

## Underwater Acoustic Localization and Packet Scheduling

Mashhadi Ramezani, Hamid

**DOI**

[10.4233/uuid:c7e61f5d-2955-4fde-8b91-fdd22bf10fce](https://doi.org/10.4233/uuid:c7e61f5d-2955-4fde-8b91-fdd22bf10fce)

**Publication date**

2016

**Document Version**

Final published version

**Citation (APA)**

Mashhadi Ramezani, H. (2016). *Underwater Acoustic Localization and Packet Scheduling*. [Dissertation (TU Delft), Delft University of Technology]. <https://doi.org/10.4233/uuid:c7e61f5d-2955-4fde-8b91-fdd22bf10fce>

**Important note**

To cite this publication, please use the final published version (if applicable).  
Please check the document version above.

**Copyright**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights.  
We will remove access to the work immediately and investigate your claim.

# Underwater Acoustic Localization and Packet Scheduling

**Ph.D. Thesis**

Hamid Ramezani



# Underwater Acoustic Localization and Packet Scheduling

## **Proefschrift**

ter verkrijging van de graad van doctor  
aan de Technische Universiteit Delft,  
op gezag van de Rector Magnificus Prof. K.Ch.A.M. Luyben,  
voorzitter van het College van Promoties,  
in het openbaar te verdedigen op  
woensdag 15 juni 2016 om 15:00 uur

door

Hamid MASHHADI RAMEZANI  
Master of Science in Electrical Engineering  
Iran University of Science and Technology

geboren te Tehran-Iran.

Dit proefschrift is goedgekeurd door de promotor:  
Prof. dr. ir. G.J.T. Leus

Samenstelling promotiecommissie:

Rector Magnificus  
Prof. dr. ir. G.J.T. Leus

voorzitter  
Technische Universiteit Delft (promotor)

Independent members:

Prof. dr. ir. A-J. van der Veen

Technische Universiteit Delft

Prof. dr. D.G. Simons

Technische Universiteit Delft

Prof. dr. ing. P.J.M. Havinga

University of Twente, The Netherlands

Prof. dr. A.G. Marques

Universidad Rey Juan Carlos, Spain

Dr. H.S. Dol

TNO, The Netherlands

Dr. J. Han

Northwestern Polytechnical University, China

Prof. dr. K.G. Langendoen

Technische Universiteit Delft (reserved)

The research presented in this thesis was partly supported by the European Commission FP7-ICT Cognitive Systems, Interaction, and Robotics under the contract #270180 (NOPTILUS).

ISBN # 978-94-6186-657-8

Printed by: Gildeprint Drukkerijen - The Netherlands

Copyright © 2016 by Hamid Mashhadi Ramezani

All rights reserved. No part of the material protected by this copyright notice may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage and retrieval system, without written permission of the author.

Thesis Cover is designed and edited by Hamid Mashhadi Ramezani.

# Summary

In the beginning of the 21th century, the goal of telecommunications was to connect everyone anytime anywhere in the world. Now, with the tremendous developments in electronics and communications, we are thinking to go beyond our prior goal and connect anything anytime anywhere (Internet of things). This dream would not have been shaped without the recent advances in wireless sensor networks, and the exploding use of this technology in our everyday life. Still to fulfill such a purpose, we have to face physical and technological challenges especially when it comes to outer space or underwater. In this thesis, we study the fundamental problem of underwater sensor node localization as an indispensable task for any network.

In the first part of this thesis, the main challenges of underwater acoustic communications are reviewed, and their effects on underwater localization algorithms are discussed. It is shown how these algorithms can be categorized into different groups and how they can be evaluated through several metrics.

The second part of the thesis focuses on the development of accurate localization algorithms in an underwater medium with a variable sound speed profile (SSP). The SSP is modeled by linear and piecewise linear functions of the water depth, and optimal localization algorithms are proposed. It is shown that for large network sizes, the performance of the proposed algorithms is much better than traditional approaches.

The large propagation delay of acoustic waves plays an important role in the required time to perform node localization. In order to minimize the localization time, two packet scheduling schemes are proposed in the third part of this thesis, collision-free and collision-tolerant schemes. In this part, through a detailed analysis and numerical results, we show how a localization algorithm can benefit from optimal medium access control design.



# Contents

|                                                                                        |            |
|----------------------------------------------------------------------------------------|------------|
| <b>Summary</b>                                                                         | <b>iii</b> |
| <b>Table of Contents</b>                                                               | <b>ix</b>  |
| <b>List of Figures</b>                                                                 | <b>xv</b>  |
| <br>                                                                                   |            |
| <b>I Preface</b>                                                                       | <b>1</b>   |
| <br>                                                                                   |            |
| <b>1 Introduction</b>                                                                  | <b>3</b>   |
| 1.1 Underwater acoustic history . . . . .                                              | 3          |
| 1.2 Underwater communications . . . . .                                                | 5          |
| 1.3 Underwater acoustic contemporary research . . . . .                                | 8          |
| 1.4 Underwater acoustic localization . . . . .                                         | 10         |
| 1.4.1 Different categories of localization algorithms . . . . .                        | 14         |
| 1.4.2 How to evaluate a localization algorithm . . . . .                               | 15         |
| 1.4.3 Generic localization problem . . . . .                                           | 17         |
| 1.4.4 Review of important localization algorithms . . . . .                            | 17         |
| 1.5 Thesis outline and contributions . . . . .                                         | 25         |
| 1.5.1 Contributions towards underwater localization algorithms                         | 26         |
| 1.5.2 Contributions towards underwater packet scheduling for<br>localization . . . . . | 29         |

|                                                                                               |           |
|-----------------------------------------------------------------------------------------------|-----------|
| <b>II Underwater Localization in Inhomogeneous Medium</b>                                     | <b>35</b> |
| <b>2 Target Localization and Tracking for an Isogradient Sound Speed Profile</b>              | <b>37</b> |
| 2.1 Introduction . . . . .                                                                    | 38        |
| 2.2 Ray tracing between two points . . . . .                                                  | 41        |
| 2.2.1 Time of flight vs. sensor node locations . . . . .                                      | 43        |
| 2.2.2 Traveled ray length . . . . .                                                           | 45        |
| 2.2.3 Ray depth overshoot . . . . .                                                           | 46        |
| 2.2.4 Range approximation using depth information . . . . .                                   | 46        |
| 2.3 Target localization based on time of flight measurements . . . . .                        | 47        |
| 2.3.1 Static network model . . . . .                                                          | 48        |
| 2.3.2 Proposed positioning algorithm . . . . .                                                | 49        |
| 2.3.3 Cramér-Rao bound . . . . .                                                              | 50        |
| 2.3.4 Localization with available depth measurements . . . . .                                | 51        |
| 2.4 Target tracking based on time of flight measurements . . . . .                            | 52        |
| 2.4.1 Dynamic network model . . . . .                                                         | 52        |
| 2.4.2 Extended Kalman filter . . . . .                                                        | 53        |
| 2.4.3 Posterior Cramér-Rao bound . . . . .                                                    | 54        |
| 2.5 Numerical results . . . . .                                                               | 55        |
| 2.6 Conclusions . . . . .                                                                     | 65        |
| <b>3 Ranging in an Underwater Medium with Multiple Isogradient Sound Speed Profile Layers</b> | <b>69</b> |
| 3.1 Introduction . . . . .                                                                    | 70        |
| 3.2 Network model . . . . .                                                                   | 72        |
| 3.3 ToF <i>Versus</i> node positions . . . . .                                                | 74        |
| 3.3.1 ToF <i>Versus</i> node positions in a single layer . . . . .                            | 75        |
| 3.3.2 ToF <i>Versus</i> node positions for two adjacent layers . . . . .                      | 78        |
| 3.3.3 Pattern definition for multi-layer ray propagation . . . . .                            | 80        |
| 3.3.4 ToF <i>Versus</i> node positions according to a given pattern . . . . .                 | 80        |
| 3.4 Pair-wise underwater ranging . . . . .                                                    | 86        |
| 3.4.1 Proposed algorithm . . . . .                                                            | 86        |
| 3.4.2 Cramér-Rao bound . . . . .                                                              | 87        |
| 3.5 Numerical results . . . . .                                                               | 90        |
| 3.5.1 Ray propagation for shallow water . . . . .                                             | 90        |
| 3.5.2 Ranging for deep water . . . . .                                                        | 93        |

|            |                                                                                                     |            |
|------------|-----------------------------------------------------------------------------------------------------|------------|
| 3.6        | Conclusions . . . . .                                                                               | 95         |
| <b>4</b>   | <b>Cramér Rao Lower Bound for Underwater Range Estimation with Noisy Sound Speed Profile</b>        | <b>97</b>  |
| 4.1        | Introduction . . . . .                                                                              | 98         |
| 4.2        | Ray tracing . . . . .                                                                               | 99         |
| 4.3        | Cramér Rao lower bound . . . . .                                                                    | 101        |
| 4.4        | Numerical results . . . . .                                                                         | 105        |
| 4.5        | Conclusions . . . . .                                                                               | 106        |
| 4.6        | Appendix . . . . .                                                                                  | 106        |
| <b>III</b> | <b>Underwater Localization Packet Scheduling</b>                                                    | <b>111</b> |
| <b>5</b>   | <b>Localization Packet Scheduling for Underwater Acoustic Sensor networks</b>                       | <b>113</b> |
| 5.1        | Introduction . . . . .                                                                              | 114        |
| 5.2        | Network model . . . . .                                                                             | 118        |
| 5.2.1      | Collision-free anchors . . . . .                                                                    | 120        |
| 5.2.2      | L-MAC: Localization packet scheduling . . . . .                                                     | 122        |
| 5.2.3      | B-MAC: Broadcasting packet scheduling . . . . .                                                     | 123        |
| 5.2.4      | Dynamic multi-channel packet scheduling . . . . .                                                   | 123        |
| 5.2.5      | Problem formulation in a TDMA system . . . . .                                                      | 125        |
| 5.2.6      | Practical issues for the problem . . . . .                                                          | 125        |
| 5.3        | Optimal solution . . . . .                                                                          | 128        |
| 5.3.1      | Optimal solution in single-channel scenario . . . . .                                               | 128        |
| 5.3.2      | Optimal solution in a multi-channel scenario . . . . .                                              | 130        |
| 5.4        | Mixed-integer linear programming . . . . .                                                          | 132        |
| 5.5        | Proposed algorithms . . . . .                                                                       | 133        |
| 5.6        | Numerical results . . . . .                                                                         | 136        |
| 5.7        | Conclusion . . . . .                                                                                | 143        |
| <b>6</b>   | <b>Collision Tolerant and Collision Free Packet Scheduling for Underwater Acoustic Localization</b> | <b>145</b> |
| 6.1        | Introduction . . . . .                                                                              | 146        |
| 6.2        | System model . . . . .                                                                              | 149        |
| 6.3        | Packet scheduling . . . . .                                                                         | 151        |
| 6.3.1      | Collision-free packet scheduling . . . . .                                                          | 151        |

|           |                                                |            |
|-----------|------------------------------------------------|------------|
| 6.3.2     | Collision-tolerant packet scheduling . . . . . | 155        |
| 6.4       | Self-localization process . . . . .            | 158        |
| 6.4.1     | Localization algorithm . . . . .               | 158        |
| 6.4.2     | Cramér-Rao bound . . . . .                     | 159        |
| 6.5       | Energy consumption . . . . .                   | 160        |
| 6.6       | Numerical results . . . . .                    | 161        |
| 6.7       | Conclusion . . . . .                           | 168        |
| <b>IV</b> | <b>Postface</b>                                | <b>173</b> |
| <b>7</b>  | <b>Conclusion and Future Work</b>              | <b>175</b> |
| 7.1       | Conclusions . . . . .                          | 175        |
| 7.2       | Suggestions for future works . . . . .         | 178        |
|           | <b>Bibliography</b>                            | <b>192</b> |
|           | <b>Samenvatting</b>                            | <b>193</b> |
|           | <b>Propositions</b>                            | <b>195</b> |
|           | <b>Acknowledgments</b>                         | <b>197</b> |
|           | <b>Curriculum Vitae</b>                        | <b>199</b> |
|           | <b>List of Publications</b>                    | <b>201</b> |
|           | <b>Glossary</b>                                | <b>205</b> |
|           | Acronyms . . . . .                             | 205        |
|           | Notations . . . . .                            | 208        |

# List of Figures

|      |                                                                                                                                                                             |    |
|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| 1.1  | Measured sound-speed profiles during the acoustic transmissions for different times of the day as indicated in the legend [1]. . . . .                                      | 7  |
| 1.2  | Sound speed profile for the Pacific environment [2]. . . . .                                                                                                                | 7  |
| 1.3  | Trilateration using three anchors. . . . .                                                                                                                                  | 11 |
| 1.4  | Underwater localization of UAVs using only two fixed anchors. . .                                                                                                           | 13 |
| 1.5  | The Sonardyne LBL system mounted on the seabed, consisting of four anchors [3]. The system is used for accurately locating an underwater structure. . . . .                 | 19 |
| 1.6  | Combination of LBL and SBL systems for positioning [4]. The SBL transducers are attached to the ship in the picture, while the LBL system is mounted on the seabed. . . . . | 19 |
| 1.7  | The vertical movement of an AUV in a network employing the Dive and Rise localization algorithm. . . . .                                                                    | 20 |
| 1.8  | Different message propagation steps in the Asymmetrical Round Trip based Localization algorithm. . . . .                                                                    | 21 |
| 1.9  | Trajectory of the AUV in a network with a multi-stage AUV-aided localization algorithm. . . . .                                                                             | 23 |
| 1.10 | The movement of the AUV equipped with a directional beacon in an underwater acoustic sensor network. . . . .                                                                | 24 |
| 1.11 | Movement of the sensor nodes in a UASN with collaborative localization. . . . .                                                                                             | 25 |
| 2.1  | Description of a ray between a target node and an anchor node. . .                                                                                                          | 42 |
| 2.2  | Error in range calculation resulting from the assumption of a straight-line propagation with a constant speed. . . . .                                                      | 56 |

|      |                                                                                                                                                                                         |    |
|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| 2.3  | Random target node position around the reference point (here the anchors' center of gravity). . . . .                                                                                   | 57 |
| 2.4  | Localization performance with distance-independent measurement noise. . . . .                                                                                                           | 58 |
| 2.5  | Localization performance with distance-dependent measurement noise. . . . .                                                                                                             | 59 |
| 2.6  | RMSE vs. the distance of the target node from the anchors' center of gravity, considering DIN. . . . .                                                                                  | 60 |
| 2.7  | RMSE vs. the distance of the target node from the anchors' center of gravity, considering DDN. . . . .                                                                                  | 61 |
| 2.8  | RMSE vs. SSP steepness. . . . .                                                                                                                                                         | 62 |
| 2.9  | RMSE vs. the distance of the target node from the anchors' center of gravity, considering DIN, and depth measurement. Note that [A] and [B] refer to [5] and [6], respectively. . . . . | 63 |
| 2.10 | Tracking comparison. . . . .                                                                                                                                                            | 64 |
| 2.11 | Effect of the time measurement error. . . . .                                                                                                                                           | 65 |
| 2.12 | Effect of the measured depth report on the proposed tracking algorithm. . . . .                                                                                                         | 66 |
| 2.13 | Effect of the depth measurement error on the proposed tracking algorithm. . . . .                                                                                                       | 66 |
| 2.14 | Effect of the number of anchors on the proposed tracking algorithm. . . . .                                                                                                             | 67 |
| 3.1  | Projection of pair-wise distances on the horizontal plane crossing the target. . . . .                                                                                                  | 74 |
| 3.2  | Samples of ray trajectories as they travel through different layers. . . . .                                                                                                            | 76 |
| 3.3  | ToF error of the straight-line propagation model in a single layer for different values of range and depth. . . . .                                                                     | 79 |
| 3.4  | Linear dependency of the reflection and crossing points under the assumption of a perfect reflection. . . . .                                                                           | 83 |
| 3.5  | (a) Number of paths <i>versus</i> the location of the two nodes. (b) Range error due to the linear approximation of the ToF. . . . .                                                    | 85 |
| 3.6  | Changing the real ray trajectory into a trajectory which is a monotonic function of the depth. . . . .                                                                                  | 89 |
| 3.7  | Sound speed profile for deep and shallow water. . . . .                                                                                                                                 | 91 |
| 3.8  | Sample of ray propagation between two nodes. . . . .                                                                                                                                    | 92 |
| 3.9  | Different possible rays between two points in the second layer. . . . .                                                                                                                 | 93 |

