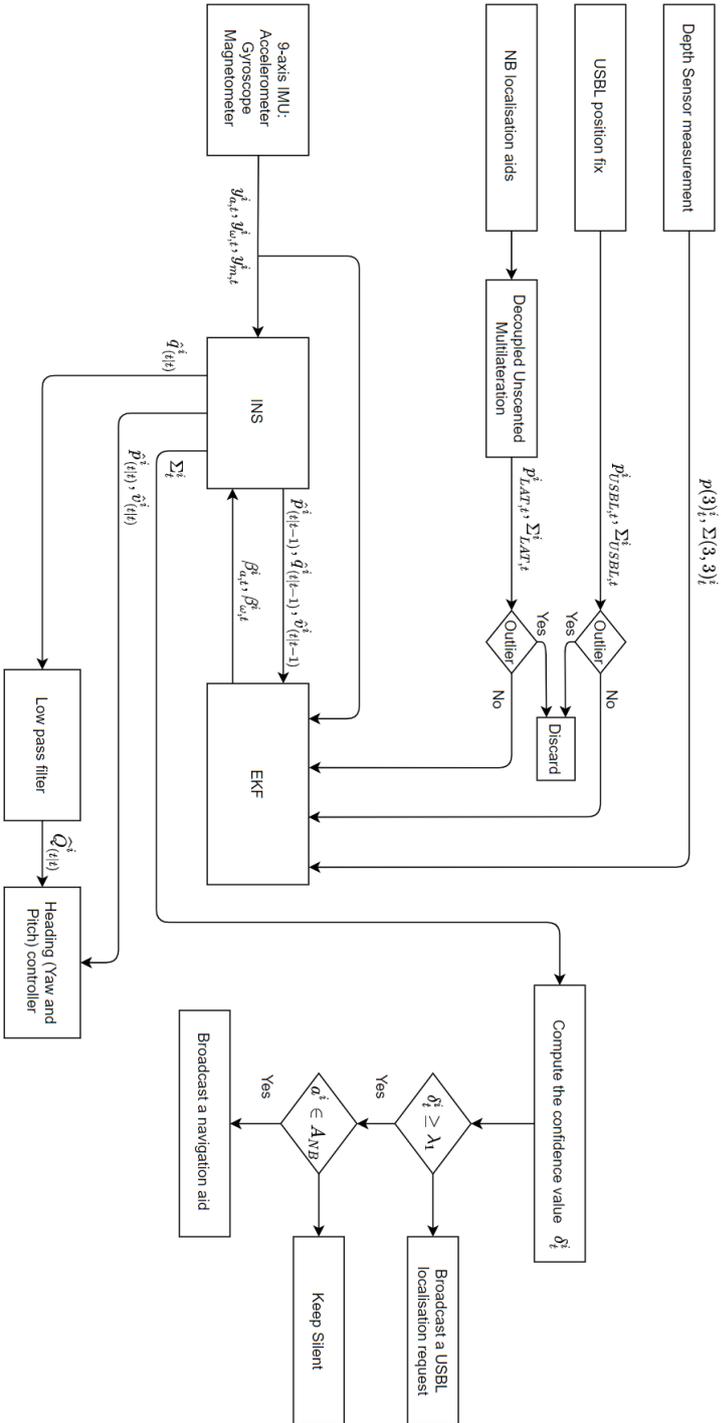


Figure 7.1: Navigation system block diagram of **AUV** a^i in a swarm of N **AUVs** where $y_{a,t}^i, y_{\omega,t}^i, y_{m,t}^i$ are the row readings of the accelerometer, gyroscope and magnetometer respectively and $\beta_{a,t}^i, \beta_{\omega,t}^i$ are the corresponding accelerometer and gyroscope measurements biases. $p_{[method],t}^i, \Sigma_{[method],t}^i$ are position aid and the associated covariance matrix where $p_{[method],t}^i \in \mathbb{R}^3 : p(1)_{[method],t}^i, p(2)_{[method],t}^i, p(3)_{[method],t}^i$ are North, East and Down coordinates respectively and the subscript $[method]$ is the localisation method i.e., USB L or Trilateration/Multilateration (LAT). $p(3)_t^i$ is the Down coordinate as returned from the depth sensor and $\Sigma(3, 3)_t^i$ is the corresponding variance. $\hat{p}_{(t|t-1)}^i, \hat{q}_{(t|t-1)}^i, \hat{\sigma}_{(t|t-1)}^i$ are the **AUV**'s prior estimate position, velocity and orientation respectively and $\hat{p}_{(t|t)}^i, \hat{q}_{(t|t)}^i, \hat{\sigma}_{(t|t)}^i$ are their posterior estimates.

as shown in Equation 7.1.

$$\begin{cases} \min f_1(\lambda_1, \lambda_2) \\ \min f_2(\lambda_1, \lambda_2) \\ \min f_3(\lambda_1, \lambda_2) \\ \min f_4(\lambda_1, \lambda_2) \\ \max f_5(\lambda_1, \lambda_2) \\ \max f_6(\lambda_1, \lambda_2) \end{cases} \text{ subject to } \begin{cases} L_1 \leq \lambda_1 \leq U_1 \\ L_2 \leq \lambda_2 \leq U_2 \end{cases} \quad (7.1)$$

where L_i and U_i ($i = 1, 2$) are lower and upper bounds of λ_1 and λ_2 respectively. The Fast and Elitist Multi-objective Genetic Algorithm NSGA-II [170], as shown in Chapter 6, is employed to find the Pareto-optimal solutions of Equation 7.1. The objective functions $f_1, f_2 \dots, f_6$ and are defined in the simulation as follows:

$$f_{1,3,5,6}(\lambda_1, \lambda_2) = \frac{\sum_{i=1}^N \left(\frac{\sum_{t=0}^T \Psi_t^i}{T} \right)}{N} \quad (7.2)$$

$$f_2(\lambda_1, \lambda_2) = \sqrt{\frac{\sum_{i=1}^N (\mathbf{s}^i - \mathbf{d})^2}{N}} \quad (7.3)$$

$$f_4(\lambda_1, \lambda_2) = \frac{\sum_{t=0}^{T_{Unet}} \mathfrak{p}_t}{T_{Unet}} \quad (7.4)$$

where N is the swarm size and T is the simulation maximum time step. Ψ_t^i in Equation 6.5 is defined as the localisation error for f_1 , true binary variable if USBL aiding is successful for f_3 , confidence value for f_5 and true binary variable if trilateration is successful for f_6 of AUV a^i at time instant t . \mathbf{s}^i and \mathbf{d} in Equation 7.3 are the mean localisation error of AUV a_i along the whole trajectory and the entire swarm mean localisation error respectively. \mathfrak{p}_t and T_{Unet} in Equation 7.4 is the entire swarm (i.e., network) packet loss at time instant t and computed in UnetStack as explained in subsection 4.1.1 and UnetStack maximum time step respectively.

7.2.2 Confidence Value Update

Three metrics are commonly used to describe the error modelled by multivariate Gaussian distribution with a single measurement, namely the covariance determinant, the

area/volume of the $3\text{-}\sigma$ error ellipse/ellipsoid and the ratio of the minor axis to the major axis of the error ellipse [186]. The first two metrics describe the overall error of the AUV's position in a single value, but it is hard to set a threshold using these metrics as they do not provide an intuitive interpretation of the localisation estimate precision. Moreover, the last metric does not capture the size of the error ellipse. None of these three metrics provides enough information about the localisation precision to be used as a confidence value. The **Circular Error Probability (CEP)** is proposed for the confidence value update. The CEP provides a more intuitive way to describe localisation confidence than the aforementioned alternatives. Shnidman in [184] defined the CEP as the probability of a realisation of a zero-mean, bivariate Gaussian distribution $\mathcal{N}(0, \Sigma)$ being within a circle with its centre at the origin and a radius of (R). The circle radius is considered as user-specified application's acceptable error. Shnidman reported an algorithm in [184] to compute the CEP. The concept is extended to 3D cases and the **Spherical Error Probability (SEP)** is computed given the covariance matrix $\Sigma \in \mathbb{R}^{3 \times 3}$ associated with each localisation estimate $\hat{p} \in \mathbb{R}^3$ of an AUV by the following:

$$P(R) = \frac{1}{\sigma_{xx} \sigma_{yy} \sigma_{zz} (2\pi)^{3/2}} \iiint_{x^2+y^2+z^2 \leq R^2} \exp\left(-\frac{1}{2} \left(\frac{x^2}{\sigma_{xx}^2} + \frac{y^2}{\sigma_{yy}^2} + \frac{z^2}{\sigma_{zz}^2}\right)\right) dx dy dz \quad (7.5)$$

Equation 7.5 can be simplified to:

$$P(R) = \frac{2}{\sigma_{xx} \sigma_{yy} \pi} \int_0^R \int_0^{\pi/2} r \exp\left(-\frac{r^2}{2} \left(\frac{\cos^2 \theta}{\sigma_{xx}^2} + \frac{\sin^2 \theta}{\sigma_{yy}^2}\right)\right) \operatorname{erf}\left(\frac{\sqrt{R^2 - r}}{\sqrt{2}\sigma_{zz}}\right) d\theta dr \quad (7.6)$$

where R is the acceptable localisation error (application dependable), $\operatorname{erf}(\cdot)$ is the error function and $P(R)$ is the confidence value δ . We attempt to optimise the confidence threshold λ_1 with acknowledging that the sphere's radius R pertains to the confidence value δ . Therefore, the sphere's radius R is set to constant in all simulation trials.

7.2.3 Multilateration in the Presence of Uncertainty

NB AUVs in the proposed algorithm are configured to broadcast navigation beacon aids based on their confidence values and a predefined confidence threshold. The confidence value is essentially a single value representation of the associated covariance matrix. The uncertainty in the navigational beacon's positions cannot be ignored in cooperative localisation when trilateration is adopted. Bahr et al. in [116] and [186] investigated the

uncertainty distribution associated with a trilaterated target position given the uncertainty in both navigation beacons' position and range measurements. The uncertainty distribution associated with the trilaterated target position is related to the Jacobian of the intersection function. In the case of a few navigation beacons e.g. 2-3 navigation beacons as in [116] and [186] a closed-form expression of the intersection function can be found and therefore the Jacobian can be directly derived. However, in the case of multilateration where navigation beacons are more than three with noisy range measurements, a closed-form expression of the target's position cannot be found. The maximum likelihood estimator of a target's position in this case is the solution of the following optimisation problem

$$\min_X \sum_{j=1}^J (r_j - \|A_j - X\|)^2 \quad (7.7)$$

where $j = \{1, 2, \dots, J\}$, J is the number of navigation beacons, r_j are range measurements from the unknown target's position X to the navigation beacon positions A_j . Problem 7.7 is nonconvex but guaranteed to have a global minimum [105]. Accurate estimates with fast convergence of Equation 7.7 can be provided by a stochastic optimisation techniques such as **PSO** as explained in 3.3.2. The most straightforward solution to investigate error propagation, given that a closed-form expression of the target's position cannot be derived, is to perform Monte-Carlo simulation to construct the covariance matrix of the trilaterated position. The main drawback of this approach is its high computational cost as Equation 7.7 will need to be solved $m^{(2 \times J)}$ times by **PSO**; where m is the Monte-Carlo sample number and J is the navigation beacons number. We instead select a few weighted samples of range measurements and navigation beacon positions using the **Unscented Transform (UT)** [178] to construct the covariance matrix of the target's trilaterated position. In the **UT**, $2L + 1$ weighted samples (i.e., sigma points) are selected from the Gaussian distribution $\mathcal{N}(\mu, \Sigma)$ and passes through a nonlinear function to find the mapped sigma points; where L is the Gaussian distribution dimension. Each sigma point S_k is associated with two weights ξ_k and ψ_k . The resultant Gaussian's parameters are extracted from the mapped weighted sigma points where ξ_k is used when computing the mean and ψ_k is used when recovering the covariance. For L -dimensional Gaussian with mean μ and covariance Σ , the resultant sigma point S_k are selected based on the following:

$$\begin{aligned} S_0 &= \mu \\ S_k &= \mu + (\sqrt{(L + \gamma)\Sigma})_k \quad \text{for } k = 1, \dots, L \\ S_k &= \mu - (\sqrt{(L + \gamma)\Sigma})_{k-L} \quad \text{for } k = L + 1, \dots, 2L \end{aligned} \quad (7.8)$$

Where $\gamma = \alpha^2(L + \kappa) - L$; α and κ are scaling parameters to determine how far the sigma points are spread from the mean.

The weights ξ_k and ψ_k associated with each sigma point S_k are computed as in the following:

$$\begin{aligned}\xi_0 &= \frac{\gamma}{L + \gamma} \\ \psi_0 &= \frac{\gamma}{L + \gamma} + (1 + \alpha^2 + \phi) \\ \xi_k = \psi_k &= \frac{1}{2(L + \gamma)} \quad \text{for } k = 1, \dots, 2L\end{aligned}\tag{7.9}$$

Where ϕ is a high-order parameter to encode further knowledge about the distribution e.g. $\phi = 2$ is chosen when the distribution is an exact Gaussian.

Assume $f(x_1, x_2)$ is a nonlinear function where x_1 and x_2 are independent variables with standard deviation of σ_{x_1} and σ_{x_2} respectively. The variance of $f(x_1, x_2)$ is given by

$$\sigma_f^2 = \left| \frac{\partial f}{\partial x_1} \right|^2 \sigma_{x_1}^2 + \left| \frac{\partial f}{\partial x_2} \right|^2 \sigma_{x_2}^2\tag{7.10}$$

Equation 7.10 shows that in the case of independent variables we can decouple the multilateration over each set of sigma points as shown in Figure 7.2 for fewer iterations and for straightforward extraction of the resulting Gaussian's parameters from the mapped sigma points. The number of iterations in one multilateration localisation run, when it is decoupled, is reduced from $(2L + 1)^{(2 \times J)}$ to $(2 \times J) \times (2L + 1)$ where J is the number of navigation beacons and L is the distribution dimension assuming that the probability distribution in range measurements has the same dimension of the probability distribution in navigation beacon positions.

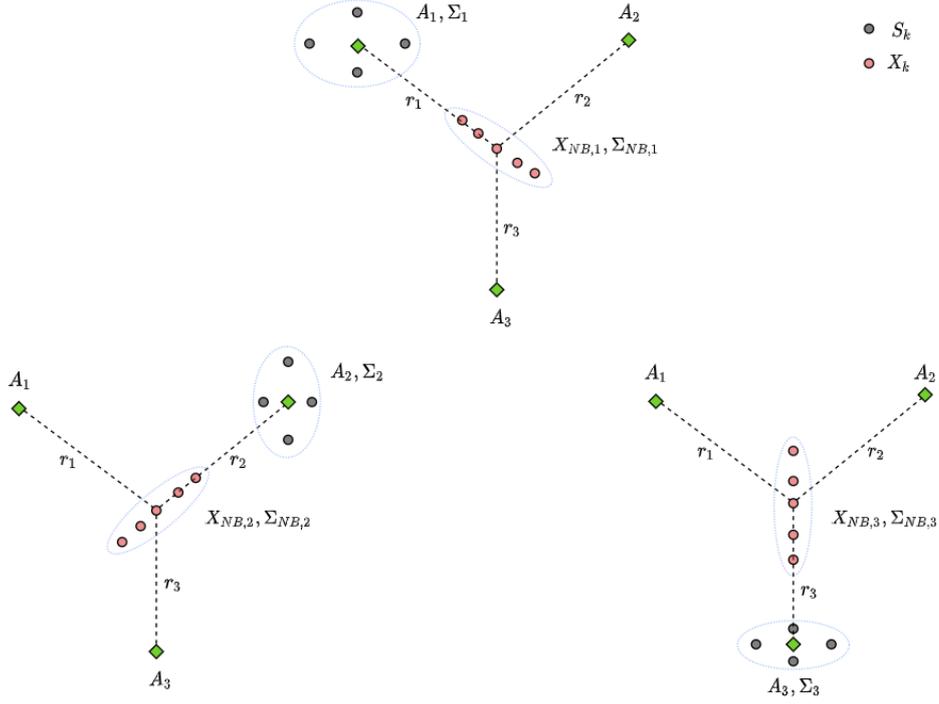


Figure 7.2: Decoupling the trilateration process over each set of sigma points. The same is carried out over range measurements sampling. Weighted sigma points S_K (in gray) through the unscented transform are selected to represent sufficient sampling of the navigation beacon's position and range measurement. Mean position $X_{NB,j}$ and the covariance matrix $\Sigma_{NB,j}$ are extracted from the resultant weighted trilaterated positions X_k (in red). The final mean position is the mean of all extracted mean positions and its final covariance matrix is the sum of all extracted covariance matrices.

The variables $X_{NB,j}$ and $\Sigma_{NB,j}$ represent the resultant Gaussian's mean and covariance respectively that can be extracted from the mapped sigma points X_k according to:

$$\begin{aligned}
 X_{NB,j} &= \sum_{k=1}^{2L+1} \xi_k X_k \\
 \Sigma_{NB,j} &= \sum_{k=1}^{2L+1} \psi_k (X_k - X_{NB,j}) (X_k - X_{NB,j})^\top
 \end{aligned} \tag{7.11}$$

The same applies over range measurements samplings to estimate $X_{RM,j}$ and $\Sigma_{RM,j}$. Let $\{\mathcal{A}_{NB}\}$ be navigation beacon whose localisation aids are received by AUV a^i within

a predefined time window and let $\{\mathcal{M}_{NB}\}$ and $\{\mathcal{M}_{RM}\}$ be their corresponding positions and range measurements respectively. Algorithm 2 shows the decoupled unscented multilateration process when a^i receives localisation aids from neighbouring NBs where the position, range measurement to a^i and covariance matrix of an NB a^j are denoted by p^j , d^j and Σ^j respectively.

Algorithm 2: Decoupled Unscented Multilateration of AUV a^i

Result: $p_{LAT}^i, \Sigma_{LAT}^i$

for each $p^j \in \mathcal{M}_{NB}$ **do**

Unscented Transform Sampling as in equations 7.8 and 7.9
 Solve equation 7.7 by PSO for each sample with $\{\mathcal{M}_{NB}\} \setminus p^j$
 Extract p_{NB}^j and Σ_{NB}^j as in equation 7.11

end

for each $d^j \in \mathcal{M}_{RM}$ **do**

Unscented Transform Sampling as in equations 7.8 and 7.9
 Solve equation 7.7 by PSO for each sample with $\{\mathcal{M}_{RM}\} \setminus d^j$
 Extract p_{RM}^j and Σ_{RM}^j as in equation 7.11

end

$$p_{LAT}^i = \text{mean}(p_{NB}^j, p_{RM}^j)_{\forall j}$$

$$\Sigma_{LAT}^i = \text{sum}(\Sigma_{NB}^j, \Sigma_{RM}^j)_{\forall j}$$

7.3 Simulation

7.3.1 Deployment Scenario and Settings

Each AUV is equipped with USBL transponder for long range communication operating at medium frequency band i.e., 20-40 kHz, 9-axis IMU, pressure gauge and CSAC for clock synchronization. Range measurements are acquired by means of OWTT as all AUVs are accurately synchronised. The error characteristics of the assumed USBL system can be found in [66]. Each AUV is also equipped with high-frequency acoustic modem for intra-swarm communication. We assume that either FDMA or CDMA MAC protocol is utilised to minimise the packet loss between the intra-swarm and USBL communication messages and within both intra-swarm and USBL communications TDMA MAC protocol is adopted. The modem S2CM-HS [22] specifications are adopted for intra-swarm communication and simulated on UnetStack as explained in Chapter 4. Table 7.1 lists the simulation settings and parameters for the intra-swarm ACOMMS.

Each **AUV** is deployed from known sea surface position and navigates towards a pre-

Parameter	Value
Communication modem Freq band	160 kHz
Communication data rate	50 kbit/s
Navigation aid length and duration	20 bytes; 3.2 ms
Navigation aid allocated TDMA time-slot length	20 ms
Noise level	60 dB
Water salinity	35 ppt
Water temperature	10 °C
Rician fading parameter	10
Fast fading	enabled

Table 7.1: Intra-swarm communication modem and channel parameters

determined seabed position. We are adopting the same deployment scenario in Section 5.4 where each **AUV** follows the shortest path to reach its seabed destination. A finite state machine has been implemented to guide each **AUV** to its destination as explained in Section 5.4. A deployment rate of a minimum of one **AUV** every second is considered for collision avoidance. All **AUVs** eventually form a grid on the seabed separated by at least 50 m from each other. Some of the **AUVs** with a priori known IDs are dedicated **NBs**. A **NB AUV** broadcasts localisation messages (i.e., navigation aids) when it is precisely localised i.e. when its confidence value is greater than a predetermined confidence threshold λ_1 . The broadcasted navigation aid by a **NB AUV** contains the broadcasting timestamp in microseconds, the **NB AUV's** north and east coordinates (i.e., $\hat{p}_t^i \in \mathbb{R}^2$) and the principal axis of the associated covariance matrix (i.e., $\text{diag}(\Sigma_t^i) \in \mathbb{R}^2$) in a data packet of 20 bytes such that 6 bytes for North and East coordinates, 8 bytes for North and East coordinates uncertainties and 6 bytes for the time stamp. Each navigation beacon has a time slot in a **TDMA** frame to broadcast their navigation aids and separated by guard intervals. The confidence threshold λ_1 and the **NB** subset size λ_2 are two key parameters and therefore they are optimised at first to enhance the proposed localisation algorithm's performance. Table 7.2 summarises simulation's key parameters to optimise the confidence threshold λ_1 and **NB** subset size λ_2 .

7.3.2 Results and Analysis

The proposed confidence-based localisation algorithm's parameters (i.e., λ_1 and λ_2) are optimised in a swarm size of 50 **AUVs** and then the algorithm's performance is analysed over scaled-up swarm sizes of 100 and 150 **AUVs**. The algorithm's performance is

Parameter	Value
Simulation time step	100 ms
Water depth	1000 m
Swarm size	50; 100; 150 AUVs
Deployment rate	1 AUV/s
Pressure sensor	2 Hz, 0.1m 1- σ error
Confidence threshold	[0.1, 0.9]
NB subset size (percentage of swarm size)	[10, 40]%
Sphere's radius for SEP (R)	5 m
USBL transponder communication range	6000 m
USBL localization accuracy in 1000 m	2.7 m 1- σ error
Max number of USBL localised AUVs in a single TDMA frame	10 AUVs
USBL TDMA Frame length	1 s
USBL update rate	4 s
NSGA-II population size	1000
NSGA-II max generation No.	500
NSGA-II non-dominated Fraction	0.015

Table 7.2: Simulation parameters

measured over the entire swarm with respect to the six aforementioned performance indicators i.e., localisation mean error, standard deviation, mean confidence value, USBL utilisation, trilateration utilisation and packet loss in the intra-swarm communication channel. The confidence threshold λ_1 was varied from 0.1 to 0.9 with an increment of 0.1 and the number of NBs λ_2 in the swarm was varied from 10% to 40% of the swarm size with an increment of 10%.

Similar to the optimisation procedures followed in Section 6.4, the fitness function of each objective has been built based on data fitting models of the objective function surfaces. The evolutionary multi-objective optimisation method NSGA-II is then employed to find the optimised confidence threshold $\lambda_1 : 0.1 \leq \lambda_1 \leq 0.9$ and number of NB subset size represented by a percentage of the swarm size $\lambda_2 : 10 \leq \lambda_2 \leq 40$. The upper bound of λ_2 40% is equivalent to 60 AUVs in a swarm size of 150 AUVs which means that the maximum length of the TDMA cycle would be 1.2 seconds given that each NB has a time slot of 20 milliseconds to broadcast its navigation aid. Figure 7.3 shows the suggested optimised parameters for λ_1 and λ_2 (i.e., Pareto-optimal set) and Figure 7.4 shows the Pareto Front i.e., Pareto-optimal set score in the objectives space.

Any of the Pareto-optimal solution can be selected to run the proposed confidence based localisation algorithm. A decision maker may have to gain in some objectives and sacrifice others, it solely depends on the objectives priority i.e., based on mission requirements.

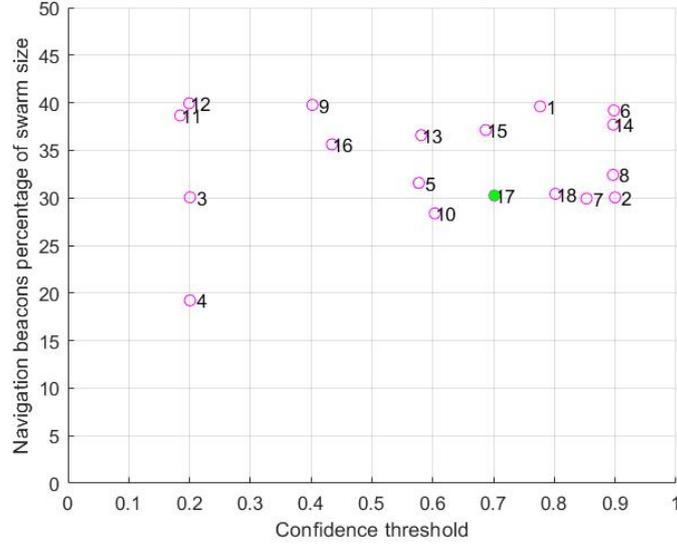


Figure 7.3: The corresponding Pareto-optimal set of Pareto Front (in confidence threshold and number of navigation beacons)

Our objectives are to:

- Minimise mean localisation error while i) maximising mean confidence value and trilateration utilisation and ii) minimising [USBL](#) utilisation, error standard deviation and packet loss of intra-swarm communication channel
- Minimise packet loss of intra-swarm communication channel while maximising trilateration utilisation.