|      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |     |
|------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| 3.10 | Performance of the proposed algorithm for deep water. Note that [A] refers to [6]. . . . .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | 94  |
| 3.11 | Performance of the proposed algorithm for difference values of noise power. . . . .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | 94  |
| 4.1  | Ray propagation between the source and destination. . . . .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 100 |
| 4.2  | The sound speed profile as a function of depth is presented, along with corresponding coefficients of each basis function. . . . .                                                                                                                                                                                                                                                                                                                                                                                                                                                               | 105 |
| 4.3  | CRB of range estimation, a) effect of noisy ToF measurement, b) effect of noisy depth measurement, c) effect of noisy sound speed sample, d) the overall CRB of range estimation. . . . .                                                                                                                                                                                                                                                                                                                                                                                                        | 107 |
| 5.1  | Structure of a localization packet. . . . .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 115 |
| 5.2  | Example of two collision-risk anchors. . . . .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 120 |
| 5.3  | Snapshot of the packet transmissions of two collision-free anchors at time $t = w_j + \frac{R}{c}$ , and at distance $d_{ij}$ where $R < d_{ij} < 2R$ . The hatched parts show the area where the localization packets reside. Given $w_j$ , anchor $i$ has transmitted its packet at minimum $w_i$ according to (5.1). At time $t = w_j + t_p + \frac{R}{c}$ , the effect of the packet transmitted from $j$ -th anchor vanishes and the sensor node which is located on the border (inside the red box in the figure) starts receiving the transmitted packet from the $i$ -th anchor. . . . . | 121 |
| 5.4  | Snapshot of the packet transmissions of two collision-free anchors located $d_{ij} < R$ meters away from each other at time $t = w_j + \frac{R}{c}$ . Given $w_j$ , anchor $i$ transmits at the minimum waiting time according to (5.3). The transmitted packet from the $i$ -th anchor follows that of the $j$ -th anchor, and does not overlap with that. . . . .                                                                                                                                                                                                                              | 122 |
| 5.5  | Network graph for a TDMA-based scheduling problem. Vertices represent the anchors where each anchor is labeled by its index, and an edge connecting two anchors shows there is a collision risk. The boxed numbers represent the time slot that each anchor can use to transmit its localization packet. . . . .                                                                                                                                                                                                                                                                                 | 129 |
| 5.6  | Optimum solution when a TDMA-based algorithm is used. Note that the anchors 1, 3, 7, and 9 are allocated the same time slot. Anchor 4 cannot transmit in time slot 1 because it is a neighbor of some previously scheduled anchors, namely 1 and 7. . . . .                                                                                                                                                                                                                                                                                                                                      | 129 |

|      |                                                                                                                                                                                                                                                                                                                                                                                                      |     |
|------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| 5.7  | Network graph for the distance-aware scheduling problem. The edge weights with white background color represent the normalized distances between collision-risk anchors, and the edge weights with gray background color show the normalized modified distances, $\min\{d_{ij}, 2R - d_{ij}\}$ , between collision-risk anchors. The boxed numbers display the waiting times of the anchors. . . . . | 131 |
| 5.8  | Optimum solution when the optimal distance-aware scheduling algorithm is used. . . . .                                                                                                                                                                                                                                                                                                               | 131 |
| 5.9  | Average packet transmission time versus number of anchors. . . .                                                                                                                                                                                                                                                                                                                                     | 137 |
| 5.10 | Average packet transmission time versus anchors' maximum transmission range. . . . .                                                                                                                                                                                                                                                                                                                 | 138 |
| 5.11 | Performance of the algorithms versus network scalability. . . . .                                                                                                                                                                                                                                                                                                                                    | 139 |
| 5.12 | Waiting time map for a specific network of $N=400$ anchors nodes, $R = 1.1c$ m, and $d_x = d_y = 20c$ m. The vertices show the anchors' locations and the edges show which anchors have a collision risk. The waiting times are computed based on the L-MAC-BS algorithm.                                                                                                                            | 140 |
| 5.13 | Average packet transmission time versus anchors density. . . . .                                                                                                                                                                                                                                                                                                                                     | 141 |
| 5.14 | Average packet transmission time versus packet length. . . . .                                                                                                                                                                                                                                                                                                                                       | 142 |
| 5.15 | Anchors positions normalized to $5c$ . . . . .                                                                                                                                                                                                                                                                                                                                                       | 143 |
| 6.1  | Packet transmission from anchors in the collision-free scheme. Here, each anchor transmits its packets according to its index value (ID number). All links between anchors are assumed to function properly in this figure (there are no missing links). . . . .                                                                                                                                     | 152 |
| 6.2  | Packet transmission from anchors in the collision-tolerant scheme. Here, each anchor transmits its packets at random according to a Poisson distribution. . . . .                                                                                                                                                                                                                                    | 153 |
| 6.3  | Probability of successful localization for different values of $\lambda$ and $T_{CT}$ . . . . .                                                                                                                                                                                                                                                                                                      | 162 |
| 6.4  | a) Probability of successful packet reception versus different number of interferers. b) Probability that $q$ interferers collide with the desired packet. For this figure, $\lambda_{low}$ , $\lambda_{opt}$ and $\lambda_{upp}$ are chosen from Fig. 6.3. . . . .                                                                                                                                  | 165 |
| 6.5  | a) Effect of transmit power on the minimum time required for localization, and the average probability of a packet-loss $\bar{p}_l$ (dashed-line); b) The minimum value of $\frac{P_l}{P_T}$ in dB below which the average energy consumption of CTS is greater than that of CFS.                                                                                                                    | 166 |

6.6 Effect of packet length on the minimum time required for localization. 167

6.7 Effect of the operating area size on the time required localization. 168

6.8 Probability distribution of the localization error, and its corresponding CRB for CTS and CFS. . . . . 169

7.1 The chronological order of our contributions during the Ph.D. based on the date of the conferences, published papers or technical reports. **C** stands for conference paper, **J** for journal paper, **P** for project meeting, and **T** for technical report. The purple line shows the duration I visited Professor *Milica Stojanovic* at Northeastern University. . . . . 204



# List of Tables

|     |                                                                                                                                                                                                                    |     |
|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| 1.1 | Comparison of different technologies for underwater communications [7]. . . . .                                                                                                                                    | 6   |
| 3.1 | All possible patterns that a fastest ray in a shallow underwater environment can follow. . . . .                                                                                                                   | 91  |
| 5.1 | Localization time vs. different transmission range and different algorithms. The $^{(M)}$ shows the number of used subchannels in DMC.                                                                             | 142 |
| 6.1 | Possible times that anchor $j$ transmits its packet. . . . .                                                                                                                                                       | 153 |
| 6.2 | Values of $\theta_s$ and $\theta_e$ based on distance $d$ . . . . .                                                                                                                                                | 161 |
| 6.3 | Simulation parameters. Note that, in this table some parameters such as $N$ , $D_{aa}$ , $T_g$ , etc. are related to other parameters, e.g., $N$ depends on the values of the $\bar{p}_l$ , and $P_{ss}$ . . . . . | 163 |



---

---

# **PART I**

---

## **PREFACE**



# Chapter 1

## Introduction

### 1.1 Underwater acoustic history

The first observation of underwater sound propagation goes back to Aristotle's era over 2000 years ago. No new developments in underwater acoustics were presented until the 14-th century when Leonardo Da Vinci documented that a ship can be heard at great distances underwater. Later, in spite of the fact that fundamental studies of Marin Mersenne, Galileo and Isaac Newton during the 16-th and 17-th centuries were shaping the foundations of acoustic physics, no progress in underwater acoustics is noted until Colladon and Sturm measured the sound speed in 1826. Soon after this experiment, echo sounding was labeled as the first underwater acoustic application, and it was designed to determine the depth of the sea. In the same century, the theory of sound was published by John William Strut (Lord Rayleigh) which smoothed the path for the emergence of the first practical uses of underwater acoustics in the 19-th century, namely, navigational safety and echo ranging. This was used during World War I (1914-1918) to detect submerged submarines and mines. The detection range was further enhanced by introducing the piezoelectric effect which explains the conversion of sound pressure to electricity.

Short after World War I, scientists developed the theory of sound propagation, and discovered that small changes in temperature, salinity, and pressure in parallel with ocean currents would affect the way sound travels underwater. During this period non-military applications such as seabed mapping for telegraph cable deployment, seafloor sublayer analysis for oil and gas extraction, seismology, and

acoustic detection of fish have been developed.

In spite of the fact that World War II (1941-1945) is the deadliest conflict in human history, it leads to the development of sound navigation and ranging systems (sonar), the discovery of irregular acoustic propagation, the extension of underwater acoustic applications into marine biology, and the basis of fish-finding sonars. In addition, the need for triggering acoustic mines for approaching ships demanded research on the ambient noise level which was later used for oceanology and ocean thermography. Moreover, the advent of the hydrophone array in sound surveillance systems, and the usage of low frequency analysis and recording invented by the AT&T company enabled the USA to monitor the soviet submarines more accurately at long distances. Nowadays, this infrastructure is mostly used for marine mammal studies, and for measuring the large-scale ocean temperature variability.

It may be claimed that the ping-based use of sonars during World-War II can be considered as digital communications; however, the general principle of signaling and digital modulation did not appear until the 1960s [8]. Still, non-coherent modulations such as most frequency-shift keying (FSK) and frequency hopping (FH) schemes were the common ways for digital communications, but demands for higher data-rates drew the attention of engineers towards phase-coherent modulations in the 1990s [9]. At the same time, different companies such as Datasonics and Benthos were introducing new digital-based commercial acoustic modems. Up to now, single carrier modulation techniques were the core of underwater digital communications, and a lot of efforts such as designing sophisticated equalizers and coding, were taken to combat the severe effects of the underwater channel such as a limited bandwidth, long propagation delays, multi-path, a rapid time variation, fading, and large Doppler shifts [10]. Through an experiment conducted in the Mediterranean sea in 2004 [11], it turns out that multi-carrier techniques such as orthogonal frequency division multiplexing (OFDM) can be a good candidate for underwater acoustic communications. This was followed by a great deal of research on OFDM-based channel coding, channel estimation, designing interleavers or waveforms for maximizing the frequency and time diversity [12], adaptive modulation and coding techniques [13], and sophisticated iterative detection algorithms [14]. For instance, in some wide-band underwater acoustic communication systems, it is observed that the multi-path arrivals are usually discrete which results in a superior performance of elaborate multi-carrier sparse channel estimation and data recovery techniques [15].

## 1.2 Underwater communications

Underwater communication is not limited to the use of acoustic signals. Other ways of energy transmission such as using electromagnetic waves [7], optical waves [16], and magnetic induction [17], can be employed for underwater information exchange. It is also claimed that the human voice can change the structure of the water molecules [18]. This may be considered as a local memory of the underwater medium, and a way of underwater communications.

Electromagnetic and optical waves suffer from a high attenuation. Depending on their transmit power and operating frequency they can barely propagate more than 100 m. In order to find out what is the best way of underwater communications, a comparison between acoustic waves, electromagnetic waves, and magnetic induction has taken place through an experiment [19]. Table 1.1 shows the relationship between the distance and the achievable data-rate when different technologies are used for underwater digital communications. Although, it has been shown that for short-range communications, magnetic induction is a better candidate (in terms of data-rate and propagation delay) than acoustic signals (below 10 m) and its communication range can be extended up to a few hundred meters through a waveguide technique; still, for large distances and telemetry applications, low-frequency acoustic signaling is the most versatile and widely used physical layer technology. However, underwater acoustic communications is quite challenging [20]. Below, we have listed the well-known challenges that engineers are confronted with, in designing an underwater communication system.

**Attenuation:** The acoustic signal is attenuated mainly because of two reasons; spreading (geometrical) and absorption. Spreading (because of sound propagation in different directions) can be spherical (deep sea), or non-spherical (due to reflection at boundaries, i.e., related to multipath). In a medium with upper and lower boundaries, a cylindrical model can be considered as a simple approximation for spreading loss.

The main reason for absorption (especially for higher frequencies) is the viscosity, and the other reason (lower frequencies) is ionic relaxation of boric acid and magnesium sulfate.

**Variable sound-speed:** As mentioned earlier in Section 1.1, the underwater sound speed is not constant. It varies with pressure, temperature and salinity of the medium. In deep water where the water temperature and salinity are constant, the sound speed is linearly proportional to the depth (pressure),

Table 1.1: Comparison of different technologies for underwater communications [7].

| Technology            | Working frequency | Modulation | Distance | Data rates |
|-----------------------|-------------------|------------|----------|------------|
| Optical waves         | -                 | PPM        | 1.8 m    | 100 kbps   |
|                       | -                 | -          | 10 m     | 10 Mbps    |
|                       | -                 | -          | 11 m     | 9.69 kbps  |
| Electromagnetic waves | 2.4 GHz           | CCK        | 0.16 m   | 11 Mbps    |
|                       | 2.4 GHz           | QPSK       | 0.17 m   | 2 Mbps     |
|                       | 1 kHz             | BPSK       | 2 m      | 1 kbps     |
|                       | 10 kHz            | BPSK       | 16 m     | 1 kbps     |
|                       | 3 kHz             | -          | 40 m     | 100 bps    |
|                       | 5 MHz             | -          | 90 m     | 500 kbps   |
| Acoustic waves        | 800 kHz           | BPSK       | 1 m      | 80 kbps    |
|                       | 70 kHz            | ASK        | 70 m     | 0.2 kbps   |
|                       | 24 kHz            | QPSK       | 2500 m   | 30 kbps    |
|                       | 12 kHz            | MIMO-OFDM  | -        | 24.36 kbps |

while in shallow water it varies from day to day, and has a divergent behavior in different seasons of the year. Although the sound speed can vary in any direction, in most of the underwater applications, it can be assumed that it is only a function of depth.

An example of a sound speed profile (SSP) in shallow and deep water is shown in Fig. 1.1 and Fig. 1.2, respectively. In general, the SSP in shallow water depends on the season and location of the ocean, and it can be modeled by three simple profile [21], mixed winter profile (fully mixed temperature conditions resulting in constant sound speed), three-layer summer profiles, and coastal front (two water masses with differing temperature profiles).

**Multi-path and Doppler:** Multipath in the underwater medium is governed by two effects: sound reflection at the surface, bottom, and any other object, and sound refraction in the water. The latter is a consequence of the spatial variability of the sound speed. In shallow water, the channel delay spread can be extended to tens or even hundreds of milliseconds [10]. The motion of the underwater medium due to the wind, ship movement, or any other marine creature, and the low speed of sound propagation causes the underwater channel to vary fast, and consequently to follow the characteristics of a so-called doubly-selective channel which results in a frequency-selective (due to the multi-path) as well as time-selective (due to the Doppler spread) signal distortion [22].

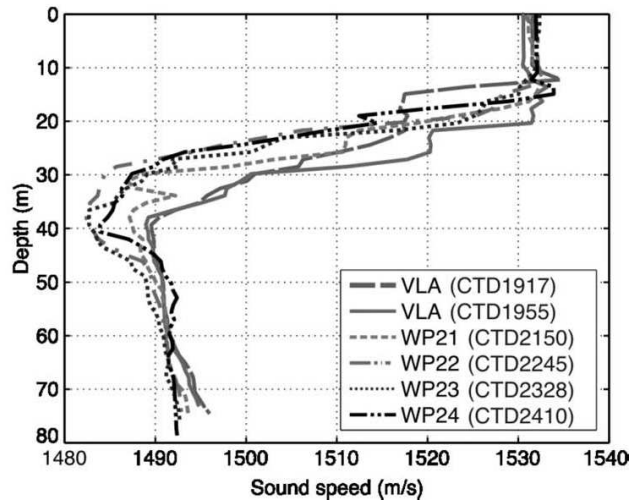


Figure 1.1: Measured sound-speed profiles during the acoustic transmissions for different times of the day as indicated in the legend [1].

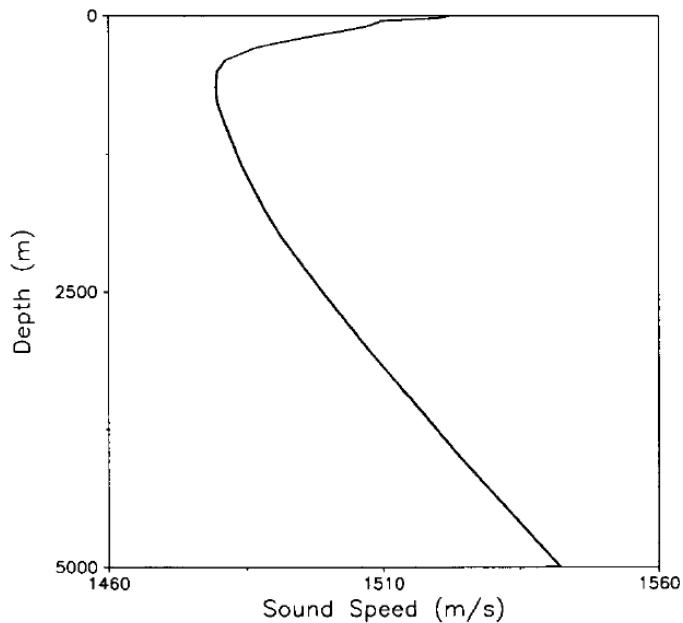


Figure 1.2: Sound speed profile for the Pacific environment [2].

**Noise:** In addition, beyond the ambient noise, man-made noise sources such as the one caused by marine machinery (offshore drilling, ships and vessel movements), and biological activities (shrimps) play a significant role in the quality of the received acoustic signal. Sometimes, the noise caused by shrimps is so powerful that submarines find it impossible to pick up other signals using sonar.

**Scattering and reverberation:** Bubbles and particles at the water surface and seabed cause scattering which results in a loss of coherence. Reverberation adds to the overall noise level but in a less predictive way than noise due to the sea state, shipping and thermal effects.

**Signal propagation:** The signal propagation model in most of the terrestrial applications is related only to the transmission distance, while the underwater acoustic propagation is also affected by the signal frequency. Theoretically, we can divide the acoustic propagation models into five groups: ray theory model, normal mode model, multi-path expansion model, fast field model, and parabolic equation model which all are derived from the wave equation [23].

### 1.3 Underwater acoustic contemporary research

Different research groups are actively participating in conducting experiments, and developing ideas to make the future underwater acoustic networks more reliable and efficient. Nowadays, we observe that acoustically directed robots are replacing divers in performing maintenance of underwater platforms [20]. The emergence of such powerful and advanced robots opens the gates for new applications, and triggers further discoveries and explorations.

The Kastner Research Group from the University of California, San Diego is developing new computer vision and image processing algorithms for rapidly and successfully detecting, categorizing, and creating a 3D model of underwater objects [24]. In order to accomplish this goal, they have developed a hardware-accelerated digital signal processing system to enable real-time processing of multi-frequency and multi-beam echosounders.

The next generation of primary standards for underwater acoustics project (coordinated by NPL, UK) offers standards for testing of acoustic transducers at simulated ocean states by utilizing the NPL acoustic pressure vessel (APV) [25]. That reduces the cost of a sea-trial of underwater equipment by an order of

magnitude. Up to now, the project has succeeded to simulate ocean conditions up to a depth of 700 meters for low-frequency (a few kHz) equipment. Besides, NPL is also coordinating on the development of a novel optically-based prime standard for the calibration of underwater hydrophones derived from laser Doppler anemometry and heterodyne interferometry. This is needed in order to enable the calibration of present hydrophones at 1 MHz frequencies.

A number of research projects, e.g., ARGO, Seaweb, RACUN, JANUS, UCAC, GREX, TRIDENT, VENUS, SHOAL, C4C, have been defined recently to address specific features of underwater communications and robotics. The ARGO project monitors temperature, currents and salinity in the oceans. Seaweb networks unite the manned command centers through acoustic, radio, fiber and wire gateway connections [26]. The RACUN project facilitates a network centric approach to the underwater surroundings [27]. JANUS basically offers a digital and open-source signaling to accomplish underwater public communications [28]. The UCAC project aims to set up covert communications [29]. Covert communications are carried out at a low signal-to-noise ratio to avoid detection or interception by an eavesdropper [30]. GREX coordinates and controls the collaborating heterogeneous unmanned systems in uncertain environments. The main goal is to attain a first level of disseminated intelligence via reliable interrelated systems and to help towards the harmonized completion of tasks [31]. The TRIDENT project [32] came up with new goals and methods for multipurpose interventional tasks for underwater systems. These hold miscellaneous potential applications and go beyond existing methods characteristically derived from purpose-built and/or manned systems. VENUS is an underwater observatory system and it is a component of the ORION network [33]. The SHOAL research project is related to developing underwater robots. These robots evaluate the waters, recognize leaks from oil pipelines or chemical pollutants in harbors [34]. C4C aims to investigate the ocean along with the ocean floor for environmental reasons. It also aims to determine the involved control and coordination problems [35].

Despite the great technological advances in underwater robotics, still most of the applications rely on human operators who can assess and understand information gathered by the network elements, and can remotely navigate the vehicles to fulfill a particular mission. However, it takes time and money to train such skilled personnel. Furthermore, the involvement of humans in an operation is risky due to unexpected human errors, and it might limit the dimensions of an operation. To address this problem, the goals of the recently conducted project NOPTILUS is to design and deploy teams of autonomous underwater

vehicles (AUVs) that can fully autonomously take over real-life complex situation awareness operations such as environmental monitoring and clean-up operations, seafloor mapping, security and surveillance, inspection of underwater structures, etc. [36].

The research presented in this thesis was carried out in the frame of the NOPTILUS project.

## **1.4 Underwater acoustic localization**

Similar to terrestrial wireless sensor networks (WSNs), applications of underwater sensor networks can be divided into three main domains [37]. First, scientific applications where scientists try to get a better picture about the animal life (i.e., micro-organisms, fish or mammals), and geological processes through analysis of water characteristics such as temperature, salinity, conductivity, pressure, oxygen level, bacterial and other pollutant content. Second, industrial applications in which companies use modern monitoring and control activities to build, inspect and maintain robust infrastructures (such as pipelines) to extract oil or minerals. Finally, there are military and homeland security applications which are related to surveillance and securing port facilities, and monitoring ships and submarines.

In most of the underwater sensor network applications, the collected data has to be labeled by their position, and time of measurement. Basically, the measured data without a position tag is usually meaningless. For instance, the water temperature is a location-based phenomenon and in order to find the temperature field from the scattered samples the positions at which they are measured are required. In a few costly old sensor networks the nodes are fixed and their locations are known. In contrast, newly developed networks contain ad-hoc deployable bottom nodes and mobile nodes such as buoys, AUVs, and remotely operated vehicles (ROVs) employed for data-gathering. As a result, underwater sensor network localization becomes an important task of the network.

Knowing the positions of the sensor nodes benefits other network tasks as well. For instance, more efficient routing algorithms for transferring data to the destination can be designed. Knowing the path-loss model, adaptive modulation and coding between two nodes increase the network efficiency. Furthermore, the medium access control protocols can be designed to minimize the collision probability, and maximize the network throughput. In addition, if two nodes know their relative distance, they can adjust their power level, and consume less energy during peer-to-peer communications.

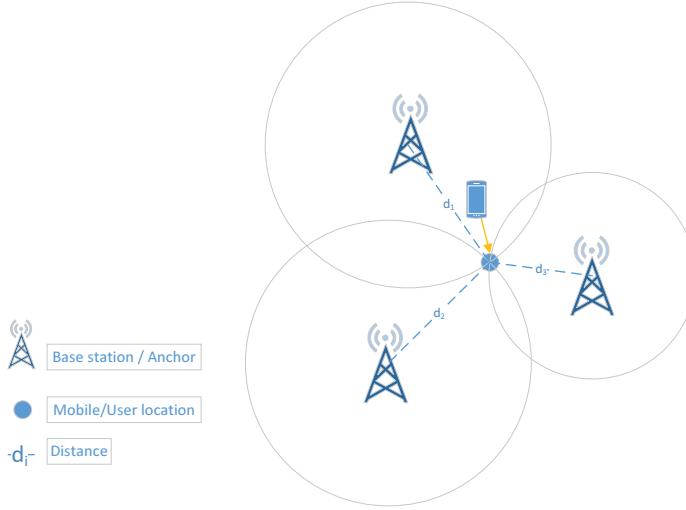


Figure 1.3: Trilateration using three anchors.

In order to find the absolute position of sensor nodes, underwater acoustic sensor networks (UASNs) use a few reference nodes whose positions are known a priori. These nodes are called references, or anchors. Below we have listed some basic localization algorithms and explain how anchors are used to estimate the location of a (passive or ordinary) sensor node.

**Trilateration:** In a two-dimensional (three-dimensional) coordinate system, if the distance of a point to three (or four) reference points is known, then the position of this point can be determined geometrically as the intersection point of three circles (four spheres) as shown in Fig. 1.3. If we assume that  $\mathbf{x}_i$  is the position of the  $i$ -th point, then the target position,  $\mathbf{x}_0$ , is the solution of the following set of equations:

$$\|\mathbf{x} - \mathbf{x}_i\| = d_i, \text{ and } i \in \{1, 2, \dots, N\} \quad (1.1)$$

where  $d_i$  is the distance of the target point to the  $i$ -th reference point, and  $N$  is the number of anchors. In some applications where the orientation of the target point with respect to the reference point is known, only two anchors are enough for localization. This simple algorithm is used for rough underwater localization in the NOPTILUS system as shown in Fig. 1.4.

**Multidimensional Scaling:** In a network of  $N$  nodes, the distance matrix is defined as a symmetric square matrix  $\mathbf{D}$  with the  $(i, j)$ -th element given by  $d_{ij}^2 = \|\mathbf{x}_i - \mathbf{x}_j\|_2^2$  which is the squared distance between the  $i$ -th and the  $j$ -th nodes. The distance matrix has interesting properties. First, all its elements are positive. Second, all elements on the diagonal are zero, and thus the trace of the distance matrix is zero. Third, its elements support the triangular inequality as  $d_{ik} + d_{kj} \geq d_{ij}$ . Forth, in a  $d$ -dimensional space, its rank is less than or equal to  $d + 2$ . Finally, if we use double centering by multiplying both sides of  $\mathbf{D}$  by the centering matrix,  $\mathbf{H} = \mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^T$ , then the rank of the obtained matrix  $\mathbf{B} = \mathbf{H}\mathbf{D}\mathbf{H}^T$  is equal or less than  $d$ . The Multidimensional Scaling (MDS) algorithm uses this last property to extract the relative positions of the nodes from the double-centered distance matrix. By using the singular value decomposition,  $\mathbf{B} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$ , and selecting the corresponding eigenvectors of the non-zero eigenvalues,  $\mathbf{U}_{[:,1:d]}$ , the coordinate matrix which gives a relative map of the nodes can be obtained as  $\mathbf{X} = \mathbf{U}_{[:,1:d]}\mathbf{\Lambda}_{[1:d,1:d]}^{\frac{1}{2}}$ . The recovered matrix  $\mathbf{X}$  is rotated and translated, as it has a different location as the original position matrix. Still, if the absolute positions of  $d + 1$  nodes ( $\{\mathbf{x}_1, \dots, \mathbf{x}_{d+1}\}$  which span a  $d$ -dimensional space) are known, the absolute position of all other nodes can be found. The main problem of the MDS algorithm is its complexity and its centralized property.

In the presence of measurement noise, the rank of the distance matrix which is built with noisy measurements is not necessarily equal or less than  $d + 2$ , and it can be anything. Under such conditions, the estimate of the nodes' relative position matrix  $\hat{\mathbf{X}}$  can be found by selecting the first  $d$  dominant eigenvalues of  $\mathbf{B}$ , and their corresponding eigenvectors [38].

**Least Squares:** With noisy distance measurements, it is shown that the least squares (LS) method performs much better than trilateration [39]. Given  $N$  distance measurements from a node to  $N$  reference nodes, the LS solution can be obtained as

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 \rightarrow \hat{\mathbf{x}} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}, \quad (1.2)$$

where  $\hat{\mathbf{x}} = [\hat{x} \ \hat{y}]^T$  is the position estimate of the unknown node, and  $\mathbf{A}$  is the

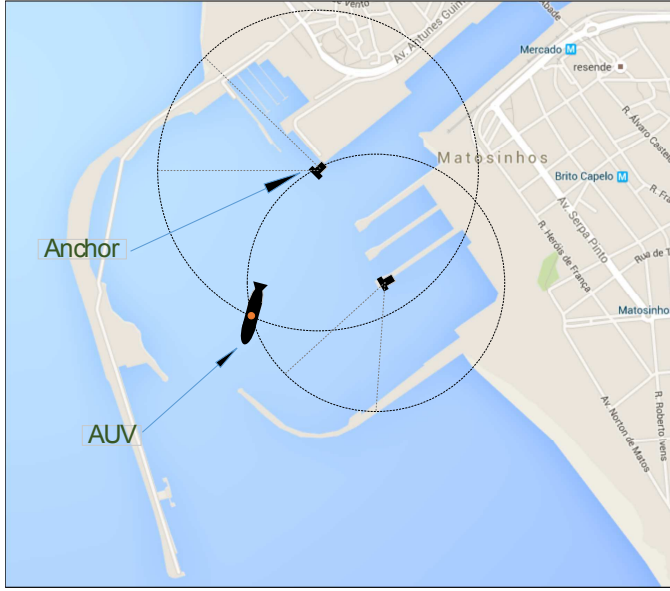


Figure 1.4: Underwater localization of UAVs using only two fixed anchors.

$(N - 1) \times 2$  matrix, and  $\mathbf{b}$  the  $(N - 1) \times 1$  vector defined as

$$\mathbf{A} = \begin{bmatrix} 2(\mathbf{x}_1 - \mathbf{x}_n)^T \\ \vdots \\ 2(\mathbf{x}_{n-1} - \mathbf{x}_n)^T \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} x_1^2 - x_n^2 + y_1^2 - y_n^2 + d_n^2 - d_1^2 \\ \vdots \\ x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2 + d_n^2 - d_{n-1}^2 \end{bmatrix}.$$

where  $\mathbf{x}_k$  is the position of the  $k$ -th anchor, and  $x_k$  and  $y_k$  are the  $x$  and  $y$  coordinates of the  $k$ -th anchor, respectively.