In case of irreconcilable objectives, mean localisation error minimisation is prioritised over all of the rest of the objectives and trilateration utilisation maximisation is prioritised over packet loss minimisation. It is worth mentioning that packet loss of intra-swarm communication channel can be minimised due to minimal intra-swarm communication, for example when the confidence threshold is too high i.e., $\lambda_1 \geq 0.9$ intra-swarm communication is minimised as [NBs](#) do not broadcast navigation aids unless their confidence value is higher than the confidence threshold. Maximising trilateration utilisation while minimising packet loss ensures that packet loss minimisation is not caused by lack of intra-swarm communication.

It can be noticed from [Figure 7.4a](#) that the optimised solution number 3 (O_3) minimises the mean localisation error but it does not minimise the standard deviation. However

solution O_9 minimises the standard deviation but the associated packet loss of solution O_9 is around 70% and the same goes for solution O_{16} . It can be seen that solutions O_{17} , O_{15} , O_5 and O_1 keep balance between the mean error and all other objectives. Solution O_1 outperforms solutions O_5 , O_{17} and O_{15} in the mean confidence value, trilateration utilisation and packet loss but it does not outperform them in the mean localisation error and **USBL** utilisation. On the other hand, solution O_{17} has a similar performance to solution O_{15} and it outperforms solution O_5 in the packet loss and trilateration utilisation. Solution O_{17} as shown in Figure 7.3 requires less number of **NBs** than those required in solutions O_1 and O_5 . We therefore select the algorithm's parameters suggested by solution O_{17} which are around 0.7 for the confidence threshold (i.e., λ_1) and around 30% of the swarm size for **NB** subset size i.e., λ_2 .

The proposed confidence-based localisation algorithm is evaluated with the selected optimised parameters λ_1 and λ_2 on scaled-up swarm sizes of 100 and 150 **AUVs**. The performance of the proposed confidence-based localisation algorithm is compared to both the Fuzzy-based localisation algorithm proposed in Chapter 5 and round-robin **EKF**-based localisation in Figure 7.5. The optimised number of **NBs** (λ_2) have been considered retrospectively in both Fuzzy-based localisation and round-robin **EKF**-based localisation methods for performance comparisons with the proposed confidence-based localisation. Figure 7.5 shows the entire swarm mean localisation error and $2\text{-}\sigma$ standard deviation. The standard deviation in Figure 7.5 is the entire swarm standard deviation of each **AUV**'s mean localisation error along its whole trajectory. Figure 7.5 shows that the proposed confidence-based localisation method outperforms both the fuzzy-based localisation and round-robin-based method in the mean localisation error of the entire swarm by around 26.69% and 20.70% respectively at swarm size of 50 **AUVs**. The proposed confidence-based localisation algorithm outperforms both the fuzzy-based and round-robin-based localisation methods by 62.3% and 59.77% respectively in the entire swarm mean localisation error and by 81.62% and 82.19% in the entire swarm $1\text{-}\sigma$ standard deviation at swarm size of 100 **AUVs**. The entire swarm mean localisation error improved by 67.10% and 59.28% and the entire swarm $1\text{-}\sigma$ standard deviation improved by 79.27% and 72.04% when the proposed confidence-based localisation algorithm performance is compared to the fuzzy-based and round-robin-based methods respectively at swarm size of 150 **AUVs**.

One-tail two sample t -test is conducted to compare the localisation accuracy performance of each simulation trial in both round-robin **EKF**-based method and the proposed confidence-based localisation method. The null hypothesis is rejected at 0.05 significance

level with p-values of 9.51×10^{-11} , 7.42×10^{-18} and 8.03×10^{-16} at swarm sizes of 50, 100 and 150 AUVs respectively. Based on the one tail test, mean localisation error of the proposed confidence-based localisation method is lower than that of the EKF-based method at significance level of 0.05. Table B.1 in Appendix B.1 shows p-values, degree of freedom, t -statistics and critical values of each simulation trial to statistically compare the localisation accuracy of the proposed confidence-based localisation method with the EKF-based method.

Figure 7.6 shows a comparison of the instantaneous RMSE of a typical AUV performance in a swarm of 150 AUVs when the proposed parameters-optimised confidence-based or round-robin EKF-based localisation is adopted. It can be noticed from Figure 7.6 that the AUV receives more navigation aids when the the confidence-based localisation algorithm is adopted. This is partly due to minimising packet loss and maximising trilateration utilisation. The average packet loss in the intra-swarm communication channel when the proposed parameters-optimised confidence-based localisation algorithm is adopted at different swarm sizes was consistent i.e., 41-42% at 50, 100 and 150 AUVs, whereas it was about 73-78% when round-robin EKF-based localisation algorithm is adopted.

It can be noticed from Figure 7.5 that round-robin EKF-based localisation method outperforms fuzzy-based localisation method as opposed to the conclusion that has been drawn in Chapter 5. It is worth reminding the readers that the number of navigation beacons in this simulation set was optimised for the confidence-based localisation method and applied retrospectively for both fuzzy-based and round-robin EKF-based localisation methods. Whereas the number of navigation beacons in Chapter 5 simulation was arbitrarily set to 10 AUVs. Fuzzy set parameters and rules can be further tuned in this simulation set so that the performance of fuzzy-based localisation method is similar (or outperforms) to that achieved in round-robin EKF-based localisation method.

Figure 7.7 shows the instantaneous RMSE with $\pm 3\sigma$ error bounds of an AUV in a swarm of 150 AUVs when the proposed localisation algorithm is adopted.

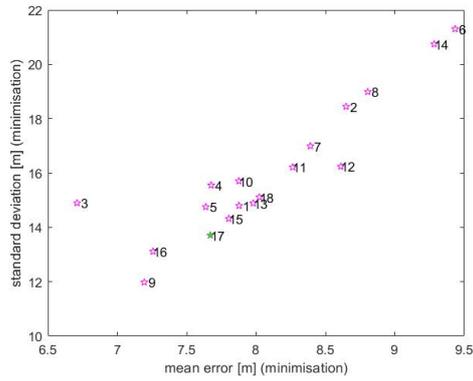
7.4 Summary

In summary, a confidence-based underwater swarm localisation using the EKF is proposed. Each AUV in the swarm has its own navigational suite which includes acoustic communication modems, depth sensor and 9-axis IMU. The confidence threshold and NB subset size are two key parameters of the proposed algorithm and therefore they

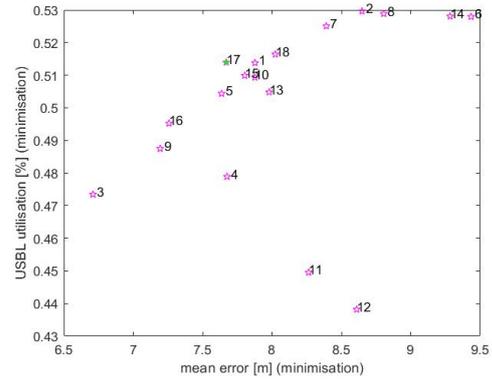
are optimised through extensive simulation using the Fast and Elitist Multi-objective Genetic Algorithm NSGA-II [170]. The confidence threshold and NB subset size parameters are optimised at fixed number of AUVs i.e., 50. The localisation accuracy of the proposed algorithm with the optimised parameters is then evaluated on different swarm sizes i.e., 100 and 150 AUVs. Navigation beacons broadcast localisation aids whenever their confidence value is higher than the confidence threshold. If an AUV with low confidence value receives navigation aids from neighbouring NB AUVs, it performs unscented multilateration and fuses its location estimate using the EKF.

USBL transceiver are placed near the sea-surface and therefore a low frequency, long range communication modems are dedicated for the USBL system. However, a high frequency, short range acoustic communication modems are dedicated for intra-swarm communication to maximise the utilisation of the underwater ACOMMS channel. The intra-swarm communication is simulated on UnetStack [142]; The environment, the AUV physics are simulated on Webots [136].

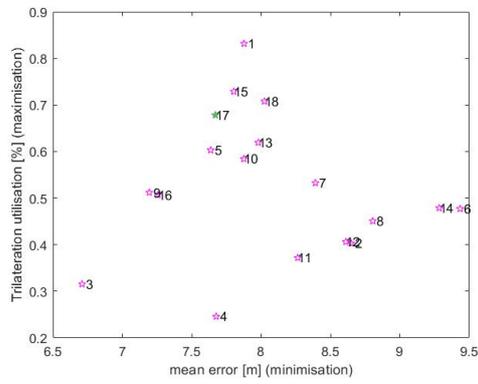
The proposed algorithm has significantly improved the entire swarm mean localisation error by 67.10% and 59.28% when compared to the fuzzy-based and round-robin EKF based localisation methods respectively at swarm size of 150 AUVs. The proposed algorithm enhanced the overall localisation accuracy of the entire swarm by minimising packet loss in the intra-swarm communication channel with maximising intra-swarm cooperation (i.e., trilateration/multilateration utilisation) through multi-objective optimisation of confidence threshold and number of navigation beacons. Packet loss in intra-swarm communication channel dropped by around 30% when the proposed confidence-based localisation method was adopted for swarm localisation as opposed to round-robin EKF-based localisation method.



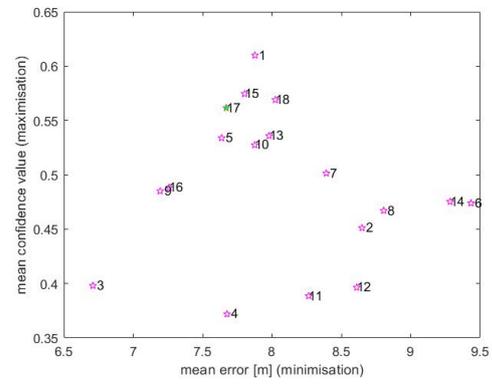
(a)



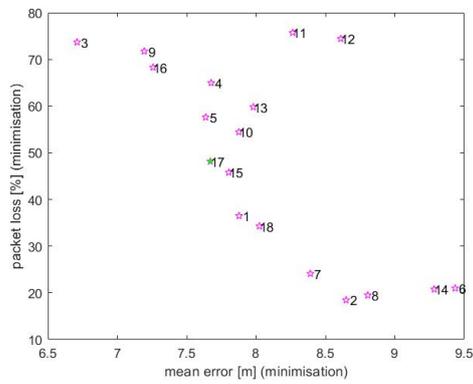
(b)



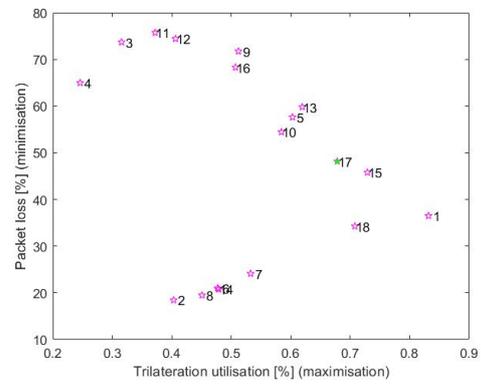
(c)



(d)



(e)



(f)

Figure 7.4: The score of Pareto-optimal set, Pareto Front, in (a) mean error and standard deviation (b) mean error and USBL utilisation (c) mean error and trilateration utilisation (d) mean error and mean confidence value (e) mean error and intra-swarm communication packet loss and (f) trilateration utilisation and intra-swarm communication packet loss. The solutions in Pareto front are numbered from 1 to 18. Solution O_{17} is selected for the algorithm's parameters.