**Optimization:** Network localization can also be modeled by a non-linear optimization problem as

$$\hat{\mathbf{X}} = \arg \min_{\mathbf{X}} \sum_{i=1}^N \sum_{j=i+1}^N \|\mathbf{x}_i - \mathbf{x}_j\| - d_{i,j}\|_2^2. \quad (1.3)$$

The optimizer tries to minimize the differences between the measured distances and the corresponding Euclidean distances. Note that the solution corresponds to the maximum likelihood estimate if the measurement noise is Gaussian. Different tools such as multi-resolution search [40], particle swarm optimization [41], conjugate-gradient [42] and Newton-Raphson [43]

can be used to find the local or global solution of the problem.

### 1.4.1 Different categories of localization algorithms

- **Range-based or range-free:** In range-based localization algorithms, an ordinary node estimates its distance to different reference points which are usually located within its communication range. Different metrics can be used for distance estimation such as time-of-flight (ToF), time-difference-of-arrival (TDoA), round-trip-time (RTT), received signal strength (RSS), the signature of the channel impulse response [44], and features (such as geometric features) [45].

In range-free algorithms, a node estimates its position within a certain area for instance by looking at the photos of different places (imaging), or at the map where the node is currently located (using bathymetric sonar) which is also called the geophysical approach. Localization based on angle of arrival (AoA) [46], the Area Localization Scheme (ALS) [47] based on a recording of the anchor node IDs and their corresponding power levels, or the finger-printing approach [48] can also be categorized in this group. In finger-printing, the combinations of various reference point signatures are used which enables the sensor node to find its position without any range estimation or multilateration.

- **Distributed or centralized:** In distributed localization algorithms, each node collects relevant information and then estimates its position individually, while in the centralized approach, a fusion center combines the information and estimates the positions of all the nodes. In [49], different localization algorithms are listed and categorized in two groups, distributed and centralized. Distributed localization algorithms do not need a central unit (fusion center) to estimate the node locations [50].
- **Static or mobile reference:** As it is clear from the name of this category, in static-reference algorithms the anchors are fixed while in mobile-reference algorithms they can move freely in the operating environment. The algorithms based on mobile (relative) reference points are more popular in current underwater networks due to the simple deployment of the network.
- **Line-of-sight or non-line-of-sight:** The assumption of line-of-sight (LOS) communications between a pair of nodes is not valid in all underwater

operating areas. In a multi-path environment, the signal is received at different discrete times with different powers. Given the time-of-arrival and power of each path, the LOS path can be distinguished from the non-line-of-sight ones [51], and appropriate ranging estimation can take place.

- **Single stage or multi-stage:** The difference between multi-stage and single-stage algorithms is that in the former, an ordinary node becomes a reference after it estimates its position, while this does not happen in the latter. In some literature, multi-stage algorithms are also called cooperative [52].
- **Synchronous or asynchronous:** Synchronous localization algorithms can only be used in synchronous networks where all the sensor nodes (either ordinary or reference nodes) are synchronized with each other. In a hybrid mode, i.e., in TDoA range estimation, the reference points are synchronized with each other while the ordinary nodes are not synchronized. Asynchronous networks which are often encountered in practice are obliged to use asynchronous localization algorithms which are either based on RTT or TDoA range estimation [53]. Note that, synchronizing reference points with each other is not a complicated task if they are located on the ocean surface (using radio).

### 1.4.2 How to evaluate a localization algorithm

Several factors help us to evaluate a localization algorithm. Note that these factors explained below are not totally independent of each other.

- **Localization accuracy:** It is usually a function of the distance between the sensor's actual position and its estimate. In some literature, the mean of the squared estimation error (MSE) represents the localization accuracy.
- **Localization time:** It is the time required for the sensor nodes to be localized. It may include the processing time, and the packet exchanging time. A faster localization time allows for a more dynamic UASN, and a more dynamic environment. The method and the order in which localization packets are emitted from the anchors play a major role in minimizing the localization time. We will address this problem in detail in this thesis.
- **Localization scalability:** A localization algorithm is scalable if its processing core (main code) does not change with the size of the operating area, and

the number of sensor nodes. As the size of the network increases, more nodes will lose their communication with each other. If the localization algorithm is independent of the network size, and how the ordinary nodes are connected, then it is scalable.

- **Localization coverage:** Coverage is the percentage of sensor nodes which can be localized during the localization time. The localization coverage of a network is related to the physics of the signal propagation, how many anchors are employed in the network, the anchors' deployment, and how the sensor nodes cooperate with each other.
- **Energy consumption:** The energy that the network consumes during the localization task is also an important factor to evaluate an algorithm. The energy consumption includes the energy required for packet transmission/reception and what sensors consume when they listen to the channel.
- **Message communications:** The average number of message exchanges between the sensor nodes is an important factor in the energy consumption, and localization time. An algorithm that requires fewer messages for the localization is preferable due to the long propagation delays of underwater acoustic communications.
- **Computational complexity:** The computational complexity of a localization algorithm is a limiting factor in using an algorithm in a practical situation. Generally, more complex algorithms lead to a better accuracy, but more energy consumption, and more processing time.

Other than acoustic modems (for acoustic communications), using additional equipment such as a global positioning system (GPS), compass, pressure sensor, camera, sonar, magnetometer, inertial navigation system (IMU), echo sounder (sonar device for measuring the distance to the sea floor or to the sea surface), and Doppler velocity log (DVL) would enhance the performance of localization or navigation algorithms [54]. Probably a GPS module, compass and pressure sensors are the cheapest (around 100 USD) products that a sensor node can be equipped with. GPS only works when a sensor node surfaces. The compass shows the earth's magnetic north pole with  $1^\circ$  or  $2^\circ$  accuracy, and via the pressure sensor, a node can determine its depth from the surface with high accuracy (around 10 cm) in calm sea. A sonar, IMU, and DVL are expensive devices and usually submarines and AUVs use them. Sonar can detect objects located underwater, and can use them as pilots

for the vehicle's position estimation. An IMU uses accelerometers, gyroscopes, and magnetometers to estimate the vehicle's orientation, and velocity in three different axes (Cartesian or Polar coordinates). DVL can be used when the AUV is moving close to the seabed. DVL transmits acoustic pulses (usually in high frequencies), measures the Doppler shift, and determines the vehicle's velocity very accurately (around 0.5 cm/s).

### 1.4.3 Generic localization problem

For a UASN with  $M$  sensor nodes operating in a  $d$ -dimensional environment, the localization problem at time  $t_n$  can be generally formulated as

$$\hat{\mathbf{X}}_n = \arg \max_{\mathbf{X}_n} P(\mathbf{X}_n | \mathbf{y}_n, \mathbf{y}_{n-1}, \dots, \mathbf{y}_1, \hat{\mathbf{X}}_{n-1}, \dots, \hat{\mathbf{X}}_1) \quad (1.4)$$

where  $\mathbf{X}_n$  is an  $M \times d$  matrix holding the positions of the  $M$  sensor nodes at time  $t_n$ ,  $P$  is the probability function,  $\hat{\mathbf{X}}_j$  is the estimate of the sensors' positions at time  $t_j$ , and  $\mathbf{y}_j$  is a vector of  $K$  noisy measurements taken at time  $t_j$  with its  $k$ -th element given by

$$y_j^k = f(\mathbf{X}_j, \mathbf{X}_{j-1}, \dots, \mathbf{X}_1) + n_j^k, \quad (1.5)$$

where  $f(\cdot)$  is in general a nonlinear function that relates the measurements to the sensors' positions, and  $n_j^k$  represents the measurement noise.

In the second part of this thesis we will show how such generic localization problems can be used for UASN ranging and localization by finding  $f(\cdot)$  for different underwater scenarios.

### 1.4.4 Review of important localization algorithms

Recently, underwater localization has drawn a lot of attention in academia and industry. Nowadays, researchers are developing different localization algorithms for various underwater applications spanning from simple multilateration techniques to more complex network-based collaborative methods for deep and shallow underwater media.

The state-of-the-art underwater localization algorithms have been extensively examined in recently published literature. Classical and newly developed terrestrial localization algorithms are summarized in [55], and their suitability for underwater applications is analyzed. In [49], a comprehensive survey of underwater architectures and localization methods is presented. The authors of [54] present a review of AUV navigation and localization, as well as a description of some of

the more commonly used methods. In [56], underwater localization algorithms are categorized in different groups, and compared with each other in terms of some of the criteria explained in Section 1.4.1.

For this section, we review some of the well-known localization algorithms which are designed specifically for underwater applications.

### **Long baseline (LBL), short baseline (SBL), and ultra short baseline (USBL) systems**

As well-established underwater positioning systems, Long Baseline (LBL), Short Baseline (SBL) and Ultra Short Baseline (USBL) systems have been widely used in various underwater applications, especially for offshore oil and gas exploration.

LBL systems use networks of sea-floor mounted baseline transponders as reference points for navigation (see Fig. 1.5). The position estimated by LBL systems is very accurate (generally better than 1 meter and can reach a few centimeters accuracy) and independent of the water depth. These systems are usually used for accurate underwater survey work where the precision or position stability of ship-based positioning systems (i.e., SBL, USBL) does not suffice.

In order to determine the location of an object in an LBL system, a sequence of actions is required. At first, the vehicle to be positioned transmits an acoustic pulse from its transducer. This pulse travels through the water and reaches each of the LBL transponders. The transponders detect the signal and respond with a unique transponder acoustic pulse. After receiving the transponder pulses at the target, it determines the round trip acoustic travel times to each of the transponders in the LBL array. Then the RTT measurements are converted to ranges to the transponders by using the sound speed of the water. The sound speed is usually measured at the target location if all network elements operate at the same depth. Otherwise, the average sound speed collected from a vertical profile is used.

There is another type of LBL system where the transponders are replaced by beacons that transmit at fixed times (like the GPS system). Such a system has to be synchronized, and consumes more energy due to continuous transmission. However, the vehicle saves energy and can stay covert.

In contrast to LBL systems, SBL and USBL systems are easy to install. They normally employ a smaller transducer array to determine the range and bearing to a transponder that is co-located with the target. The sequence of events required for target localization is almost the same as for LBL systems. However, as the transponders are attached to a boat or small ship and move with water drift, the

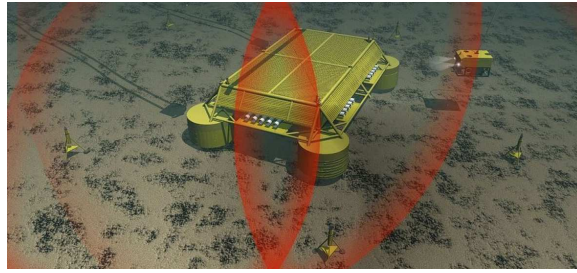


Figure 1.5: The Sonardyne LBL system mounted on the seabed, consisting of four anchors [3]. The system is used for accurately locating an underwater structure.

accuracy of these systems can be lower than that of LBL systems. Combinations of LBL, SBL, and USBL can also be considered (see Fig. 1.6).

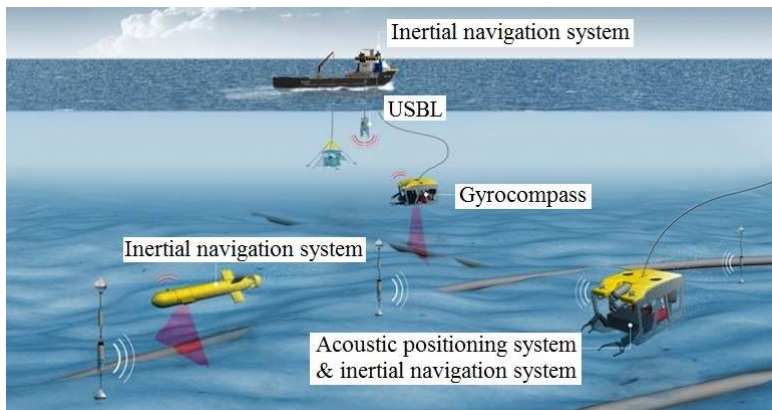


Figure 1.6: Combination of LBL and SBL systems for positioning [4]. The SBL transducers are attached to the ship in the picture, while the LBL system is mounted on the seabed.

As an extension of the LBL systems, the underwater transponder positioning (UTP) integrates range measurements from one or several transponders at different times for position estimation. In addition, the system is coupled with INS and has a better estimation accuracy in comparison to LBL systems [57]. In this positioning algorithm, a transponder is fixed at the seabed, and provides AUVs with ranges relative to its fixed geographical location. This information can be fused and used for aiding INS. Based on a few experimental results, it has been shown that UTP can be used in AUVs which require accurate and real time localization information [57].

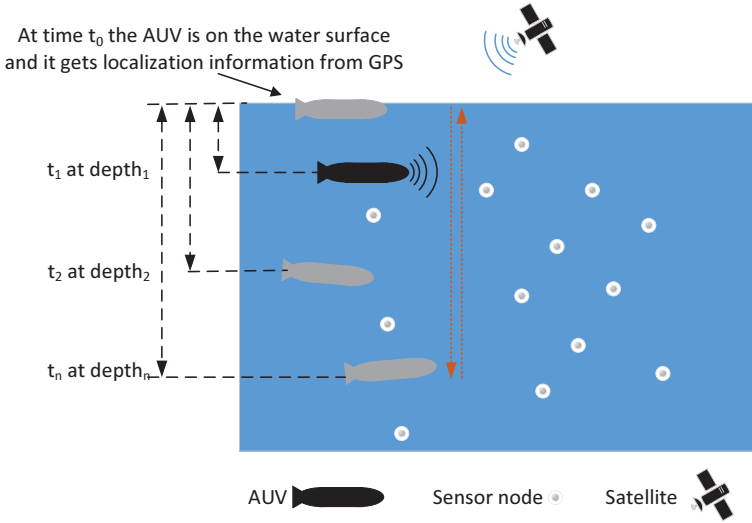


Figure 1.7: The vertical movement of an AUV in a network employing the Dive and Rise localization algorithm.

### Dive and rise (DNR) positioning

The idea of Dive and Rise (DNR) positioning is introduced in [58]. This algorithm is designed for synchronous networks consisting of DNR anchor nodes which are equipped with GPS, and can move vertically underwater as shown in Fig. 1.7. The anchor nodes need to surface regularly to collect GPS data. In the meantime, they broadcast localization packets to ordinary nodes while they are sinking and rising (at different water depths). Ordinary nodes listen to those packets and calculate the ToF, and equivalently their distance to several anchor positions. Then, through a simple multilateration, they estimate their position passively. The frequency of the dive and rise repetition is much less than that of message propagation, and that makes the localization task slow. In order to speed up the process, the authors of [59] combined the DNR positioning technique with a multi-stage strategy in which ordinary nodes are converted to anchors (reference points) after they estimate their position. That would reduce the localization accuracy of the nodes which got ToF measurements from reference points.

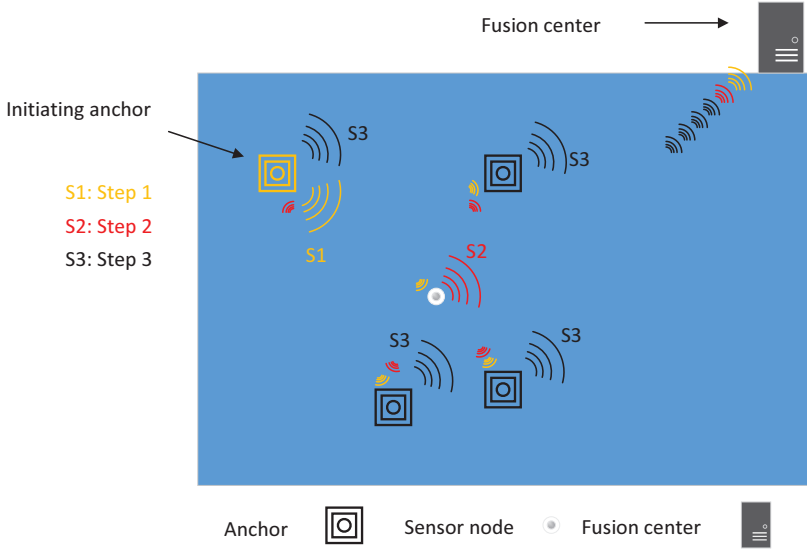


Figure 1.8: Different message propagation steps in the Asymmetrical Round Trip based Localization algorithm.

### Asymmetrical round trip based Localization (ARTL)

Asymmetrical Round Trip based Localization (ARTL) [60] is an asynchronous algorithm which relies on RTT measurements. The ARTL assumes that anchor nodes can receive their own packets, i.e., they listen to the channel all the time. As it is shown in Fig. 1.8, the initiator anchor transmits a ranging request packet. This packet is received by the ordinary node, as well as the other anchors at different times. After a short time, the ordinary node transmits an ACK packet which will be received by the anchors. Having calculated the time difference of arrival of the received signals, the anchors transmit this information to the fusion center, and there a localization task will be launched.

It is clear that neither a synchronization algorithm, nor a complex localization task is performed by the network nodes, and that saves energy and cost. Furthermore, very little communication overhead is required (only  $N + 1$  broadcast packets). The hidden drawback of this algorithm is that in case the anchors are not connected to a central unit via a cable, they might consume most of their energy while they are transmitting information to the fusion center. It should be noted that the localization time of ARTL can be long due to the information exchange with the central unit.

### **Node discovery and localization protocol (NDLP)**

The Node Discovery and Localization Protocol (NDLP) is a GPS-less and anchor-free scheme which is used to manage sub-sea localization [61]. Being capable of determining the relative positions of neighboring nodes, the primary anchor node initiates the localization task, and selects the farthest anchor node which is within its communication range. The third anchor is selected inside the coverage range of the first and the second one, and to be located far from them. In this way, all the nodes which are located in the coverage area of the three selected anchors can be localized. A node which is located in the communication range of only two anchors can obtain only two distance measurements. The node keeps these measurements and waits for the network to select a fourth anchor. The fourth anchor (based on a given algorithm) is selected in such a way that it provides the widest possible coverage to the network. This procedure continues until the whole operating area is covered.

Although the NDLP tries to minimize the number of anchors to cover the whole area and saves broadcasting energy in the node discovery phase, it is quite slow. Furthermore, in a sparse network with a few anchor nodes this strategy fails as the node discovery algorithm stops when there is no other anchor in the vicinity. In addition it is not suitable for a mobile network, because in the second phase of the node discovery (selecting the fourth anchor), the ordinary node might have changed its position.

### **AUV-aided and multi-stage AUV-aided localization (MS-AUV)**

The AUV-aided localization scheme is designed for an asynchronous mobile UASN with many ordinary nodes and one AUV [62]. The sensor nodes are distributed at random in the operating area, and are able to communicate omni-directionally with the AUV. The AUV repeatedly surfaces, gets its position information through GPS, and descends following a specific trajectory. Meanwhile it broadcasts “wakeup” messages, and receives back “request” packets from ordinary nodes. As a reply to the “request” packets, the AUV sends “respond” packets. In this way, the ordinary nodes can estimate their distance to where the AUV was located. Since the ordinary nodes are equipped with a pressure sensor, they only require to estimate their position in the  $x - y$  plane as shown in Fig. 1.9.

The localization accuracy and coverage of the AUV-aided algorithm is satisfactory. However, the accuracy depends on how accurate the AUV can estimate its location while it is inside the water. Furthermore, in a large network the AUV

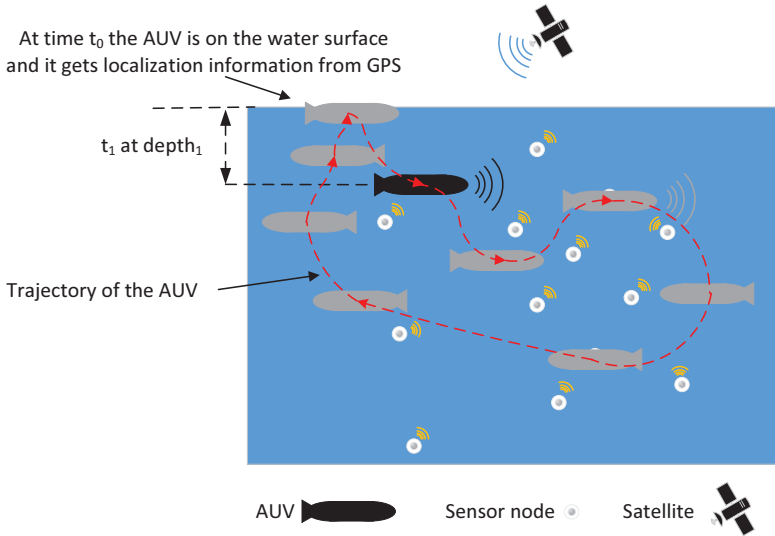


Figure 1.9: Trajectory of the AUV in a network with a multi-stage AUV-aided localization algorithm.

has to sweep a large area and that makes the process time-consuming. To reduce the localization time and improve the localization accuracy, a multi-stage AUV (MS-AUV) aided algorithm is proposed in [63]. In this algorithm, the ordinary nodes listen to the transmitted packets from the AUV. After they have received enough packets, they can localize themselves passively. Finally, after a certain time-out, they all broadcast their location estimates, and therefore the nodes which had not received enough packets from the AUV, obtain this opportunity to localize themselves.

### Underwater positioning scheme (UPS)

In the underwater positioning scheme (UPS) [64], it is assumed that the anchors are within the communication range of each other, and they transmit packets in a predefined order. Every  $T$  seconds the primary anchor transmits a packet, then the second anchor receives this packet and broadcasts its location information, and this continues until all anchors have transmitted their packets. An ordinary node gathers this information and calculates the TDoAs, converts them to distance and performs multilateration to estimate its position. The UPS algorithm is fast, and can be used in asynchronous networks. However, it is not very stable as a

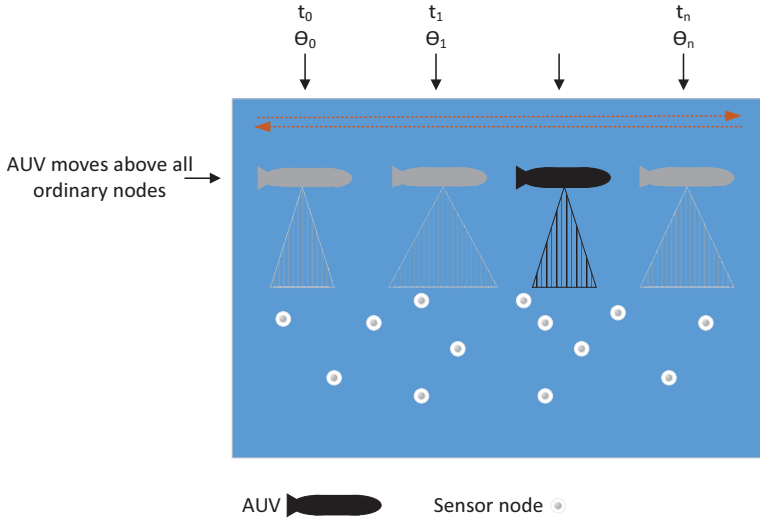


Figure 1.10: The movement of the AUV equipped with a directional beacon in an underwater acoustic sensor network.

packet loss between the anchors stops the localization process. To overcome such a problem other versions of this algorithm, such as the enhanced UPS [65], have been proposed. Furthermore, UPS cannot localize all sensor nodes uniquely in an enclosed area of four anchor nodes. It has been observed that up to 16% of the network is not localizable. As proposed by the Wide Coverage Positioning (WPS) algorithm [66], this issue can be solved by employing five anchors for localization. The drawback of the algorithm is its higher localization delay and more communications in comparison to UPS.

### Localization with directional beacons (LDB)

The Localization with Directional Beacons (LDB) is quite similar to the AUV-aided algorithm. The only difference is that in LDB, the AUV moves above ordinary nodes, and it has a directional transducer with an adjustable angle. When the AUV broadcasts its location, it includes the angle of the transducer in its packet. This information is used by each ordinary node to map the AUV coordinates to the same horizontal plane where the node resides as shown in Fig.1.10.

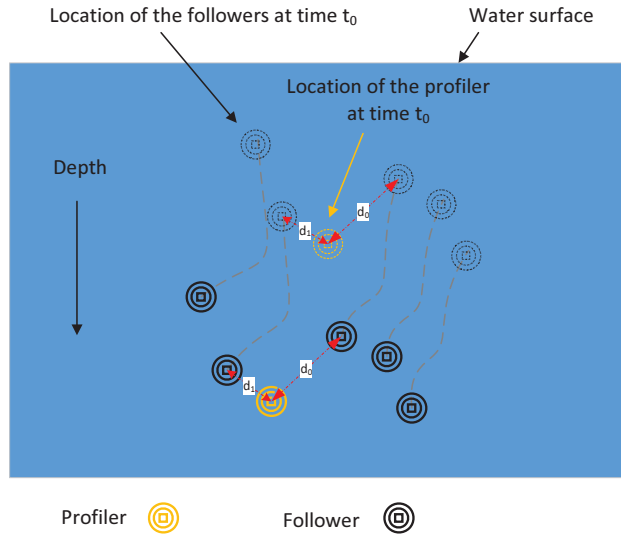


Figure 1.11: Movement of the sensor nodes in a UASN with collaborative localization.

### Collaborative localization (CL)

Proposed in [67], Collaborative Localization (CL) focuses on a mobile UASN application where underwater sensor nodes collect data from the deep ocean and carry them to the surface. The CL uses two types of underwater nodes, profilers and followers. The profilers take the lead in descending underwater, and the followers try to follow the profilers and pass the same trajectory that the profilers have passed. If we assume that the network descends with a constant speed, we have to keep the relative distances of the nodes constant. To fulfill that constraint, a ToF technique is used to position the profilers with respect to the followers as depicted in Fig. 1.11. The two main drawbacks of this network structure is the need for synchronization, and network dependency. If some of the nodes fail to operate, the performance of CL could degrade notably.

## 1.5 Thesis outline and contributions

The thesis is structured in four parts. In this first part consisting of Chapter 1, an introduction on the history, contemporary research, and advances of underwater

communications is given. Furthermore, different categories of localization algorithms, and the metrics used to evaluate these algorithms are listed. Afterwards, the general localization problem is formulated, and the most well-known localization algorithms designed for an underwater medium are reviewed. In this part, we also disclose the main contributions of this thesis.

The second part is composed of three chapters (Chapters 2 to 4), and it mainly focuses on accurate self-localization algorithms in an underwater medium with a variable sound speed profile. A concise introduction, as well as our contributions in this part are discussed in Subsection 1.5.1

The third part explains the problem of localization packet scheduling in two chapters (Chapters 5 and 6). Our contributions from this part are reflected in Subsection 1.5.2. Finally, the fourth part (Chapter 5) of this thesis provides conclusions, and highlights the major results and observations. This part also presents future directions for research in this field, and mentions a couple of ideas as a possible continuation of this thesis.

### **1.5.1 Contributions towards underwater localization algorithms**

In an underwater medium the sound speed is not constant, but varies with depth. This phenomenon upsets the linear dependency of the distance traveled by an acoustic wave to the time it takes for the wave to travel that distance, and therefore it makes existing distance-based localization algorithms less effective in an underwater environment. The problem of underwater localization in an environment with a depth-dependent sound speed profile has been decomposed into three topics where each one is explained individually as a thesis chapter. The list of contributions made in Chapter 2 is presented below.

A simple form of the problem where the sound speed profile (SSP) changes linearly with the depth (isogradient sound speed profile) has been considered.

The problems of localizing a fixed node and tracking a mobile target from acoustic time-of-flight (ToF) measurements in a three-dimensional underwater environment is addressed.

To solve these problems, we first analytically relate the acoustic wave ToF between two nodes to their positions.

After obtaining sufficient ToF measurements, we then adopt the Gauss-Newton algorithm to localize the fixed node in an iterative manner, and we

utilize the extended Kalman filter for tracking the mobile target in a recursive manner.

Although the proposed iterative algorithm does not require any depth information to localize a node, we also analyze the problem when depth information of the nodes is available. This is a valid assumption as most AUVs are equipped with pressure sensors.

In either case (with or without depth information) the Cramér-Rao bound (CRB) for localization and the posterior CRB (PCRB) for tracking have been derived.

In addition, through several simulations, we will illustrate that the proposed algorithms perform superb since they meet their CRB and PCRB.

This chapter has been published as

- $\mathcal{J}_1$  **Hamid Ramezani**, Hadi Jamali-Rad, and Geert Leus. “Target localization and tracking for an isogradient sound speed profile.” *Signal Processing, IEEE Transactions on* 61.6 (2013): 1434-1446.

Part of this chapter and some early results related to this chapter have also appeared in

- $\mathcal{C}_1$  **Hamid Ramezani**, Hadi Jamali-Rad, and Geert Leus. “Localization and tracking of a mobile target for an isogradient sound speed profile.” *Communications (ICC), 2012 IEEE International Conference on.* IEEE, 2012.

Chapter 3 provides the fundamental principles to analyze target localization in an underwater medium with a piece-wise linear SSP (multiple isogradient layers). Here, it is assumed that each sensor node is able to measure its depth and can exchange this information with the other nodes. Under these assumptions, we have shown the following results.

We have demonstrated how the problem of underwater localization can be converted to the traditional range-based terrestrial localization problem when the depth information of the nodes is known a priori.

Similar to Chapter 2, the pair-wise time of flight measurements between the nodes are also formulated as a function of their positions.

In contrast to the single layer scenario (isogradient SSP), it is shown that even without any reflection from the surface or the seabed, the transmitted signal may travel through more than one path between two given nodes. That causes an ambiguity in any localization algorithm.

To simplify the multi-layer analysis, the concept of the ray pattern is introduced in this chapter. A ray pattern is a set consisting of all possible rays that can travel between two points through different layers with a given pattern.

Given the assumption that, at a specific depth, the ToF of the fastest ray is a monotonic function of the horizontal range between the two nodes, a novel iterative ranging algorithm is proposed, and its CRB is extracted.

Extensive numerical results support the superiority of the proposed algorithm over the existing ones.

This chapter has been published as

$\mathcal{J}_1$  **Hamid Ramezani**, and Geert Leus. “Ranging in an underwater medium with multiple isogradient sound speed profile layers.” A special issue of *Sensors; Underwater Sensor Nodes and Underwater Sensor Networks*, Feb. 2012.

Part of this chapter and some early results related to this chapter have also appeared in

$\mathcal{C}_1$  **Hamid Ramezani**, and Geert Leus. “Accurate Ranging in a Stratified Underwater Medium with Multiple Iso-gradient Sound Speed Profile Layers.” *Navigation, Guidance and Control of Underwater Vehicles*. Vol. 3. No. 1. 2012.

In Chapters 2 and 3, it is assumed that the SSP does not vary with time, and it is known accurately by each sensor node. However, in practice, the SSP has to be measured via CTD (conductivity-temperature-depth) sensors or by direct acoustic measurements of the sound speed (sound velocimeter) in a noisy environment. A noisy sound speed measurement will indirectly affect the accuracy of the range estimation, and consequently the performance of the location estimation. In Chapter 4, we consider this problem, and we derive the CRB of range estimation under the assumptions of a depth-dependent sound speed profile wherein (Gaussian) noisy time of flight, depth, and sound speed measurements are

available. The effect of each measurement noise on the CRB of the range estimation is evaluated analytically in this chapter. It is shown that, for long distances, the noise power of the depth measurements does not play a significant role in the CRB, while those of the ToF and the sound speed measurements are dominant. However, the inaccuracy caused by noisy sound speed measurements can be improved as the number of measurements increases. This chapter has been published as

- $\mathcal{C}_1$  **Hamid Ramezani**, Raj Thilak Rajan, and Geert Leus. “Cramer Rao Lower Bound for Underwater Range Estimation with Noisy Sound Speed Profile.” 49th Annual Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, California November 8-11, 2015.