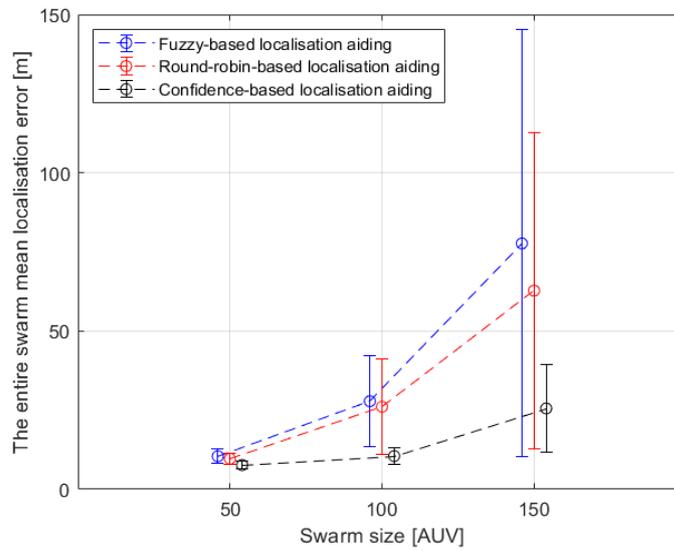


Figure 7.5: The entire swarm mean and standard deviation localisation error in Fuzzy-based USBL/trilateration aided DR navigation (in blue), Round-robin-based USBL-aided DR navigation (in red) and the proposed confidence-based USBL/trilateration aided DR navigation (in black) at swarm size of 50, 100 and 150 AUVs. The error bar around the mean point represents 2- σ standard deviation.

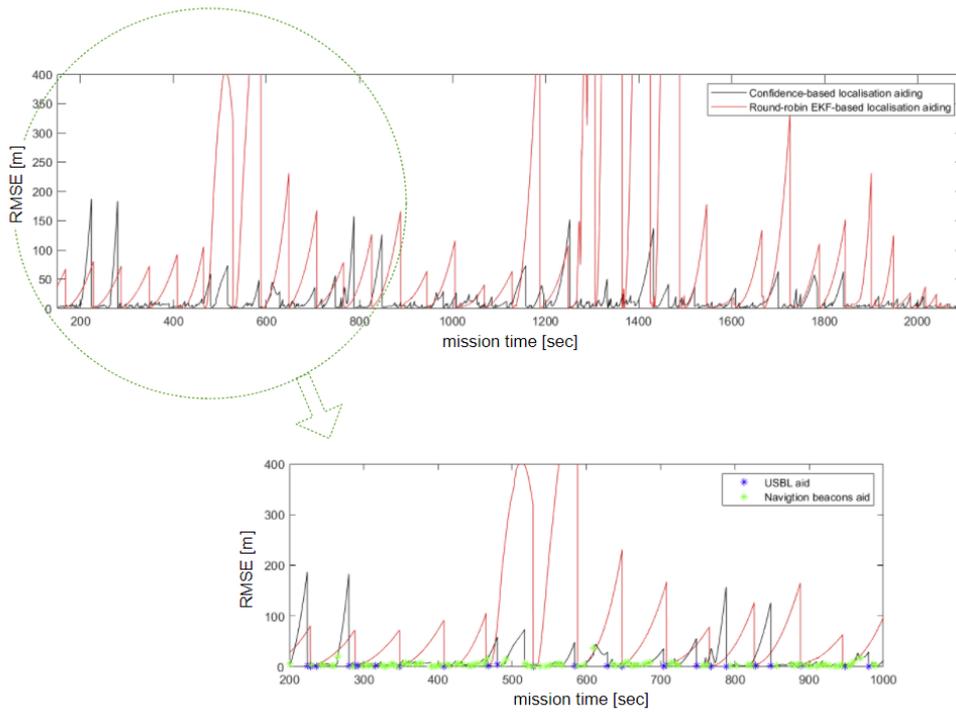


Figure 7.6: Instantaneous root mean square localisation error of a typical AUV in a swarm of 150 when the proposed parameters-optimised confidence-based and fuzzy-based localisation methods are adopted. A time window of the first 1000 seconds of the mission time shows the fused external navigation aid.

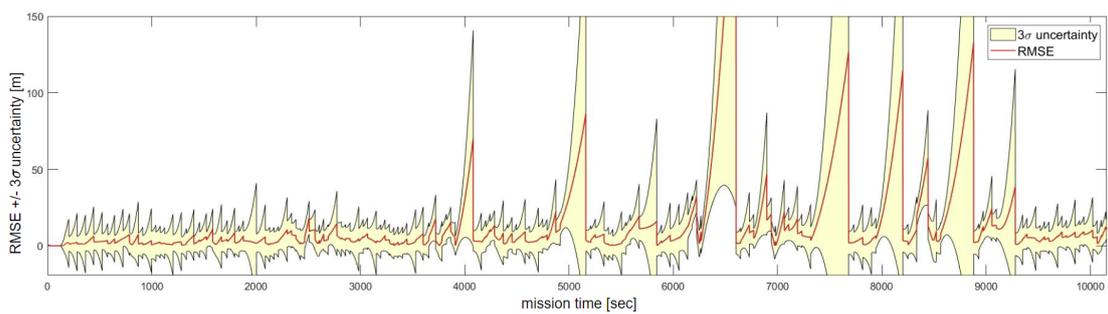


Figure 7.7: An AUV's RMSE with 3σ error bounds in a swarm of 150 AUVs when the proposed parameters-optimised confidence-based localisation algorithm is adopted.

Chapter 8

Conclusion and Future Work

This chapter summarises the main conclusions resulting from this body of work and discusses future directions and recommendations to potentially further improve the proposed cooperative navigation methods for underwater robotic swarms.

This thesis contributes to the advancement of the state-of-the-art in underwater navigation approaches for robotic swarms. In particular, the proposed cooperative navigation methods in this thesis address: 1) implementation and validation of a holistic navigation framework that is suitable for underwater robotic swarms and flexible enough to accommodate newly introduced localisation methods; 2) implementation and validation of an optimised confidence based navigation algorithm for underwater robotic swarms aims to control the cooperation dynamics in the swarm in a fully distributive fashion and optimise the acoustic channel utilisation.

8.1 Summary

Swarm robotics have become an attractive research area due to its capability of providing relatively low-cost solutions for complicated tasks in many applications, especially when wide region coverage is required. Locations of each individual node must be known and tracked during operation for location-aware applications e.g., marine geology, seismic imaging, environmental monitoring and exploration. A swarm of AUVs utilises acoustic channels for communication that can be deployed for ocean bottom seismic imaging is considered in this thesis. Radio signals in underwater environment suffer from high attenuation that prevents it from penetrating water bodies hence the absence of the

GNSS in underwater environment. The high cost of marine assets, the severely limited bandwidth and long latency of underwater acoustic communication limit the number of **AUVs** that can be deployed at once to collaboratively complete a mission. Therefore, most terrestrial and aerial swarm localisation algorithms reported in the literature are not directly applicable to underwater robotic swarms. A fuzzy-based localisation and a confidence-based localisation methods for underwater cooperative robotic swarms are proposed, the performance of both localisation methods are first validated on a simple proof-of-concept simulation platform and then a sophisticated simulation platform is implemented for further evaluation. A physics-based robotic co-simulation platform that considers the underwater acoustic communications characteristics is developed to validate the proposed localisation algorithms for cooperative underwater robotic swarms. The developed co-simulation platform is composed of three main elements, namely Webots robotic simulator for **AUVs** physics and hydrodynamics simulation, UnetStack for intra-swarm acoustic communications simulation and MATLAB Navigation toolbox for industrial grade **IMU** simulation.

A fuzzy-based navigation framework was proposed to enhance the entire swarm localisation accuracy by providing the nodes with external navigation aids more frequently. The proposed fuzzy-based aiding has improved the entire swarm mean localisation error and standard deviation by 16.53% and 35.17% respectively at swarm size of 150 **AUVs** when its compared to round-robin **EKF**-based **USBL**/trilateration-aided **DR** navigation. The proposed fuzzy-based algorithm performance is emphasised on large swarm sizes. Mamadani fuzzy inference system has been adopted for the localisation problem due to its intuition, simplicity and well-suitability to human input. The proposed navigation framework can accommodate other external navigation aid methods such as **DVL** aid by simply expanding the fuzzy rule base. Expanding the fuzzy rule base to integrate other external navigation aids is relatively easy compared to Kalman filter-based integration. Substantial efforts are needed when integrating new localisation methods in an existent **EKF**-based navigation for two reasons: Gaussian error modelling; and the entire filter re-implementation to expand the covariance matrix and the state vector.

A confidence-based localisation algorithm for underwater robotic swarm is proposed in which a **ToF**-acoustic navigation aid (i.e., **USBL**) is utilised for a subset of the **AUV** swarm localisation based on their confidence value and the **EKF** is adopted for **MDSF**. The proposed confidence value represents the localisation uncertainty in a single scalar measurements. **AUVs** with confidence values higher than a predefined confidence threshold broadcast navigation aids i.e., navigation beacons. If an **AUV** with low confidence

value passively receives three or more navigation aids with a predefined time window, multilateration that considers both range measurements and navigation beacon positions uncertainties is performed. To ensure an optimal utilisation of the acoustic channel:

- 1) Only a subset of the swarm is configured to broadcast navigation aids i.e., **NB AUV**;
- 2) A predefined confidence threshold is set to let an **NB AUV** to broadcast navigation aids and the same confidence threshold is used by all swarm nodes to request a **USBL** navigation aid.
- 3) The proposed confidence-based algorithm's parameters i.e., confidence threshold and **NB** subset size are optimised using the **EMO** algorithm NSGA-II [170] for localisation error minimisation and localisation precision maximisation.

The proposed confidence-based algorithm has significantly improved the entire swarm mean localisation error by 62.3% and 67.10% when it is compared to the fuzzy-based localisation method and by 59.77% and 59.28% when it is compared to round-robin **EKF**-based localisation method at swarm sizes of 100 and 150 **AUV**s respectively.

The robustness of the proposed cooperative localisation methods is ensured by the four robustness factors of swarm systems [9]. These are:

- 1) Redundancy; that is, malfunction or loss of any **AUV** does not impact the navigation performance of the entire swarm.
- 2) Decentralisation; that is, the entire swarm navigation does not rely on a single command centre and destroying part of the swarm does not halt the navigation algorithm.
- 3) Simplicity; that is, the decision making process is relatively simple as each **AUV** takes its own decision.
- 4) Multiplicity of sensing; that is, the navigation suit of each **AUV** relies on multiple sensory data.

8.2 Future Directions

Task decomposition, negotiation and allocation is a major challenge in cooperative robotics. The global aim of a sub-sea mission e.g., deep sea exploration and sub-sea structure scanning is given by the operator at the control and command station. These missions, either for each robot to conduct them on its own or conduct them in cooperation, are with different degrees of complexity. In case of cooperation, every robot receives the same mission. A mission then must be set in ordered tasks with sub-goals to be attained. Tasks are allocated and can be executed in the most efficient manner by considering the local knowledge of each robot's environment and each robot's capability i.e., in case of a heterogeneous robotic swarm. The problem of optimally evaluating

and allocating tasks is not trivial in a distributive sense when the mission to be accomplished by the robots involves collaboration. Several algorithms based on contract nets protocol (i.e., contract-based negotiation) can be found in the literature for multiple robot task allocation problems [187, 188]. The negotiation has a common structure 1) distributed negotiation, not centralised and operates locally at each robot level 2) Two-way communications between robots to exchange information 3) each node evaluates the information from its own perspective and 4) final agreement among nodes is achieved by mutual selection based on contract-based or auction-based algorithms [189]. The proposed cooperative localisation algorithm considers a homogeneous robotic swarm with a fixed set of navigation beacons. A negotiation-based algorithm can be adopted to select the navigation beacons in the swarm to further improve the localisation accuracy. In addition, optimal path planning for the surface vehicle where the USBL is mounted can greatly enhance the localisation accuracy of the underwater vehicles [190]. This can be further improved by considering multiple surface vehicles for the ToF acoustic navigation aid and utilise an auction-based algorithm to optimise each surface vehicle position when transmitting navigation aids for the underwater vehicles. Figure 8.1 shows an example of task allocation scheme in a heterogeneous robotic swarm of AUVs, USVs and static sensor nodes.