### 1.5.2 Contributions towards underwater packet scheduling for localization

An underwater node is able to estimate its location, if it obtains enough measurements. As explained before, these measurements can be ToFs, TDoAs, RTTs, channel impulse response measurements or any other physical phenomenon related to the position of the node. The following scenarios might be considered to obtain a measurement.

A node obtains a measurement independently via mounted sensors. The depth (pressure sensor), temperature (thermometer), conductivity (salinity sensor), displacement (accelerometers and gyroscopes), geometrical map of the environment (optical camera, sonar, or Doppler velocity log), and direction (compass) are examples of such measurements. Due to various ambiguities an underwater node cannot localize itself through independent measurements accurately; however, it can use them to improve its estimation.

An anchor with a known position broadcasts a known acoustic, optical, magnetic or electrical signal to the sensor nodes. Underwater nodes record the signal via an appropriate receiver, and extract position-based or distance-based features such as ToF, RSS, CIS and so on. Having gathered enough measurements of this type, a node would be able to estimate its position. This scenario can be used for UASNs with fixed-located anchor nodes.

An anchor broadcasts a packet which contains not only a known signal, but also information about its position and time of transmission. The underwater

nodes now extract the useful features as well as the anchors' information. Since the position information of the anchors is included in the packet, the anchors can freely move in the operating environment. This approach is accepted for UASNs since the nodes drift with water currents, waves and wind.

Although a great deal of research exists on underwater localization algorithms, little work has been done to determine how the anchors should transmit their packets to the sensor nodes, especially for UASNs where the propagation delay is very large. In this part, we tackle this problem in two chapters (Chapters 5 and 6).

Chapter 5 concerns the problem of designing a medium access control protocol for a UASN which efficiently schedules the localization packets of the anchors. Considering the fact that acoustic packet exchanging suffers from long propagation delays, the protocol utilizes the relative positions of the anchors and their maximum transmission range, and schedules the packet transmissions from the anchors with the goal of minimizing the duration of the localization task. The contributions of this chapter are as follows.

We formulate the problem of minimizing the duration of the localization task in partially connected networks.

An anchor is usually not interested in the transmitted packets from the other anchors as it knows or can measure its position. However, if the anchors want to estimate a physical phenomenon, or obtain another kind of information encapsulated in the localization packets, then each anchor has to broadcast its data to all nodes of the network. We show how the scheduling problem can be modified to support such a broadcast scenario.

We further introduce the idea of dynamic multi-channel packet scheduling, in which the system is able to split the existing channel into a few subchannels, and to schedule the localization packets not only in time but also in a specific subchannel. This algorithm is independent of the technique (i.e., FDMA, CDMA, etc.) that is used to increase the number of channels as long as the created sub-channels are orthogonal to each other.

We also consider several practical challenges (such as multi-path, sound speed variations, interference and moving anchors) in the implementation of the proposed algorithm for an actual network set-up. We show how the formulation can be adapted accordingly.

Moreover, we show that the optimization problem can be modeled as an NP-hard combinatorial problem. The optimal solution of such a problem can be obtained through exhaustive search among all possible solutions. The complexity of the exhaustive search method grows exponentially as the number of anchors gets larger. Nonetheless, it can be done for a network with a few anchors.

Finally, two low-complexity approximate algorithms are proposed, and their performance is evaluated through comprehensive simulations and comparisons with the optimal solution as well as with other existing methods. Numerical results show that the proposed algorithms perform near optimum and can be considered as good candidates for real UASNs.

This chapter has been published as

- $\mathcal{J}_1$  **Hamid Ramezani**, and Geert Leus. “Localization Packet Scheduling for Underwater Acoustic Sensor networks.” IEEE Journal on Selected Areas in Communications, vol. 33 , no. 7, May 2015.

Part of this chapter and some early results related to this chapter have also appeared in

- $\mathcal{C}_3$  **Hamid Ramezani**, and Geert Leus. Dynamic Multi-Channel Packet Scheduling in an Underwater Acoustic Sensor Network, accepted in Proc. of the Asilomar Conference on Signals, systems, and Computers (Asilomar 2013), Pacific Grove, California, USA, November 2013.
- $\mathcal{C}_2$  **Hamid Ramezani**, and Geert Leus. DMC-MAC: Dynamic Multi-channel MAC in Underwater Acoustic Networks, accepted in the European Signal Processing Conference (EUSIPCO’13), Marrakech, Morocco.
- $\mathcal{C}_1$  **Hamid Ramezani**, and Geert Leus. L-MAC: Localization packet scheduling for an underwater acoustic sensor network, accepted in Proc. of the International Conference on Communications (ICC2013), Budapest, Hungary.

Chapter 6 considers the joint problem of packet scheduling and self-localization in an underwater acoustic sensor network where the nodes are distributed at random with a given probability distribution function. As it is known, a shorter localization time allows for a more dynamic network, and leads to a better network efficiency in terms of throughput. Therefore, one goal of the optimization problem in this

chapter is to minimize the localization time through an optimum packet scheduling. Another goal is to minimize the error between the actual nodes' positions and their estimates. The contributions of this chapter are mentioned below.

Two packet transmission schemes are proposed, namely a collision-free scheme (CFS), and a collision-tolerant scheme (CTS). The required localization time is formulated for these schemes, and through analytical results and numerical examples their performance is shown to depend on the circumstances.

Considering packet loss and collisions, the localization time is formulated for each scheme. The minimum required localization time for each type of scheduling (CFS and CTS) is obtained analytically for a predetermined probability of successful localization for each sensor node.

In a UASN, less anchors (from the ones available in the network) can be used for the localization task. The number of anchors play an important role in the localization coverage. We show how the minimum number of anchors can be determined in order to reach the desired probability of self-localization.

Furthermore, an iterative Gauss-Newton self-localization algorithm is proposed for a sensor node which experiences packet loss or collision. An anchor can transmit a packet multiple times (time-diversity), and a sensor might receive none, one or many of them. We will show how the problem of self-localization can be addressed under such conditions.

The Cramér-Rao lower bound (CRB) of localization is derived for each scheme. In this derivation, the distance dependent signal to noise ratio, and the effects of packet loss due to fading, shadowing, and collisions, are considered.

We show that in cases where the packet duration is short (as is the case for a localization packet), the operating area is large (above 3km in at least one dimension), and the average probability of packet-loss is not close to zero, the collision-tolerant scheme is found to require a shorter localization time. At the same time, its implementation complexity is lower than that of the collision-free scheme, because in CTS, the anchors work independently. CTS consumes slightly more energy to make up for packet collisions, but it is shown to provide a better localization accuracy.

This chapter has been published as

- $\mathcal{J}_1$  **Hamid Ramezani**, Fatemeh Fazel, Milica Stojanovic, and Geert Leus. “Collision Tolerant and Collision Free Packet Scheduling for Underwater Acoustic Localization.” *IEEE Transactions on Wireless Communications*, vol. 14, no. 5, pp. 2584-2595, Jan 2015.

Part of this chapter and some early results related to this chapter have also appeared in

- $\mathcal{C}_1$  **Hamid Ramezani**, Fatemeh Fazel, Milica Stojanovic, and Geert Leus. Packet Scheduling for Underwater Acoustic Sensor Network Localization., accepted in the Proceeding of IEEE ICC 2014 Workshop on Advances in Network Localization and Navigation (ANLN), 10-14 June 2014, Sydney, Australia.



---

---

## **PART II**

---

# **UNDERWATER LOCALIZATION IN INHOMOGENEOUS MEDIUM**



# Chapter 2

## Target Localization and Tracking for an Isogradient Sound Speed Profile

Hamid Ramezani<sup>1</sup>

---

<sup>1</sup>This chapter is a verbatim copy of the journal paper published as: Hamid Ramezani, Hadi Jamali-Rad, and Geert Leus. “Target localization and tracking for an isogradient sound speed profile.” *Signal Processing, IEEE Transactions on* 61.6 (2013): 1434-1446.

## Abstract

In an underwater medium the sound speed is not constant, but varies with depth. This phenomenon upsets the linear dependency of the distance traveled by an acoustic wave to the time it takes for the wave to travel that distance, and therefore makes existing distance-based localization algorithms less effective in an underwater environment. This paper addresses the problems of localizing a fixed node and tracking a mobile target from acoustic time-of-flight (ToF) measurements in a three-dimensional underwater environment with an isogradient sound speed profile. To solve these problems we first analytically relate the acoustic wave ToF between two nodes to their positions. After obtaining sufficient ToF measurements, we then adopt the Gauss-Newton algorithm to localize the fixed node in an iterative manner, and we utilize the extended Kalman filter for tracking the mobile target in a recursive manner. Through several simulations, we will illustrate that the proposed algorithms perform superb since they meet the Cramér-Rao bound (CRB) for localization and posterior CRB for tracking.

## 2.1 Introduction

A wide variety of applications including early warning systems for natural disasters (e.g., tsunamis), ecosystem monitoring, oil drilling and military surveillance are the main driving force behind exploring underwater environments [55]. Recent advances in the design of wireless sensor networks (WSNs) motivated system designers to exploit underwater acoustic sensor networks (UASNs) for data gathering and ocean explorations. In order to interpret the sensed data in a meaningful manner, we require the sensor positions either remotely or locally as in terrestrial WSNs. Very low bit rate, low link quality, multi-path, time variability, and a depth-dependent sound speed profile (SSP) are the most important characteristics that make underwater acoustic communications a challenging field of research [68]. The aforementioned challenging characteristics, therefore, necessitate the design and development of new localization and tracking algorithms. A complete survey of techniques and challenges in underwater localization can be found in [69], [49]. In [70], the authors propose a centralized algorithm to overcome the severe multi-path property of the underwater environment due to scattering from the seabed and ocean surface. In [71], a time-difference-of-arrival-based localization scheme for stationary UASNs is proposed which does not require time synchronization among network nodes. In [72], depth information as well as range

measurements are used to localize a target node inside a three-dimensional (3-D) area.

As stated before, one of the underwater localization challenges is the depth-dependent SSP which varies with temperature, pressure, and salinity [73]. Due to this property, an acoustic ray does not propagate along a straight line, but it bends. Even if the nodes are located at the same depth the distance between the two nodes in an underwater environment is not linearly proportional to the wave travel time. However, in all the above mentioned underwater localization schemes, the propagation sound speed is assumed constant, and thus the trajectory of the ray will be a straight line. However, this assumption is unrealistic in general and degrades the performance of underwater localization algorithms. In contrast to the aforementioned algorithms, [74] evaluates the localization performance degradation of the straight-line propagation model compared to the real propagation model. As the target node measures the time-of-flight (ToF) from an anchor node, the corresponding constant range interval surface for this measured ToF is constructed. To construct such a constant range interval surface (or a curve in a 2-D medium), the path trajectory for each departing ray from the considered anchor node is calculated. Then, on each path trajectory a point is selected related to the ToF. All these points together yield the desired constant range interval surface. After sufficient ToF measurements are collected, the position of the target is estimated as the point whose sum of squared distances from all these surfaces is minimum. The main drawback of this approach is the computational complexity which depends on the network size and the required accuracy. In [75], it is stated that in an underwater medium with an isogradient SSP the path trajectory becomes an arc of a circle. Nonetheless, non-straight-line wave propagation is neglected in [75]. Since the recovery of missing links is the main goal of [75], the positioning error is basically dominated by the error due to missing links. The authors of [5] consider a real wave propagation model for UASNs localization based on the depth information and SSP. They eliminate the underwater range computation by using a look up table (LUT), which relates the travel time information to the horizontal distance between two nodes. Their proposed algorithm is very fast, but to scan the whole environment a huge LUT is required which may not be practical. Furthermore, the SSP in an underwater medium is subject to changes in temperature and conductivity, and any change in SSP degrades the LUT accuracy and therefore affects the localization performance. Finally, [6] considers the problem of ranging in an underwater environment. In that paper, a numerical range estimator is proposed which is based on reconstructing the slanted path using Fermat's principle

and calculus of variations. Basically, after depth and time measurements are taken, an integral equality is formed which is taken over the depth between the nodes. Then, the constant defined by Snell's law is numerically calculated, which is used to compute the horizontal distance between the nodes through another integral equality. The work of [6] is really comprehensive, since with any given SSP, the horizontal distance is computable. However, the algorithm may compute the constant (defined by Snell's law) with an ambiguity, since in an underwater medium it is common that a traveling ray from one node to another passes a given depth more than once. As the depth of a node on a traveling ray is not a monotonic function of the depth, this phenomenon yields an ambiguous value for any integral taken with respect to (w.r.t.) the depth along the traveling path.

In this work, we propose a UASN localization and tracking approach for an underwater medium with an isogradient SSP. The isogradient SSP is a good assumption for deep water environments [2], [76], since the conductivity and water temperature in a deep underwater medium are constant, and the only factor that affects the SSP is the pressure which linearly depends on the depth. Notably, the measured SSP in a deep underwater medium is more accurate than the measured SSP in shallow waters [77]. In order to find the location of a target, we analytically relate the position of that node to the ToFs. Using at least four ToF measurements from four anchors, we formulate the localization problem. It will be shown that the ToF measurements in an underwater medium are a non-linear function of the target position, and consequently the localization problem is categorized as a non-linear least squares problem. The analytical relationship between the ToFs and the nodes' positions also allows us to compute the derivatives of the ToFs with respect to the target's position in closed form, and hence enables us to utilize efficient methods to solve the non-linear least squares localization problem, such as the Gauss-Newton method, the Levenberg-Marquardt method, the Powell's Dog Leg method, and so on. The Gauss-Newton algorithm (GNA) is the basis of many efficient methods for solving non-linear least squares problems, and in this paper we use this algorithm for estimating the target's position. In addition, since tracking is also important, we perform multilateration recursively by using the extended Kalman filter (EKF). Although other tracking methods could be adopted, we select the EKF because of the availability of the derivative of the measurements w.r.t the location variables.

We do not require any depth information in our algorithms, and we directly work with ToF measurements based on a given SSP. However, since some autonomous acoustic vehicles are equipped with pressure sensors [76], [78], we also investigate the existence of depth information in our algorithms. To the best of

our knowledge, this is the first work that analytically solves the problem of accurate localization and tracking in an isogradient SSP underwater environment with only ToF information.

The rest of the paper is organized as follows. In Section 2.2, we analyze the characteristics of a ray traveling between two points, and also explain how the positions of the two nodes are related to the ToF. In Section 2.3, the static localization algorithm is introduced, and its corresponding Cramér-Rao bound (CRB) is derived. We analyze the problem of mobile target tracking in Section 2.4, where we calculate the posterior CRB (PCRB). We evaluate the performance of the proposed algorithms in Section 2.5 through several simulations, and finally conclude the paper in Section 2.6.

## 2.2 Ray tracing between two points

We consider the problem of tracing a ray between two nodes, e.g., A (anchor) and T (target), in a 3-D environment with an isogradient sound speed where the SSP is only dependent on the depth, and has the following form

$$C(z) = b + az, \quad (2.1)$$

where  $z$  denotes the depth,  $b$  indicates the sound speed at the surface, and  $a$  is a constant depending on the environment. Without loss of generality, to solve the ray tracing problem between the two nodes, we assume that the  $z$  axis crosses node A. Therefore, due to the cylindrical symmetry around the  $z$  axis we can transfer the ray tracing problem to the plane which includes both nodes and the  $z$  axis as shown in Fig. 2.1. In this figure,  $r^T - r^A$  represents the horizontal distance between the nodes, and it can be written as

$$r^T - r^A = \sqrt{(x^T - x^A)^2 + (y^T - y^A)^2}, \quad (2.2)$$

where  $x^T$ ,  $y^T$ ,  $x^A$ , and  $y^A$  indicate the  $x$ -coordinate and  $y$ -coordinate of respectively point T and point A in a 3-D environment. Since the  $z$ -axis is assumed to cross point A, we actually have  $r^A = 0$  but we keep it in our formulation for representation purposes.