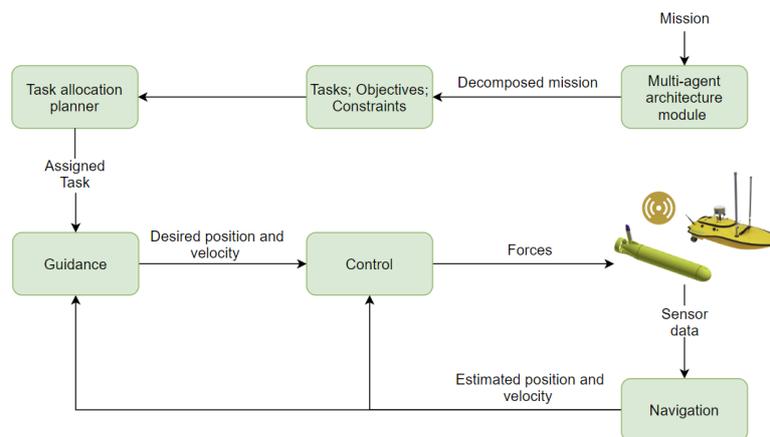


Figure 8.1: Task allocation in a heterogeneous robotic swarm of AUVs, USVs and static sensor nodes.

The proposed cooperative localisation algorithm considers OWTT for range measurements among the AUVs together with TDMA for cooperative localisation as all AUVs are accurately synchronised. However, it is always possible that a few AUVs' clocks

fall out of synchronisation. Therefore, adopting [TWTT](#) for ranging and self-organising [TDMA](#) that does not heavily rely on time synchronisation can possibly further tune the cooperation dynamics in the proposed localisation algorithm. Although, [TWTT](#) does not require time synchronisation among [AUVs](#) but the minimum update rate for an [AUV](#) using [TWTT](#) ranging is the cumulative sum of the [TWTT](#) for each [AUV](#).

Another major challenge for marine robotic systems is the presence of disturbances caused by sea currents and waves while retrieving [AUVs](#) into structures that may be moving. Station-keeping of [AUVs](#) is crucial specially in [AUVs](#) retrieval scenarios that require [AUVs](#) docking. Vision-based and ocean's currents model-based station keeping algorithm has been reported in the literature [[191](#), [192](#)]. The presence of disturbances is neglected in the current implementation based on the assumption that [AUVs](#) deployment is taking place from at least 20 meters below sea-surface where ocean's water column layer is almost static. This assumption is reasonable for deep sea deployment however, the presence of disturbances caused by waves and sea currents cannot be neglected in retrieval scenarios. Guiding the [AUVs](#) to a docking station, for retrieval, that is lowered from a surface vessel requires careful consideration of these interactions to improve the controllability and station-keeping of the [AUVs](#) and the docking station.

Bibliography

- [1] Sverdrup HU, Johnson MW, Fleming RH, et al. *The Oceans: Their physics, chemistry, and general biology*. vol. 7. Prentice-Hall New York; 1942.
- [2] Ballard RD, Hively W. *The eternal darkness: a personal history of deep-sea exploration*. vol. 50. Princeton University Press; 2017.
- [3] Gage JD, Tyler PA. *Deep-sea biology: a natural history of organisms at the deep-sea floor*. Cambridge University Press; 1991.
- [4] Smit F, Perkins C, Lepre L, Craft K, Woodard R. Seismic data acquisition using ocean bottom seismic nodes at the Deimos Field, Gulf of Mexico. In: *SEG Technical Program Expanded Abstracts 2008*. Society of Exploration Geophysicists; 2008. p. 998–1002.
- [5] IOGP. An overview of marine seismic operations. The International Association of Oil Gas producers; 2011. 448. Available from: <https://www.iogp.org/bookstore/product/an-overview-of-marine-seismic-operations/>.
- [6] CGG. Passion for Geoscience;. Available from: <https://www.cgg.com/en>.
- [7] SAAB;. SAAB SEAEYE. Available from: <https://www.saabseaeye.com/>.
- [8] ecoSUB;. EcoSUB Robotics. Available from: <https://www.ecosub.uk/>.
- [9] Şahin E. Swarm robotics: From sources of inspiration to domains of application. In: *International workshop on swarm robotics*. Springer; 2004. p. 10–20.
- [10] Chen J, Gauci M, Li W, Kolling A, Groß R. Occlusion-based cooperative transport with a swarm of miniature mobile robots. *IEEE Transactions on Robotics*. 2015;31(2):307–321.
- [11] Arafat MY, Moh S. Localization and Clustering Based on Swarm Intelligence in UAV Networks for Emergency Communications. *IEEE Internet of Things Journal*. 2019;6(5):8958–8976.
- [12] Wang X, Feng W, Chen Y, Ge N. UAV Swarm-Enabled Aerial CoMP: A Physical Layer Security Perspective. *IEEE Access*. 2019;7:120901–120916.
- [13] Chung S, Paranjape AA, Dames P, Shen S, Kumar V. A Survey on Aerial Swarm Robotics. *IEEE Transactions on Robotics*. 2018;34(4):837–855.
- [14] Giuggioli L, Potts JR, Rubenstein DI, Levin SA. Stigmergy, collective actions, and animal social spacing. *Proceedings of the National Academy of Sciences*. 2013;110(42):16904–16909.
- [15] Ferri G, Munafò A, Tesei A, Braca P, Meyer F, Pelekanakis K, et al. Cooperative robotic networks for underwater surveillance: an overview. *IET Radar, Sonar Navigation*. 2017;11(12):1740–1761.

- [16] Petillot YR, Antonelli G, Casalino G, Ferreira F. Underwater Robots: From Remotely Operated Vehicles to Intervention-Autonomous Underwater Vehicles. *IEEE Robotics Automation Magazine*. 2019;26(2):94–101.
- [17] Tal A, Klein I, Katz R. Inertial navigation system/Doppler velocity log (INS/DVL) fusion with partial DVL measurements. *Sensors*. 2017;17(2):415.
- [18] Snyder J. Doppler Velocity Log (DVL) navigation for observation-class ROVs. In: *OCEANS 2010 MTS/IEEE SEATTLE*. IEEE; 2010. p. 1–9.
- [19] Stojanovic M, Preisig J. Underwater acoustic communication channels: Propagation models and statistical characterization. *IEEE Communications Magazine*. 2009 January;47(1):84–89.
- [20] Napolitano F, Chapelon A, Urgell A, Paturel Y. PHINS: The autonomous navigation solution. *Sea Technology*. 2004;45(2):55–58.
- [21] Medagoda L, Kinsey JC, Eilders M. Autonomous Underwater Vehicle localization in a spatiotemporally varying water current field. In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*; 2015. p. 565–572.
- [22] EvoLogics. HS Devices;. Communication and positioning devices. Available from: <https://evologics.de/acoustic-modem/hs>.
- [23] Grewal MS, Andrews AP. *Kalman filtering: Theory and Practice with MATLAB*. 2nd ed. John Wiley & Sons, Ltd; 2001.
- [24] Autonomous Robotics;. Flying Nodes - The future of Ocean Bottom Sensing. Available from: <http://autonomousroboticsltd.com/>.
- [25] OGTC;. The Oil and Gas Technology Centre. Available from: <http://www.theogtc.com/>.
- [26] Sabra A, Fung WK. Dynamic localization plan for underwater mobile sensor nodes using fuzzy decision support system. In: *OCEANS 2017 - Anchorage*; 2017. p. 1–8.
- [27] Sabra A, Fung WK. A Fuzzy Cooperative Localisation Framework for Underwater Robotic Swarms. *Sensors*. 2020 Sep;20(19):5496. Available from: <http://dx.doi.org/10.3390/s20195496>.
- [28] Sabra A, Fung W, Radhakrishna P. Confidence-based Underwater Localization Scheme for Large-Scale Mobile Sensor Networks. In: *OCEANS 2018 MTS/IEEE Charleston*; 2018. p. 1–6.
- [29] Sabra A, Fung WK, Churn P. Multi-objective Optimization of Confidence-Based Localization in Large-Scale Underwater Robotic Swarms. In: Correll N, Schwager M, Otte M, editors. *Distributed Autonomous Robotic Systems*. Cham: Springer International Publishing; 2019. p. 109–123.
- [30] Shell. Finding Oil and Gas;. Overcoming technology challenges in Energy and innovation. Available from: <https://www.shell.com/energy-and-innovation/overcoming-technology-challenges/finding-oil-and-gas.html>.
- [31] statista. statista, editor. <https://www.statista.com/>; Daily demand for crude oil worldwide from 2006 to 2021. Available from: <https://www.statista.com>.
- [32] Repsol. 3D oil;. Available from: https://www.repsol.com/imagenes/global/en/3D_Oil_tcm14-15322.pdf.
- [33] Che X, Wells I, Dickers G, Kear P, Gong X. Re-evaluation of RF electromagnetic communication in underwater sensor networks. *IEEE Communications Magazine*. 2010;48(12):143–151.

- [34] EvoLogics. 7/17 Devices;. Communication and positioning devices. Available from: <https://evollogics.de/acoustic-modem/7-17>.
- [35] Blondel P, Murton B. Handbook of seafloor sonar imagery. Wiley/Praxis Series; 1997.
- [36] Chevron. seismic imaging;. Available from: <https://www.chevron.com/stories/seismic-imaging>.
- [37] BBC. Seismic waves;. Available from: <https://www.bbc.co.uk/bitesize/guides/zswkjty/revision/3>.
- [38] Krohn CE, Chen S. Comparisons of downhole geophones and hydrophones. *Geophysics*. 1992;57(6):841–847.
- [39] USGS. 3D seismic profile animation;. U.S. Geological Survey. Available from: <https://www.usgs.gov/media/images/3d-seismic-profile-animation>.
- [40] Al-Khatib H, Antonelli G, Caffaz A, Caiti A, Casalino G, de Jong IB, et al. The widely scalable Mobile Underwater Sonar Technology (WiMUST) project: An overview. In: *OCEANS 2015 - Genova*; 2015. p. 1–5.
- [41] Al-Khatib H, Antonelli G, Caffaz A, Caiti A, Casalino G, de Jong IB, et al. Navigation, Guidance and Control of Underwater Vehicles within the Widely scalable Mobile Underwater Sonar Technology Project: an overview. *IFAC-PapersOnLine*. 2015;48(2):189 – 193. 4th IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles NGCUV 2015. Available from: <http://www.sciencedirect.com/science/article/pii/S2405896315002700>.
- [42] Reimagine Ocean monitoring and Operations;. Unmanned Robots Powered by Nature. Available from: <https://www.liquid-robotics.com/wave-glider/overview/>.
- [43] Carragher P, Hine G, Legh-Smith P, Mayville J, Nelson R, Pai S, et al. A new platform for offshore exploration and production. *Oilfield Review*. 2014;4(25):40–50.
- [44] Moldoveanu N, Salama A, Lien O, Muzyert E, Pai S, Monk D, et al. Marine acquisition using autonomous marine vehicles: A field experiment. In: *2014 SEG Annual Meeting*. Society of Exploration Geophysicists; 2014. .
- [45] Moldoveanu N, Combee L, Muzyert E, Pai S, Caprioli P, et al. Marine seismic acquisition with 3D sensor arrays towed by wave gliders. In: *2016 SEG International Exposition and Annual Meeting*. Society of Exploration Geophysicists; 2016. .
- [46] CGG. Node Acquisition. CGGVeritas; 2011. Available from: https://www.cgg.com/data/1/rec_docs/888_CGGV__Node_Acquisition_LR0ct08.pdf.
- [47] FairfieldNodal. Z3000 Specification Sheet. Fairfield Geotechnologies; 2016. Available from: <http://fairfieldgeo.com/media/pdfs/Z3000-spec-sheet-2018.pdf>.
- [48] Geosolutions S. MANTA - A Seismic Shift in OBN Efficiency. Seabed Geosolutions; 2019. Available from: <https://www.seabed-geo.com/sites/default/files/downloads/Manta-Spec-Sheet-2019-April-US-Letter-Web.pdf>.
- [49] Beaudoin G, Reasnor M, Pfister M, Openshaw G. First Wide-azimuth Time-lapse Seismic Acquisition Using Ocean Bottom Seismic Nodes at Atlantis Field–Gulf of Mexico. In: *72nd EAGE Conference and Exhibition-Workshops and Fieldtrips*. European Association of Geoscientists & Engineers; 2010. p. cp–161.