Acoustic propagation is usually modeled using a ray tracing approach which is a valid approximation for the aforementioned isogradient SSP underwater environment. Ray tracing is guided by Snell's law given by [5]

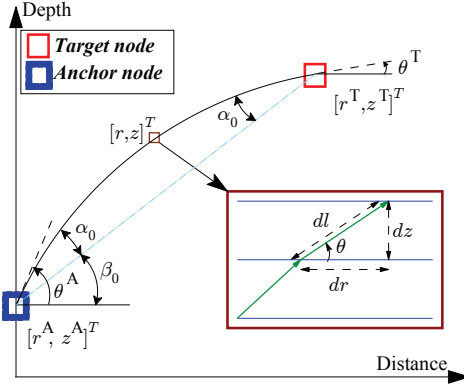


Figure 2.1: Description of a ray between a target node and an anchor node.

$$\frac{\cos \theta}{C(z)} = \frac{\cos \theta^A}{C(z^A)} = \frac{\cos \theta^T}{C(z^T)} = k_0, \text{ and } \theta \in \left[ -\frac{\pi}{2}, \frac{\pi}{2} \right], \quad (2.3)$$

where  $\theta^T$  and  $\theta^A$  are the ray angles at the target node and anchor node locations, respectively, as illustrated in Fig. 2.1.  $z^A$  and  $z^T$  represent the depth of the anchor node and the target node, respectively, and  $k_0$  is constant along a ray traveling between the nodes. Moreover, the parameters  $\theta$  and  $z$  represent the angle and depth of a given point along the ray. From Fig. 2.1, we can write

$$\partial r = \frac{\partial z}{\tan \theta}, \quad (2.4a)$$

$$\partial l = \frac{\partial z}{\sin \theta}, \quad (2.4b)$$

$$\partial t = \frac{\partial l}{C(z)}, \quad (2.4c)$$

where  $l$  is the arc length of a ray traveling between the two nodes, and  $t$  is its corresponding travel time. Moreover, using (2.1) and (2.3), and by taking derivatives w.r.t.  $z$  and  $\theta$ , we can write

$$\partial z = -\frac{1}{ak_0} \sin \theta \partial \theta. \quad (2.5)$$

In the following subsections, we show how the above partial derivatives can be used for extracting the ray characteristics.

### 2.2.1 Time of flight vs. sensor node locations

In this part of the paper, it is shown how the ToF between the two nodes is related to their positions. By substituting (2.5) into (2.4a), and integrating w.r.t.  $\theta$  we have

$$r^T - r^A = -\frac{1}{ak_0} (\sin \theta^T - \sin \theta^A), \quad (2.6)$$

for the horizontal distance, and for the vertical distance between the two nodes we can write

$$z^T - z^A = \frac{1}{ak_0} (\cos \theta^T - \cos \theta^A). \quad (2.7)$$

Dividing (2.7) by (2.6), considering  $r^T \neq r^A$  we end up with

$$\frac{z^T - z^A}{r^T - r^A} = -\frac{\cos \theta^T - \cos \theta^A}{\sin \theta^T - \sin \theta^A}, \text{ for } r^T \neq r^A. \quad (2.8)$$

Furthermore, by substituting (2.1) into (2.3) we can write one more equality as

$$\frac{b + az^T}{b + az^A} = \frac{\cos \theta^T}{\cos \theta^A}. \quad (2.9)$$

By applying the change of variables  $\theta^A = \beta_0 + \alpha_0$ , and  $\theta^T = \beta_0 - \alpha_0$ , (2.8) and (2.9) can be modified to

$$\frac{z^T - z^A}{r^T - r^A} = \tan \beta_0, \text{ for } r^T \neq r^A, \quad (2.10)$$

$$\frac{b + az^T}{b + az^A} = \frac{1 - \tan \beta_0 \tan \alpha_0}{1 + \tan \beta_0 \tan \alpha_0}. \quad (2.11)$$

The parameter  $\beta_0$  denotes the angle of the straight line between the two nodes w.r.t. the horizontal axis, and  $\alpha_0$  represents the angle at which the ray trajectory deviates from this straight line as shown in Fig. 2.1. For the exceptional condition where  $z^A = z^T$ , (2.11) is not informative and should be modified to

$$\tan \alpha_0 = \frac{1}{2} \frac{a(r^T - r^A)}{b + az^T}, \text{ for } z^T = z^A, \quad (2.12)$$

which is extracted from (2.6). Now, by integrating (2.4c) w.r.t.  $\theta$ , the ToF can be calculated as

$$t = -\frac{1}{a} \left( \ln \frac{1 + \sin \theta^T}{\cos \theta^T} - \ln \frac{1 + \sin \theta^A}{\cos \theta^A} \right). \quad (2.13)$$

In the above equation for the special case where  $r^A = r^T$ , one node is located on top of the other node, and thus according to the Snell's law we have  $\theta = \pm \frac{\pi}{2}$ , or  $\partial s = \pm \partial z$ . In this exceptional case, the ToF can be given by

$$t = \begin{cases} -\frac{1}{a} \ln \frac{C(z^T)}{C(z^A)} & \text{for } z^T < z^A \\ -\frac{1}{a} \ln \frac{C(z^A)}{C(z^T)} & \text{for } z^T > z^A \end{cases}. \quad (2.14)$$

Since the occurrence probability of one node being located on top of the other is zero, we ignore it in the rest of this paper.

Up to now, the ToF for an isogradient SSP can be computed using (2.13) by first calculating  $\beta_0$  from (2.10), substituting it into (2.11) and computing  $\alpha_0$ , and consequently  $\theta^A$  and  $\theta^T$ . Since we will adopt the GNA for the static localization and the EKF for tracking a mobile target, in addition to the ToF as a function of the node locations, we also need the derivatives of the ToF w.r.t. the target location. Here, we assume that point A represents a fixed anchor node and point T represents the target node which can be fixed or mobile. To derive  $\frac{\partial t}{\partial r^T}$  and  $\frac{\partial t}{\partial z^T}$  using (2.13) we take the following partial derivatives

$$\frac{\partial t}{\partial r^T} = -\frac{1}{a} \left( \frac{1}{\cos \theta^T} \frac{\partial \theta^T}{\partial r^T} - \frac{1}{\cos \theta^A} \frac{\partial \theta^A}{\partial r^T} \right), \quad (2.15a)$$

$$\frac{\partial t}{\partial z^T} = -\frac{1}{a} \left( \frac{1}{\cos \theta^T} \frac{\partial \theta^T}{\partial z^T} - \frac{1}{\cos \theta^A} \frac{\partial \theta^A}{\partial z^T} \right). \quad (2.15b)$$

The above equations depend on the partial derivatives of the ray angles at the target and anchor location. These partial derivatives can be computed from (2.8) and (2.9) as

$$\frac{\partial \theta^T}{\partial r^T} + \frac{\partial \theta^A}{\partial r^T} = -\frac{z^T - z^A}{(r^T - r^A)^2} \frac{(\sin \theta^T - \sin \theta^A)^2}{1 - \cos(\theta^T - \theta^A)}, \quad (2.16a)$$

$$\frac{\partial \theta^T}{\partial r^T} - \frac{b + az^T}{b + az^A} \frac{\sin \theta^A}{\sin \theta^T} \frac{\partial \theta^A}{\partial r^T} = 0, \quad (2.16b)$$

$$\frac{\partial \theta^T}{\partial z^T} + \frac{\partial \theta^A}{\partial z^T} = \frac{1}{r^T - r^A} \frac{(\sin \theta^T - \sin \theta^A)^2}{1 - \cos(\theta^T - \theta^A)}, \quad (2.17a)$$

$$\frac{\partial \theta^T}{\partial z^T} - \frac{b + az^T}{b + az^A} \frac{\sin \theta^A}{\sin \theta^T} \frac{\partial \theta^A}{\partial z^T} = -\frac{a}{b + az^A} \frac{\cos \theta^A}{\sin \theta^T}, \quad (2.17b)$$

where (2.16a) and (2.17a) are calculated from (2.8), and (2.16b) and (2.17b) are derived from (2.9). Observe that (2.16) and (2.17) are linear in  $\frac{\partial \theta^T}{\partial z^T}$ ,  $\frac{\partial \theta^A}{\partial z^T}$ ,  $\frac{\partial \theta^T}{\partial r^T}$ ,  $\frac{\partial \theta^A}{\partial r^T}$ , and can thus simply be solved in closed form. By computing these values for each anchor and substituting them into (2.15a) and (2.15b), we are able to compute the derivative of all measured ToFs w.r.t. the target node position. Finally,  $\frac{\partial t}{\partial x^T}$  and  $\frac{\partial t}{\partial y^T}$  can be derived as

$$\frac{\partial t}{\partial x^T} = \frac{\partial t}{\partial r^T} \frac{x^T - x^A}{r^T - r^A}, \quad (2.18a)$$

$$\frac{\partial t}{\partial y^T} = \frac{\partial t}{\partial r^T} \frac{y^T - y^A}{r^T - r^A}. \quad (2.18b)$$

### 2.2.2 Traveled ray length

As stated before, in an underwater medium, the traveled ray length between two points is not the same as the distance between them. In the next section, we will see that the distance-dependent noise is related to the received signal power, and consequently to the ray length. The ray length in an underwater medium with an isogradient SSP can be easily obtained by substituting (2.5) into (2.4b), and taking an integral w.r.t.  $\theta$ , leading to

$$l = -(az^T + b) \frac{\theta^T - \theta^A}{a \cos \theta^T}, \quad (2.19)$$

where  $l$  is the traveled ray length between the nodes A and T.

Further, we will observe later on that in order to extract a lower bound on the position estimation variance, the partial derivatives of the traveled ray length w.r.t. the target location are needed. Below, we compute the derivative of the ray length w.r.t.  $r^T$  and  $z^T$  as a function of  $\frac{\partial \theta^T}{\partial r^T}$ ,  $\frac{\partial \theta^T}{\partial z^T}$ ,  $\frac{\partial \theta^A}{\partial r^T}$ ,  $\frac{\partial \theta^A}{\partial z^T}$ :

$$\frac{\partial l}{\partial r^T} = -\frac{az^T + b}{a \cos \theta^T} \left[ (1 + (\theta^T - \theta^A) \tan \theta^T) \frac{\partial \theta^T}{\partial r^T} - \frac{\partial \theta^A}{\partial r^T} \right] \quad (2.20a)$$

$$\begin{aligned} \frac{\partial l}{\partial z^T} &= -\frac{az^T + b}{a \cos \theta^T} \left[ (1 + (\theta^T - \theta^A) \tan \theta^T) \frac{\partial \theta^T}{\partial z^T} - \frac{\partial \theta^A}{\partial z^T} \right] \\ &\quad - \frac{\theta^T - \theta^A}{\cos \theta^T}, \end{aligned} \quad (2.20b)$$

where  $\frac{\partial l}{\partial x^T}$  and  $\frac{\partial l}{\partial y^T}$  can simply be calculated from  $\frac{\partial l}{\partial r^T}$ . The partial derivatives  $\frac{\partial l}{\partial x^T}$  and  $\frac{\partial l}{\partial y^T}$  can be obtained from  $\frac{\partial l}{\partial r^T}$  similar to the computation of the partial derivatives of the ToF w.r.t.  $x^T$  and  $y^T$ .

### 2.2.3 Ray depth overshoot

In practice, the SSP of the entire underwater medium cannot be considered isogradient. However, the SSP can be modeled as isogradient within a certain depth range. In other words, the ocean environment can be divided into several isogradient SSP layers with various thicknesses. For instance in [2], it is shown that the SSP of the Pacific ocean from a depth of 600 m to a depth of 5000 m can be estimated as isogradient.

In an underwater environment with an isogradient SSP, it is probable that the depth of a node along a given ray between two points, say A and T, exceeds the region  $[z^A, z^T]$ . The depth of a node along a given ray can be expressed as a function of the ray angle as

$$z(\theta) = \frac{1}{a} \left( \frac{\cos \theta}{k_0} - b \right), \quad (2.21)$$

where  $k_0$  is a positive constant defined earlier in (2.3). It is obvious that  $z(\theta)$  follows the behavior of  $\cos \theta$ , and its extremum occurs at the maximum of  $\cos \theta$ , i.e.,  $\theta = 0$ , since when  $a > 0$  the ray bends towards the deeper regions whereas when  $a < 0$  the ray bends upwards, i.e., to smaller depths. In other words, the depth of a node on a ray exceeds the region  $[z^A, z^T]$  if and only if the sign of the ray angle at the two points differs from each other. Thereby, when we have  $\theta^A \theta^T < 0$ , the value of the minimum or the maximum depth can be computed as

$$\begin{aligned} \max_{\theta} z(\theta) &= 1/a(1/k_0 - b) \text{ if } a > 0 \\ \min_{\theta} z(\theta) &= 1/a(1/k_0 - b) \text{ if } a < 0 \end{aligned} \quad (2.22)$$

If the computed  $\max_{\theta} z(\theta)$  or  $\min_{\theta} z(\theta)$  lies within the boundaries of the isogradient SSP layer, then the formulas derived in the previous subsections are valid.

### 2.2.4 Range approximation using depth information

The underwater nodes can also be equipped with a pressure sensor, which allows them to estimate their depth. Using this depth information as well as (2.8) to (2.13), the target node can compute its horizontal distance from each anchor, and use the traditional range-based WSN localization algorithms to find its position [79]. However, due to the computational complexity, it is sometimes preferable to approximate the underwater medium as a homogeneous one. Below, we will show

that if the depth of the two nodes is known, the underwater environment can be approximated as a homogeneous one using the assumption of a straight-line wave propagation. By adopting such an approximation, the computational complexity of the localization scheme decreases, but its accuracy degrades.

With the assumption of a straight-line wave propagation, the ToF between two points can be written as

$$t_{sl} = \int_{z^A}^{z^T} \frac{1}{\sin(\beta_0) C(z)} dz = -\frac{1}{a \sin(\beta_0)} \ln \left( \frac{v^T}{v^A} \right), \quad (2.23)$$

where  $\beta_0$  is the angle of the straight line between the two nodes w.r.t. the horizontal axis, as defined in (2.8). Hence, the average sound speed is

$$\bar{v}_{sl} = \frac{\|\mathbf{x}^T - \mathbf{x}^A\|}{t_{sl}} = \frac{v^T - v^A}{\ln(\frac{v^T}{v^A})}. \quad (2.24)$$

where  $\|\mathbf{x}^T - \mathbf{x}^A\|$  is the distance between the two points which is related to the depth of the nodes as

$$\|\mathbf{x}^T - \mathbf{x}^A\| = \frac{z^T - z^A}{\sin \beta_0}. \quad (2.25)$$

It can be seen that the average sound speed, under this assumption, only depends on the sound speed at the depths where the nodes are located. Moreover, it can be shown that based on the depth information of the nodes, (2.24) is the best linear approximation of the sound speed in an isogradient SSP medium. Regardless of the availability of the depth information, the distance error originating from the assumption of a constant wave propagation speed can be acquired as

$$E_r = t\bar{v} - \|\mathbf{x}^T - \mathbf{x}^A\|, \quad (2.26)$$

where  $t$  is the actual ToF between the two nodes, and  $\bar{v}$  is the assumed constant sound speed. This also holds for the case of a straight line propagation based on availability of depth information, i.e.,  $\bar{v} = \bar{v}_{sl}$ .