- [50] Saudi Arabian Oil Co;. Available from: <https://www.saudiarabco.com/>.
- [51] Holloway A, Grant D, Watts G, Beaudoin G, et al. The future of deepwater ocean bottom seismic—Are flying nodes the next big step? In: 2015 SEG Annual Meeting. Society of Exploration Geophysicists; 2015. .
- [52] Cheng W, Thaeler A, Cheng X, Liu F, Lu X, Lu Z. Time-synchronization free localization in large scale underwater acoustic sensor networks. In: 2009 29th IEEE International Conference on Distributed Computing Systems Workshops. IEEE; 2009. p. 80–87.
- [53] SBG Systems. Ellipse 2 Micro Series. Ellipse 2 Micro Series datasheet. 2018;(1.2):1–4. Available from: <https://www.sbg-systems.com/products/ellipse-micro-series/>.
- [54] IEEE Standard for Inertial Systems Terminology. IEEE Std 1559-2009. 2009;p. c1–30.
- [55] Kok M, Hol JD, Schön TB. Using Inertial Sensors for Position and Orientation Estimation. Foundations and Trends in Signal Processing. 2017;11(1-2):1–153. Available from: <http://dx.doi.org/10.1561/20000000094>.
- [56] NOAA. Geomagnetism;. National Centers for Environmental Information. Available from: <https://www.ngdc.noaa.gov/geomag/geomag.shtml>.
- [57] Webster SE, Eustice RM, Singh H, Whitcomb LL. Advances in single-beacon one-way-travel-time acoustic navigation for underwater vehicles. The International Journal of Robotics Research. 2012;31(8):935–950.
- [58] Simanek J, Reinstein M, Kubelka V. Evaluation of the EKF-Based Estimation Architectures for Data Fusion in Mobile Robots. IEEE/ASME Transactions on Mechatronics. 2015;20(2):985–990.
- [59] Tang Y, Wu Y, Wu M, Wu W, Hu X, Shen L. INS/GPS Integration: Global Observability Analysis. IEEE Transactions on Vehicular Technology. 2009;58(3):1129–1142.
- [60] Barfoot TD. State estimation for robotics. Cambridge University Press; 2017.
- [61] Bloesch M, Sommer H, Laidlow T, Burri M, Nuetzi G, Fankhauser P, et al. A primer on the differential calculus of 3d orientations. arXiv preprint arXiv:160605285. 2016;.
- [62] Giroux R, Gourdeau R, Landry R. Extended Kalman filter implementation for low-cost INS/GPS Integration in a Fast Prototyping Environment. In: Symposium on Navigation of the Canadian Navigation Society; 2005. p. 1–11.
- [63] Morgado M, Oliveira P, Silvestre C, Vasconcelos JF. USBL/INS tightly-coupled integration technique for underwater vehicles. In: 2006 9th International Conference on Information Fusion. IEEE; 2006. p. 1–8.
- [64] Open Source Sensor Fusion;. Available from: <https://github.com/memsindustrygroup/Open-Source-Sensor-Fusion/tree/master/docs>.
- [65] Georgy JFW. Advanced nonlinear techniques for low cost land vehicle navigation. Queen’s University, Kingston, Ontario, Canada; 2010. Ph.D. thesis.
- [66] Sonardyne International limited. Ranger USBL Acoustic Positioning System for DP Reference and Survey Positioning Ranger USBL DP Reference and Survey Positioning. Ranger USBL datasheet. 2009;p. 1–6.
- [67] Morgado M, Oliveira P, Silvestre C, Vasconcelos JF. Embedded Vehicle Dynamics Aiding for USBL/INS Underwater Navigation System. IEEE Transactions on Control Systems Technology. 2014;22(1):322–330.

- [68] Zhang T, Shi H, Chen L, Li Y, Tong J. AUV Positioning Method Based on Tightly Coupled SIN-S/LBL for Underwater Acoustic Multipath Propagation. *Sensors*. 2016 Mar;16(3):357. Available from: <http://dx.doi.org/10.3390/s16030357>.
- [69] Reis J, Morgado M, Batista P, Oliveira P, Silvestre C. Design and Experimental Validation of a USBL Underwater Acoustic Positioning System. *Sensors*. 2016 Sep;16(9):1491. Available from: <http://dx.doi.org/10.3390/s16091491>.
- [70] Kinsey JC, Eustice RM, Whitcomb LL. A survey of underwater vehicle navigation: Recent advances and new challenges. In: *IFAC Conference of Manoeuvring and Control of Marine Craft*. vol. 88. Lisbon; 2006. p. 1–12.
- [71] Borenstein J, Everett HR, Feng L, Wehe D. Mobile robot positioning: Sensors and techniques. *Journal of robotic systems*. 1997;14(4):231–249.
- [72] Tan HP, Diamant R, Seah WKG, Waldmeyer M. A survey of techniques and challenges in underwater localization. *Ocean Engineering*. 2011;38(14):1663 – 1676. Available from: <http://www.sciencedirect.com/science/article/pii/S0029801811001624>.
- [73] Arkhipov M. Utilizing Johnson Solids for Designing Multielement USBL Systems. *IEEE Journal of Oceanic Engineering*. 2016;41(4):783–793.
- [74] Brian Bingham, Seering W. Hypothesis grids: improving long baseline navigation for autonomous underwater vehicles. *IEEE Journal of Oceanic Engineering*. 2006;31(1):209–218.
- [75] Paull L, Saeedi S, Seto M, Li H. AUV Navigation and Localization: A Review. *IEEE Journal of Oceanic Engineering*. 2014;39(1):131–149.
- [76] Fengzhong Q, Shiyuan W, Zhihui W, Zubin L. A survey of ranging algorithms and localization schemes in underwater acoustic sensor network. *China Communications*. 2016;13(3):66–81.
- [77] Han G, Jiang J, Shu L, Xu Y, Wang F. Localization algorithms of underwater wireless sensor networks: A survey. *Sensors*. 2012;12(2):2026–2061.
- [78] Islam T, Park SH. A Comprehensive Survey of the Recently Proposed Localization Protocols for Underwater Sensor Networks. *IEEE Access*. 2020;8:179224–179243.
- [79] Zhou Z, Cui JH, Zhou S. Efficient localization for large-scale underwater sensor networks. *Ad Hoc Networks*. 2010;8(3):267 – 279. Available from: <http://www.sciencedirect.com/science/article/pii/S157087050900081X>.
- [80] Zhou Z, Peng Z, Cui J, Shi Z, Bagtzoglou A. Scalable Localization with Mobility Prediction for Underwater Sensor Networks. *IEEE Transactions on Mobile Computing*. 2011 March;10(3):335–348.
- [81] Albowicz J, Chen A, Zhang L. Recursive position estimation in sensor networks. In: *Proceedings Ninth International Conference on Network Protocols. ICNP 2001*. IEEE; 2001. p. 35–41.
- [82] Niculescu D, Nath B. Ad hoc positioning system (APS). In: *GLOBECOM'01. IEEE Global Telecommunications Conference (Cat. No.01CH37270)*. vol. 5; 2001. p. 2926–2931 vol.5.
- [83] Novikov A, Bagtzoglou AC. Hydrodynamic model of the lower hudson river estuarine system and its application for water quality management. *Water resources management*. 2006;20(2):257–276.
- [84] Bagtzoglou AC, Novikov A. Chaotic Behavior and Pollution Dispersion Characteristics in Engineered Tidal Embayments: A Numerical Investigation 1. *JAWRA Journal of the American Water Resources Association*. 2007;43(1):207–219.

- [85] Strobach P. Linear prediction theory: a mathematical basis for adaptive systems. vol. 21. Springer Science & Business Media; 2012.
- [86] Watfa MK, Nsouli T, Al-Ayache M, Ayyash O. Reactive Localization in Underwater Wireless Sensor Networks. In: 2010 Second International Conference on Computer and Network Technology; 2010. p. 244–248.
- [87] Cheng X, Shu H, Liang Q, Du DHC. Silent positioning in underwater acoustic sensor networks. *IEEE Transactions on vehicular technology*. 2008;57(3):1756–1766.
- [88] Tan HP, Eu ZA, Seah WK. An enhanced underwater positioning system to support deepwater installations. In: *OCEANS 2009. IEEE*; 2009. p. 1–8.
- [89] Mackenzie KV. Nine-term equation for sound speed in the oceans. *The Journal of the Acoustical Society of America*. 1981;70(3):807–812. Available from: <https://doi.org/10.1121/1.386920>.
- [90] Ameer P, Jacob L. Localization using ray tracing for underwater acoustic sensor networks. *IEEE Communications Letters*. 2010;14(10):930–932.
- [91] Erol M, Vieira LF, Gerla M. Localization with Dive’N’Rise (DNR) beacons for underwater acoustic sensor networks. In: *Proceedings of the second workshop on Underwater networks*; 2007. p. 97–100.
- [92] Caruso A, Paparella F, Vieira LFM, Erol M, Gerla M. The meandering current mobility model and its impact on underwater mobile sensor networks. In: *IEEE INFOCOM 2008-The 27th Conference on Computer Communications*. IEEE; 2008. p. 221–225.
- [93] Abougamila S, Elmorsy M, Elmallah ES. A Graph Theoretic Approach to Localization under Uncertainty. In: *2018 IEEE International Conference on Communications (ICC)*; 2018. p. 1–7.
- [94] Teymorian AY, Cheng W, Ma L, Cheng X, Lu X, Lu Z. 3D Underwater Sensor Network Localization. *IEEE Transactions on Mobile Computing*. 2009 Dec;8(12):1610–1621.
- [95] Thomas F, Ros L. Revisiting trilateration for robot localization. *IEEE Transactions on Robotics*. 2005;21(1):93–101.
- [96] McLaren-Young-Sommerville D. *An Introduction to the Geometry of n Dimensions*. 1958;.
- [97] Coope ID. Reliable computation of the points of intersection of n spheres in \mathbb{R}^n . *ANZIAM Journal*. 2000;42:461–477.
- [98] FOY WH. Position-Location Solutions by Taylor-Series Estimation. *IEEE Transactions on Aerospace and Electronic Systems*. 1976;AES-12(2):187–194.
- [99] Navidi W, Murphy WS, Hereman W. Statistical methods in surveying by trilateration. *Computational Statistics Data Analysis*. 1998;27(2):209 – 227. Available from: <http://www.sciencedirect.com/science/article/pii/S0167947397000534>.
- [100] Sand S, Dammann A, Mensing C. In: *Position Estimation*. Wiley; 2013. p. 69–99. Available from: <https://ieeexplore.ieee.org/document/8042588>.
- [101] Kennedy J, Eberhart R. Particle swarm optimization. In: *Proceedings of ICNN’95 - International Conference on Neural Networks*. vol. 4; 1995. p. 1942–1948 vol.4.
- [102] Pedersen MEH. Good parameters for particle swarm optimization. Hvass Lab, Copenhagen, Denmark, Tech Rep HL1001. 2010;p. 1551–3203.
- [103] Zhou Y. An efficient least-squares trilateration algorithm for mobile robot localization. In: *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*; 2009. p. 3474–3479.

- [104] Golub GH, Van Loan CF. Matrix computations. vol. 4. The Johns Hopkins University Press; 2012.
- [105] Soares C, Xavier J, Gomes J. Simple and fast convex relaxation method for cooperative localization in sensor networks using range measurements. *IEEE Transactions on Signal Processing*. 2015;63(17):4532–4543.
- [106] Sahin E, Winfield AF. Special issue on swarm robotics. *Swarm Intelligence*. 2008;2(2-4):69–72.
- [107] Roumeliotis SI, Bekey GA. Distributed multirobot localization. *IEEE Transactions on Robotics and Automation*. 2002 Oct;18(5):781–795.
- [108] Mu H, Bailey T, Thompson P, Durrant-Whyte H. Decentralised Solutions to the Cooperative Multi-Platform Navigation Problem. *IEEE Transactions on Aerospace and Electronic Systems*. 2011 April;47(2):1433–1449.
- [109] Caiti A, Munafò A, Viviani R. Adaptive on-line planning of environmental sampling missions with a team of cooperating autonomous underwater vehicles. *International Journal of Control*. 2007;80(7):1151–1168. Available from: <https://doi.org/10.1080/00207170701230973>.
- [110] Leonard NE, Paley DA, Davis RE, Fratantoni DM, Lekien F, Zhang F. Coordinated control of an underwater glider fleet in an adaptive ocean sampling field experiment in Monterey Bay. *Journal of Field Robotics*. 2010;27(6):718–740. Available from: <https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.20366>.
- [111] Alvarez A, Mourre B. Oceanographic Field Estimates from Remote Sensing and Glider Fleets. *Journal of Atmospheric and Oceanic Technology*. 2012 11;29(11):1657–1662. Available from: <https://doi.org/10.1175/JTECH-D-12-00015.1>.
- [112] Ferri G, Cococcioni M, Alvarez A. Mission planning and decision support for underwater glider networks: A sampling on-demand approach. *Sensors*. 2016;16(1):28.
- [113] Caiti A, Calabrò V, Dini G, Lo Duca A, Munafò A. Secure Cooperation of Autonomous Mobile Sensors Using an Underwater Acoustic Network. *Sensors*. 2012 Feb;12(2):1967–1989. Available from: <http://dx.doi.org/10.3390/s120201967>.
- [114] Webster SE, Walls JM, Whitcomb LL, Eustice RM. Decentralized extended information filter for single-beacon cooperative acoustic navigation: Theory and experiments. *IEEE Transactions on Robotics*. 2013;29(4):957–974.
- [115] Caiti A, Calabrò V, Fabbri T, Fenucci D, Munafò A. Underwater communication and distributed localization of AUV teams. In: 2013 MTS/IEEE OCEANS - Bergen; 2013. p. 1–8.
- [116] Bahr A, Leonard JJ, Fallon MF. Cooperative localization for autonomous underwater vehicles. *The International Journal of Robotics Research*. 2009;28(6):714–728.
- [117] Fenucci D, Munafò A, Phillips AB, Neasham J, Gold N, Sitbon J, et al. Development of smart networks for navigation in dynamic underwater environments. In: 2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV); 2018. p. 1–6.
- [118] Phillips AB, Gold N, Linton N, Harris CA, Richards E, Templeton R, et al. Agile design of low-cost autonomous underwater vehicles. In: OCEANS 2017 - Aberdeen; 2017. p. 1–7.
- [119] Munafò A, Śliwka J, Petroccia R. Localisation Using Undersea Wireless Networks. In: 2018 OCEANS - MTS/IEEE Kobe Techno-Oceans (OTO); 2018. p. 1–7.

- [120] Che X, Wells I, Dickers G, Kear P, Gong X. Re-evaluation of RF electromagnetic communication in underwater sensor networks. *IEEE Communications Magazine*. 2010;48(12):143–151.
- [121] Shaneyfelt T, Joordens MA, Nagothu K, Jamshidi M. RF communication between surface and underwater robotic swarms. In: 2008 World Automation Congress; 2008. p. 1–6.
- [122] Gabriel C, Khalighi M, Bourennane S, Leon P, Rigaud V. Monte-Carlo-based channel characterization for underwater optical communication systems. *IEEE/OSA Journal of Optical Communications and Networking*. 2013;5(1):1–12.
- [123] Pontbriand C, Farr N, Ware J, Preisig J, Popenoe H. Diffuse high-bandwidth optical communications. In: *OCEANS 2008*; 2008. p. 1–4.
- [124] Melodia T, Kulhandjian H, Kuo LC, Demirors E. 23. In: *Advances in Underwater Acoustic Networking*. John Wiley Sons, Ltd; 2013. p. 804–852. Available from: <https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118511305.ch23>.
- [125] Freitag LE, Merriam JS, Frye DE, Catipovic JA. A Long-term Deep-water Acoustic Telemetry Experiment. In: *OCEANS 91 Proceedings*. vol. 1; 1991. p. 254–260.
- [126] Jones JC, Di Meglio A, Wang LS, Coates RFW, Tedeschi A, Stoner RJ. The design and testing of a DSP, half-duplex, vertical, DPSK communication link. In: *Oceans '97. MTS/IEEE Conference Proceedings*. vol. 1; 1997. p. 259–266 vol.1.
- [127] Edelmann GF, Song HC, Kim S, Hodgkiss WS, Kuperman WA, Akal T. Underwater acoustic communications using time reversal. *IEEE Journal of Oceanic Engineering*. 2005;30(4):852–864.
- [128] Chirdchoo N, Soh W, Chua KC. Aloha-Based MAC Protocols with Collision Avoidance for Underwater Acoustic Networks. In: *IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications*; 2007. p. 2271–2275.
- [129] Ergen SC, Varaiya P. TDMA scheduling algorithms for wireless sensor networks. *Wireless Networks*. 2010;16(4):985–997.
- [130] Morozs N, Mitchell P, Zakharov YV. TDA-MAC: TDMA without clock synchronization in underwater acoustic networks. *IEEE Access*. 2017;6:1091–1108.
- [131] Colvin A. CSMA with collision avoidance. *Computer Communications*. 1983;6(5):227–235.
- [132] Pompili D, Melodia T, Akyildiz IF. A CDMA-based medium access control for underwater acoustic sensor networks. *IEEE Transactions on Wireless Communications*. 2009;8(4):1899–1909.
- [133] Dixit S, Tripathi P, Shukla M. SC-FDMA-IDMA scheme for underwater Acoustic communications. In: *2015 Communication, Control and Intelligent Systems (CCIS)*; 2015. p. 204–207.
- [134] Xilin Cheng, Fengzhong Qu, Liuqing Yang. Single carrier FDMA over underwater acoustic channels. In: *2011 6th International ICST Conference on Communications and Networking in China (CHINACOM)*; 2011. p. 1052–1057.
- [135] Petritoli E, Cagnetti M, Leccese F. Simulation of Autonomous Underwater Vehicles (AUVs) Swarm Diffusion. *Sensors*. 2020 Sep;20(17):4950. Available from: <http://dx.doi.org/10.3390/s20174950>.
- [136] Webots. Cyberbotics, editor. <http://www.cyberbotics.com>; Commercial Mobile Robot Simulation Software. Available from: <http://www.cyberbotics.com>.

- [137] EASY-ROB. 3D Robot Simulation Tool;. EASY-ROB. Available from: <https://easy-rob.com/en/start-2/>.
- [138] Gazebo. GAZEBO;. Robot simulation made easy. Available from: <http://gazebosim.org/>.
- [139] Coppelia. Coppelia Robotics;. Create Compose Simulate Any Robot. Available from: <https://www.coppeliarobotics.com/>.
- [140] Smith R. Open Dynamics Engine;. High performance library for simulating rigid body dynamics. Available from: <https://www.ode.org/>.
- [141] Staranowicz A, Mariottini GL. A survey and comparison of commercial and open-source robotic simulator software. In: Proceedings of the 4th International Conference on PErvasive Technologies Related to Assistive Environments. ACM; 2011. p. 56.
- [142] UnetStack. Acoustic Research Laboratory NUoS, Ltd SP, editors. <https://unetstack.net/>;. The Underwater Networks Project. Available from: <https://unetstack.net>.
- [143] Chitre M, Bhatnagar R, Soh WS. UnetStack: An agent-based software stack and simulator for underwater networks. In: 2014 Oceans-St. John's. IEEE; 2014. p. 1–10.
- [144] Pompili D, Akyildiz IF. A Multimedia Cross-Layer Protocol for Underwater Acoustic Sensor Networks. IEEE Transactions on Wireless Communications. 2010;9(9):2924–2933.
- [145] Issariyakul T, Hossain E. In: Introduction to Network Simulator 2 (NS2). Boston, MA: Springer US; 2012. p. 21–40. Available from: https://doi.org/10.1007/978-1-4614-1406-3_2.
- [146] Baldo N, Miozzo M, Guerra F, Rossi M, Zorzi M. Miracle: the multi-interface cross-layer extension of ns2. EURASIP Journal on Wireless Communications and Networking. 2010;2010:26.
- [147] Masiero R, Azad S, Favaro F, Petrani M, Toso G, Guerra F, et al. DESERT Underwater: an NS-Miracle-based framework to DEsign, Simulate, Emulate and Realize Test-beds for Underwater network protocols. In: 2012 Oceans-Yeosu. IEEE; 2012. p. 1–10.
- [148] Petrioli C, Petroccia R, Spaccini D. SUNSET version 2.0: Enhanced framework for simulation, emulation and real-life testing of underwater wireless sensor networks. In: Proceedings of the Eighth ACM International Conference on Underwater Networks and Systems. ACM; 2013. p. 43.
- [149] Potter J, Alves J, Green D, Zappa G, Nissen I, McCoy K. The JANUS underwater communications standard. In: 2014 Underwater Communications and Networking (UComms); 2014. p. 1–4.
- [150] Urick RJ. Principles of underwater sound 3rd edition. Peninsula Publising Los Atlos, California. 1983;.
- [151] Neasham J, editor. USMART - Newcastle University;. Smart dust for large scale underwater wireless sensing. Available from: <https://research.ncl.ac.uk/usmart/>.
- [152] Neasham JA, Goodfellow G, Sharpouse R. Development of the “Seatrac” miniature acoustic modem and USBL positioning units for subsea robotics and diver applications. In: OCEANS 2015 - Genova; 2015. p. 1–8.
- [153] Sherlock B, Neasham JA, Tsimenidis CC. Spread-Spectrum Techniques for Bio-Friendly Underwater Acoustic Communications. IEEE Access. 2018;6:4506–4520.
- [154] Smith CM, Leonard JJ, Bennett AA, Shaw C. Feature-based concurrent mapping and localization for AUVs. In: Oceans' 97. MTS/IEEE Conference Proceedings. vol. 2. IEEE; 1997. p. 896–901.

- [155] Fang BT. Trilateration and extension to global positioning system navigation. *Journal of Guidance, Control, and Dynamics*. 1986;9(6):715–717.
- [156] Dubois DJ. *Fuzzy sets and systems: theory and applications*. vol. 144. Academic press; 1980.
- [157] Eichfeld H, Kunemund T, Menke M. A 12b general-purpose fuzzy logic controller chip. *IEEE Transactions on Fuzzy Systems*. 1996;4(4):460–475.
- [158] Arpaci-Dusseau RH, Arpaci-Dusseau AC. Scheduling Introduction. In: *Operating systems: Three easy pieces*. Arpaci-Dusseau Books LLC; 2014. p. 63–74.
- [159] Zadeh LA. The concept of a linguistic variable and its application to approximate reasoning. In: *Learning systems and intelligent robots*. Springer; 1974. p. 1–10.
- [160] Zadeh LA. In: Yager RR, Zadeh LA, editors. *Knowledge Representation in Fuzzy Logic*. Boston, MA: Springer US; 1992. p. 1–25. Available from: https://doi.org/10.1007/978-1-4615-3640-6_1.
- [161] Mamdani EH. Application of fuzzy logic to approximate reasoning using linguistic synthesis. *IEEE transactions on computers*. 1977;(12):1182–1191.
- [162] Chame HF, dos Santos MM, da Costa Botelho SS. Neural network for black-box fusion of underwater robot localization under unmodeled noise. *Robotics and Autonomous Systems*. 2018;110:57 – 72. Available from: <http://www.sciencedirect.com/science/article/pii/S0921889018302926>.
- [163] Chame HF, dos Santos MM, da Costa Botelho SS. Reliable fusion of black-box estimates of underwater localization. In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*; 2018. p. 1900–1905.
- [164] Kussat NH, Chadwell CD, Zimmerman R. Absolute positioning of an autonomous underwater vehicle using GPS and acoustic measurements. *IEEE Journal of Oceanic Engineering*. 2005 Jan;30(1):153–164.
- [165] Jiang C, Chen S, Chen Y, Bo Y. Research on a chip scale atomic clock aided vector tracking loop. *IET Radar, Sonar Navigation*. 2019;13(7):1101–1106.
- [166] Kepper JH, Claus BC, Kinsey JC. A Navigation Solution Using a MEMS IMU, Model-Based Dead-Reckoning, and One-Way-Travel-Time Acoustic Range Measurements for Autonomous Underwater Vehicles. *IEEE Journal of Oceanic Engineering*. 2019;44(3):664–682.
- [167] Mourya R, Saafin W, Dragone M, Petillot Y. Ocean monitoring framework based on compressive sensing using acoustic sensor networks. In: *OCEANS 2018 MTS/IEEE Charleston*. IEEE; 2018. p. 1–10.
- [168] Stuckmann P. *The GSM evolution: mobile packet data services*. John Wiley & Sons; 2002.
- [169] Rahn CD, Wang CY. *Battery systems engineering*. John Wiley & Sons; 2013.
- [170] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*. 2002;6(2):182–197.
- [171] Yu S, Xu Y, Jiang P, Wu F, Xu H. Node Self-Deployment Algorithm Based on Pigeon Swarm Optimization for Underwater Wireless Sensor Networks. *Sensors*. 2017 Mar;17(4):674. Available from: <http://dx.doi.org/10.3390/s17040674>.
- [172] Goldberg DE. *Genetic algorithms in search, optimization, and machine learning*, Addison-Wesley, Reading, MA. 1989;3(1).

- [173] Konak A, Coit DW, Smith AE. Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering & System Safety*. 2006;91(9):992–1007.
- [174] Deb K. *Multi-objective optimization using evolutionary algorithms*. vol. 16. John Wiley & Sons; 2001.
- [175] Bendsoe MP, Sigmund O. *Topology optimization: theory, methods, and applications*. Springer Science & Business Media; 2013.
- [176] Bhuvanewari PTV, Karthikeyan S, Jeeva B, Prasath MA. An Efficient Mobility Based Localization in Underwater Sensor Networks. In: *2012 Fourth International Conference on Computational Intelligence and Communication Networks*; 2012. p. 90–94.
- [177] Yang Z, Liu Y. Quality of Trilateration: Confidence-Based Iterative Localization. *IEEE Transactions on Parallel and Distributed Systems*. 2010 May;21(5):631–640.
- [178] Thrun S, Burgard W, Fox D. *PROBABILISTIC ROBOTICS*. Cambridge, Mass.: MIT Press; 2006.
- [179] Codling EA, Plank MJ, Benhamou S. Random walk models in biology. *Journal of the Royal society interface*. 2008;5(25):813–834.
- [180] Leonard JJ, Bahr A. Autonomous underwater vehicle navigation. In: *Springer Handbook of Ocean Engineering*. Springer; 2016. p. 341–358.
- [181] Paroscientific I. Submersible Depth Sensors Series 8000. Paroscientific, Inc.; 2005. Available from: https://www.esonetyellowpages.com/datasheets/datasheet-series_8cb_1254684176.pdf.
- [182] Sozer EM, Stojanovic M, Proakis JG. Underwater acoustic networks. *IEEE Journal of Oceanic Engineering*. 2000 Jan;25(1):72–83.
- [183] Lanbo L, Shengli Z, Jun-Hong C. Prospects and problems of wireless communication for underwater sensor networks. *Wireless Communications and Mobile Computing*. 2008;8(8):977–994.
- [184] Shnidman DA. Efficient computation of the circular error probability (CEP) integral. *IEEE Transactions on Automatic Control*. 1995;40(8):1472–1474.
- [185] Shoemake K. Animating Rotation with Quaternion Curves. In: *Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '85*. New York, NY, USA: Association for Computing Machinery; 1985. p. 245–254. Available from: <https://doi.org/10.1145/325334.325242>.
- [186] Bahr A, Leonard JJ. Minimizing trilateration errors in the presence of uncertain landmark positions. In: *3rd European Conference on Mobile Robots (ECMR). CONF*; 2007. p. 48–53.
- [187] Cao H, Lacroix S, Ingrand F, Alami R. Complex tasks allocation for multi robot teams under communication constraints. *Control architecture for robots*. 2010;p. 1–12.
- [188] Botelho SC, Alami R. M+: a scheme for multi-robot cooperation through negotiated task allocation and achievement. In: *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*. vol. 2; 1999. p. 1234–1239 vol.2.
- [189] Gerkey BP, Mataric MJ. Sold!: auction methods for multirobot coordination. *IEEE Transactions on Robotics and Automation*. 2002;18(5):758–768.
- [190] Willners JS, Petillot YR, Patron P, Gonzalez-Adell D. Kinodynamic Path Planning for Following and Tracking Vehicles. In: *OCEANS 2018 MTS/IEEE Charleston*; 2018. p. 1–6.

- [191] Frizzell-Makowski L, Shelsby R, Mann J, Scheidt D. An autonomous energy harvesting station-keeping vehicle for persistent ocean surveillance. In: OCEANS'11 MTS/IEEE KONA. IEEE; 2011. p. 1-4.
- [192] Cufi X, Garcia R, Ridaou P. An approach to vision-based station keeping for an unmanned underwater vehicle. In: IEEE/RSJ International Conference on Intelligent Robots and Systems. vol. 1. IEEE; 2002. p. 799-804.

Appendix A

A Fuzzy Localisation Framework for Underwater Robotic Swarms

A.1 Fuzzy rules - MATLAB simulation

- IF \mathcal{D} is Shallow AND \mathcal{B} is High AND \mathcal{U} is Available AND \mathcal{G} is High THEN \mathcal{Y} is L_3 .
- IF \mathcal{D} is Shallow AND \mathcal{B} is High AND \mathcal{U} is Available AND \mathcal{G} is Adequate THEN \mathcal{Y} is L_2 .
- IF \mathcal{D} is Shallow AND \mathcal{B} is High AND \mathcal{U} is Available AND \mathcal{G} is Low THEN \mathcal{Y} is L_1 .
- IF \mathcal{D} is Shallow AND \mathcal{B} is Mid AND \mathcal{U} is Available AND \mathcal{G} is High THEN \mathcal{Y} is L_3 .
- IF \mathcal{D} is Shallow AND \mathcal{B} is Mid AND \mathcal{U} is Available AND \mathcal{G} is Adequate THEN \mathcal{Y} is L_2 .
- IF \mathcal{D} is Shallow AND \mathcal{B} is Mid AND \mathcal{U} is Available AND \mathcal{G} is Low THEN \mathcal{Y} is L_1 .
- IF \mathcal{D} is Shallow AND \mathcal{B} is Low AND \mathcal{U} is Available AND \mathcal{G} is High THEN \mathcal{Y} is L_3 .
- IF \mathcal{D} is Shallow AND \mathcal{B} is Low AND \mathcal{U} is Available AND \mathcal{G} is Adequate THEN \mathcal{Y}

is L_2 .

- IF \mathcal{D} is Shallow AND \mathcal{B} is Low AND \mathcal{U} is Available AND \mathcal{G} is Low THEN \mathcal{Y} is L_1 .
- IF \mathcal{D} is Shallow AND \mathcal{U} is Not Available AND \mathcal{G} is Low THEN \mathcal{Y} is L_2 .
- IF \mathcal{D} is Shallow AND \mathcal{U} is Not Available AND \mathcal{G} is Adequate THEN \mathcal{Y} is L_2 .
- IF \mathcal{D} is Shallow AND \mathcal{U} is Not Available AND \mathcal{G} is High THEN \mathcal{Y} is L_3 .
- IF \mathcal{D} is Mid-water AND \mathcal{U} is Available AND \mathcal{G} is Low THEN \mathcal{Y} is L_1 .
- IF \mathcal{D} is Mid-water AND \mathcal{U} is Not Available AND \mathcal{G} is Low THEN \mathcal{Y} is L_2 .
- IF \mathcal{D} is Mid-water AND \mathcal{G} is High THEN \mathcal{Y} is L_3 .
- IF \mathcal{D} is Mid-water AND \mathcal{G} is Adequate THEN \mathcal{Y} is L_2 .
- IF \mathcal{D} is Deep AND \mathcal{B} is High AND \mathcal{G} is High THEN \mathcal{Y} is L_3 .
- IF \mathcal{D} is Deep AND \mathcal{B} is Mid AND \mathcal{G} is High THEN \mathcal{Y} is L_3 .
- IF \mathcal{D} is Deep AND \mathcal{B} is Low AND \mathcal{G} is High THEN \mathcal{Y} is L_4 .
- IF \mathcal{D} is Deep AND \mathcal{G} is Adequate THEN \mathcal{Y} is L_4 .
- IF \mathcal{D} is Deep AND \mathcal{G} is Low THEN \mathcal{Y} is L_4 .
- IF \mathcal{D} is Mid-water AND \mathcal{U} is Available AND \mathcal{B} is High AND \mathcal{G} is Low THEN \mathcal{Y} is L_1 .
- IF \mathcal{D} is Deep AND \mathcal{G} is Mid AND \mathcal{G} is Low THEN \mathcal{Y} is L_4 .

A.2 Fuzzy rules - Webots simulation

- IF \mathcal{D} is Shallow AND \mathcal{R} is Short THEN \mathcal{Y} is L_1 .
- IF \mathcal{U} is Not Available AND \mathcal{G} is Not Enough THEN \mathcal{Y} is L_1 .
- IF \mathcal{B} is Low AND \mathcal{U} is Available AND \mathcal{G} is Not Enough AND \mathcal{R} is Long THEN \mathcal{Y} is L_2 .
- IF \mathcal{B} is Low AND \mathcal{U} is Not Available AND \mathcal{G} is Enough AND \mathcal{R} is Long THEN \mathcal{Y} is L_3 .

- IF \mathcal{U} is Available AND \mathcal{R} is Long THEN \mathcal{Y} is L_2 .
- IF \mathcal{B} is High AND \mathcal{U} is Available AND \mathcal{G} is Not Enough AND \mathcal{R} is Mid THEN \mathcal{Y} is L_2 .
- IF \mathcal{B} is High AND \mathcal{U} is Not Available AND \mathcal{G} is Enough AND \mathcal{R} is Long THEN \mathcal{Y} is L_3 .
- IF \mathcal{D} is Shallow AND \mathcal{B} is High AND \mathcal{U} is Not Available AND \mathcal{G} is Enough AND \mathcal{R} is Mid THEN \mathcal{Y} is L_3 .
- IF \mathcal{D} is Shallow AND \mathcal{B} is High AND \mathcal{U} is Available AND \mathcal{G} is Enough AND \mathcal{R} is Mid THEN \mathcal{Y} is L_2 .
- IF \mathcal{U} is Available AND \mathcal{R} is Mid THEN \mathcal{Y} is L_2 .
- IF \mathcal{B} is Low AND \mathcal{U} is Available AND \mathcal{G} is Enough AND \mathcal{R} is Long THEN \mathcal{Y} is L_3 .
- IF \mathcal{D} is Deep AND \mathcal{B} is High AND \mathcal{U} is Available AND \mathcal{R} is Mid THEN \mathcal{Y} is L_2 .
- IF \mathcal{D} is Deep AND \mathcal{B} is High AND \mathcal{U} is Available AND \mathcal{R} is Long THEN \mathcal{Y} is L_2 .
- IF \mathcal{D} is Shallow AND \mathcal{B} is High AND \mathcal{U} is Available AND \mathcal{R} is Long THEN \mathcal{Y} is L_2 .
- IF \mathcal{B} is High AND \mathcal{U} is Not Available AND \mathcal{G} is Enough AND \mathcal{R} is Long THEN \mathcal{Y} is L_3 .
- IF \mathcal{D} is Deep AND \mathcal{B} is High AND \mathcal{U} is Not Available AND \mathcal{G} is Enough AND \mathcal{R} is Mid THEN \mathcal{Y} is L_1 .
- IF \mathcal{B} is High AND \mathcal{U} is Available AND \mathcal{R} is Long THEN \mathcal{Y} is L_2 .
- IF \mathcal{B} is High AND \mathcal{U} is Available AND \mathcal{R} is Mid THEN \mathcal{Y} is L_2 .
- IF \mathcal{B} is High AND \mathcal{U} is Not Available AND \mathcal{G} is Enough AND \mathcal{R} is Long THEN \mathcal{Y} is L_3 .
- IF \mathcal{U} is Not Available AND \mathcal{G} is Enough AND \mathcal{R} is Mid THEN \mathcal{Y} is L_3 .
- IF \mathcal{D} is Deep AND \mathcal{U} is Available THEN \mathcal{Y} is L_2 .

- IF \mathcal{D} is Deep AND \mathcal{U} is Not Available AND \mathcal{G} is Enough AND \mathcal{R} is Short THEN \mathcal{Y} is L_1 .

A.3 One-tail two sample t -test

The null hypothesis (H0) and the alternate hypothesis (H1) of the one-tail two sample t -test performed in subsection 5.4.3 are listed as follows:

H0: Mean_EKF_Error = Mean_Fuzzy_Error

H1: Mean_EKF_Error > Mean_Fuzzy_Error

Swarm size	H0 Rejected	P-value	DoF	t -statistics	Critical value
50	No	1.0000	75.45	-4.35	1.6653
100	Yes	0.0252	194.48	1.96	1.6527
150	Yes	0.0324	255.45	1.85	1.6508

Table A.1: One-tail two sample t -test

where the critical value is the inverse cumulative density function (CDF) of t distribution at 0.05 significant level. Rejection of the null hypothesis H0 indicates that the mean localisation error of EKF-based method is greater than that of the proposed fuzzy-based method at 0.05 significance level.

Appendix B

Confidence based Localisation for Cooperative Underwater Robotic Swarms using the Extended Kalman Filter

B.1 One-tail two sample t -test

The null hypothesis (H0) and the alternate hypothesis (H1) of the one-tail two sample t -test performed in subsection 7.3.2 are listed as follows:

H0: Mean_EKF_Error = Mean_Confidence_Error

H1: Mean_EKF_Error > Mean_Confidence_Error

Swarm size	H0 Rejected	P-value	DoF	t -statistics	Critical value
50	Yes	9.51×10^{-11}	83.74	7.25	1.6633
100	Yes	7.42×10^{-18}	105.30	10.26	1.6595
150	Yes	8.03×10^{-16}	172.14	8.77	1.6538

Table B.1: One-tail two sample t -test

where the critical value is the inverse cumulative density function (CDF) of t distribution at 0.05 significant level. Rejection of the null hypothesis H0 indicates that the mean

localisation error of [EKF](#)-based method is greater than that of the proposed confidence-based localisation method at 0.05 significance level.