## 2.3 Target localization based on time of flight measurements

In this paper, we consider a single target node whose position will be determined by a number of anchor nodes. However, the extension to multiple target nodes

is easy. Two scenarios for multiple target localization can be considered. In a first scenario, the anchor nodes are transmitters and the target nodes are receivers. Then, each target node can measure the ToFs to the anchors individually and estimate its position. In a second scenario, the target nodes are transmitters and the anchors are receivers. Under this condition, if the target nodes send the ranging signal simultaneously, then there would be lots of uncertainties and ambiguities for the localization algorithm. Nevertheless, if we assume that each target transmits its ranging signal while the others are silent (for instance as in a TDMA scheme), then there will be no ambiguity and the proposed algorithm can be extended to a multiple target scenario.

In real scenarios we may only know that the sound speed varies linearly with depth,  $C(z) = az + b$ , but we do not know the value of  $a$  and  $b$ . In addition, the characteristics of the environment may change slowly with time, due to the water temperature and salinity, and consequently the values of  $a$  and  $b$  may change. Since we know the anchor positions, we can estimate the value of  $a$  and  $b$  by a simple training phase. For instance, the  $i$ -th anchor transmits a signal to the  $j$ -th anchor who can then compute the ToF. Repeating this procedure for all combinations of two anchors, we have  $N(N - 1)/2$  ToF measurements, and based on equations (2.10), (2.11) and (2.13) we are able to estimate the values of  $a$  and  $b$ . In this way, the algorithm can cope with slow variations of the environment.

To be able to measure the ToFs between the target node and the anchors, the target node needs to be synchronized with the anchor nodes of the network. Quite contrary to terrestrial WSNs, synchronizing a UASN is a difficult task. Large propagation delays and possible node movements are two significant attributes that severely affect UASN synchronization [80]. To eliminate this problem, a ping-pong style scheme to measure the round-trip delay between the target node and each of the anchor nodes can be employed [81]. However, in this paper we assume that all the nodes are synchronized, and we only focus on the error that results from the assumption of a straight-line propagation.

### 2.3.1 Static network model

We consider a 3-D underwater wireless sensor network consisting of  $N \geq 4$  anchor nodes with known locations and one fixed target node. The ToF measurements are assumed to be affected by Gaussian distributed noise as

$$\tilde{\mathbf{t}} = \mathbf{f}(\mathbf{x}) + \mathbf{v}, \quad (2.27)$$

where  $\mathbf{f}(\cdot) = [f_1(\cdot), f_2(\cdot), \dots, f_N(\cdot)]^T$  is a function relating the actual ToFs to the target location  $\mathbf{x} = [x, y, z]^T$  (we omit the superscript T for simplicity),  $\tilde{\mathbf{t}} = [\tilde{t}_1, \tilde{t}_2, \dots, \tilde{t}_N]^T$  is a vector containing the ToF measurements between the target node and each of the anchor nodes, and  $\mathbf{v}$  represents the measurement noise. We assume that the noise components are mutually independent, and hence the covariance matrix of the noise vector can be obtained as

$$\mathbf{R}_v = \text{diag}(\sigma_1^2, \sigma_2^2, \dots, \sigma_N^2), \quad (2.28)$$

where  $\sigma_n^2$ , for  $n \in \{1, 2, \dots, N\}$ , is the noise variance of the ToF measurement based on the  $n$ -th anchor node. Since the measurement errors in the ToFs are mutually uncorrelated, the maximum likelihood (ML) solution for  $\mathbf{x} = [x, y, z]^T$  will be given by

$$\arg \min_{\mathbf{x}} \|\mathbf{f}(\mathbf{x}) - \tilde{\mathbf{t}}\|^2. \quad (2.29)$$

### 2.3.2 Proposed positioning algorithm

The optimization problem in (2.29) is non-linear w.r.t. the variable  $\mathbf{x}$ , and therefore it is difficult to be solved analytically. Here, we adopt a numerical system solver such as the GNA. The algorithm starts with an initial point and improves the estimate recursively as stated in Algorithm 1.

---

#### Algorithm 1 Gauss-Newton Algorithm

---

Start with an initial location guess.

Set  $k = 1$  and put a large value in  $E$ .

**while**  $k \leq K$  and  $E \geq \epsilon$  **do**

    Next state:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \left( \nabla \mathbf{f}(\mathbf{x}^{(k)})^T \nabla \mathbf{f}(\mathbf{x}^{(k)}) \right)^{-1} \nabla \mathbf{f}(\mathbf{x}^{(k)})^T (\mathbf{f}(\mathbf{x}^{(k)}) - \tilde{\mathbf{t}})$$

$$E = \|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|$$

$$k = k + 1$$

**end while**

$$\hat{\mathbf{x}} = \mathbf{x}^{(k)}$$


---

In this algorithm,  $\nabla \mathbf{f}(\mathbf{x}^{(k)}) = \left[ \frac{\partial f_1}{\partial \mathbf{x}}, \frac{\partial f_2}{\partial \mathbf{x}}, \dots, \frac{\partial f_N}{\partial \mathbf{x}} \right]_{\mathbf{x}=\mathbf{x}^{(k)}}^T$  represents the gradient of the vector  $\mathbf{f}$  w.r.t. the variable  $\mathbf{x}$  at  $\mathbf{x}^{(k)}$ , where  $\mathbf{x}^{(k)}$  is the estimate at the  $k$ -th iteration, which can be computed using (2.15) and (2.18), and  $\frac{\partial f_i}{\partial \mathbf{x}} = \left[ \frac{\partial f_i}{\partial x}, \frac{\partial f_i}{\partial y}, \frac{\partial f_i}{\partial z} \right]^T$  for  $i = 1$  to  $N$ . Here,  $K$  and  $\epsilon$  are the user-defined limits on

the stopping criteria that determine when the algorithm exits from the loop. The parameter  $K$  denotes the total number of iterations, which depends on the required precision. In general, only a small  $K$  is required, i.e.,  $K = 7$  or even less.

In terms of computational complexity, Algorithm 1 indicates that each iteration requires two simple matrix multiplications, namely one  $(.)_{3 \times N}$  by  $(.)_{N \times 3}$  multiplication (i.e., roughly  $9N$  floating operations) as well as one  $(.)_{3 \times 3}$  by  $(.)_{3 \times N}$  multiplication (again roughly  $9N$  floating operations), and one  $3 \times 3$  matrix inversion (i.e., 27 floating operations). Furthermore, we also have to compute the elements of one  $N \times 3$  matrix,  $\nabla \mathbf{f}(\mathbf{x}^{(\cdot)})$ , and one  $N \times 1$  vector,  $\mathbf{f}(\mathbf{x}^{(\cdot)})$ , which are of order  $N$ . This means that in total we have a complexity of order  $N$  for each iteration.

### 2.3.3 Cramér-Rao bound

The Cramér Rao bound (CRB) expresses a lower bound on the variance of any unbiased estimator of a deterministic parameter. In this subsection, we derive two CRBs for two different noise characteristics; distance-independent noise (DIN), for which the variance of the measurement noise is independent of the distance between the nodes, and distance-dependent noise (DDN), for which the variance of the measurement noise depends on the traveled ray length between the nodes. DDN is more realistic compared to DIN, since the accuracy of ToF estimation is related to the received signal power, which itself is related to the traveled distance and transmit power. The Fisher information matrix (FIM) for a system affected by independent Gaussian noise can be computed as [82]

$$\mathbf{I}(\mathbf{x})_{i,j} = \frac{\partial \mathbf{f}}{\partial x_i}^T \mathbf{R}_v^{-1} \frac{\partial \mathbf{f}}{\partial x_j} + \frac{1}{2} \text{tr} \left[ \mathbf{R}_v^{-1} \frac{\partial \mathbf{R}_v}{\partial x_i} \mathbf{R}_v^{-1} \frac{\partial \mathbf{R}_v}{\partial x_j} \right], \quad (2.30)$$

where

$$\frac{\partial \mathbf{f}}{\partial x_i} = \left[ \frac{\partial f_1}{\partial x_i}, \frac{\partial f_2}{\partial x_i}, \dots, \frac{\partial f_N}{\partial x_i} \right]^T, \quad (2.31)$$

and

$$\frac{\partial \mathbf{R}_v}{\partial x_i} = \text{diag} \left( \frac{\partial [\mathbf{R}_v]_{11}}{\partial x_i}, \frac{\partial [\mathbf{R}_v]_{22}}{\partial x_i}, \dots, \frac{\partial [\mathbf{R}_v]_{NN}}{\partial x_i} \right), \quad (2.32)$$

and  $x_i$  is the  $i$ -th element of  $\mathbf{x}$ , i.e.,  $x_1 = x$ ,  $x_2 = y$ , and  $x_3 = z$ . Once the FIM is computed, the lower bound on the variance of the estimation error can be expressed as  $\text{CRB} = \sum_{i=1}^3 \text{CRB}_{x_i}$  where  $\text{CRB}_{x_i}$  is the variance of the estimation error in the

$i$ -th variable and it is defined as

$$\text{CRB}_{x_i} = [\mathbf{I}^{-1}(\mathbf{x})]_{i,i}. \quad (2.33)$$

For DIN, the noise covariance matrix is fixed which means that the second term of the FIM in (2.30) is zero, and consequently the CRB computation can be simplified. On the other hand, for DDN, the noise covariance matrix depends on the distance traveled between each anchor and the target node according to

$$\sigma_n^2 = K_E A(l_n, f), \quad (2.34)$$

where  $K_E$  is a constant that is related to the transmission power and the environment noise floor, and  $A(l_n, f)$  is the overall path loss, which can be defined as [10]

$$A(l_n, f) = \left(\frac{l_n}{l_0}\right)^\beta L(f)^{l_n - l_0}, \quad (2.35)$$

where  $f$  is the signal frequency, and  $l_n$  is the traveled distance which is taken in reference to some  $l_0$ . The path loss exponent  $\beta$  models the spreading loss, which is usually in between 1 and 2. The absorption coefficient  $L(f)$  can be obtained using an empirical formula [10].

The computation of  $\frac{[\partial \mathbf{R}_v]_{n,n}}{\partial x_i}$  in (2.32) requires the partial derivatives  $\frac{\partial \sigma_n^2}{\partial x_i}$  for  $i = 1, 2, 3$ . The derivative of  $\sigma_n^2$  w.r.t. the variable  $x_i$ , can be calculated as

$$\frac{\partial \sigma_n^2}{\partial x_i} = K_E \frac{l_n^{\beta-1}}{l_0^\beta} L(f)^{l_n - l_0} [\beta + l_n \ln L(f)] \frac{\partial l_n}{\partial x_i}, \quad (2.36)$$

which is related to the derivatives of the traveled ray length w.r.t.  $x_i$ . Once the above expressions are computed we are able to form  $\frac{\partial \mathbf{R}_v}{\partial x_i}$  for each variable  $x_i$ , and based on that the FIM and consequently the CRB can be calculated.

### 2.3.4 Localization with available depth measurements

The earlier localization algorithm does not require any depth information. In this subsection, we show how the optimization model will change if depth measurements are available. The result of this subsection is also useful for a comparison of the proposed algorithm with other existing state-of-the-art algorithms which mostly demand depth information. As stated in Subsection 2.2.4, the underwater target can measure its depth with a pressure sensor, and may send

this information to the central unit to potentially improve the localization accuracy. In this situation, the function  $\mathbf{f}(\mathbf{x})$ , the measurement vector,  $\tilde{\mathbf{t}}$ , and the covariance matrix of the noise vector,  $\mathbf{R}_v$ , have to be modified to the following format:

$$\mathbf{f}(\cdot) = [f_1(\cdot), f_2(\cdot), \dots, f_N(\cdot), f_z(\cdot)], \quad (2.37a)$$

$$\tilde{\mathbf{t}} = [\tilde{t}_1, \tilde{t}_2, \dots, \tilde{t}_N, \tilde{z}]^T, \quad (2.37b)$$

$$\mathbf{R}_v = \text{diag}(\sigma_1^2, \sigma_2^2, \dots, \sigma_N^2, \sigma_z^2) \quad (2.37c)$$

where  $f_z(\mathbf{x}) = z$ ,  $\tilde{z}$  is the noisy depth measurement, and  $\sigma_z^2$  is the power of the corresponding noise. Here, like the ToF measurements, it is assumed that the depth information is affected by Gaussian noise but does not generally depend on the distance from the anchors. As before, the ML solution for  $\mathbf{x} = [x, y, z]^T$  is the same as (2.29). The GNA and CRB can be extended using (2.37).

## 2.4 Target tracking based on time of flight measurements

### 2.4.1 Dynamic network model

To be able to localize a mobile target in a recursive manner (sometimes referred to as tracking), we exploit the EKF to estimate and track the position. Let us denote the location of the mobile target at time instant  $k$  as  $\mathbf{x}_k = [x_k, y_k, z_k]^T$ , and the corresponding state vector for the EKF as  $\mathbf{s}_k = [\mathbf{x}_k^T, \dot{\mathbf{x}}_k^T]^T$ , which contains both the location and velocity of the mobile target at time instant  $k$ .

In general, a discrete-time linear movement process model can be considered as

$$\mathbf{s}_k = \Phi \mathbf{s}_{k-1} + \mathbf{w}_k, \quad (2.38)$$

where the matrix  $\Phi$  relates the state of the previous time instant to the current one, and  $\mathbf{w}_k$  represents an i.i.d. Gaussian process noise with covariance matrix  $\mathbf{Q}_k$ .

It is noteworthy that we can further improve the accuracy of our location estimate with the help of a depth measurement in cases where this information can be acquired. However, for the network to be able to exploit the depth of the mobile target, the node will have to transmit a signal containing the depth information to the anchors which itself is resource-demanding due to the bandwidth limitations of the underwater channel. In order to make this process more bandwidth efficient, we suppose that the mobile target transmits the depth information every  $\rho$ -th

transmission frame. On the other hand, scenarios can be considered where the mobile target itself requires its location. Then, we can consider that depth information is always available. Although velocity measurements of the mobile target would aid the localization accuracy, in practice it requires the use of Doppler sensors, which increases the implementation cost as well as the computational complexity, and hence, we avoid measuring the velocity. Thus, the measurement model under consideration can be described as

$$\tilde{\mathbf{t}}_k = \mathbf{h}(\mathbf{s}_k) + \mathbf{v}_k, \quad (2.39)$$

$$\tilde{z}_k = z_k + v_k \quad \text{if } \text{mod}(k, \rho) = 0, \quad (2.40)$$

where  $\mathbf{h}(\cdot) = [h_1(\cdot), h_2(\cdot), \dots, h_N(\cdot)]^T$  is the function relating the state of the mobile target,  $\mathbf{s}_k$  to the wave travel times between the mobile target and the  $N$  anchors,  $\mathbf{t}_k = [t_{k,1}, \dots, t_{k,N}]^T$  (note that  $\mathbf{h}(\mathbf{s}_k) = \mathbf{f}(\mathbf{x}_k)$  from (2.27)).  $\mathbf{v}_k$  and  $v_k$  represent the i.i.d. Gaussian noise of the measurements with covariance matrix  $\sigma_t^2 \mathbf{I}_N$  and variance  $\sigma_z^2$ , respectively, where  $\mathbf{I}_N$  is the  $N \times N$  identity matrix. In the following, we explain how we can utilize the EKF for localization and tracking of a mobile target in an underwater environment.

### 2.4.2 Extended Kalman filter

The EKF algorithm for underwater tracking considering the exact SSP (EKF-ESSP) is shown in Algorithm 2. In this algorithm,  $\mathbf{P}_k$ ,  $\mathbf{R} = \sigma_t^2 \mathbf{I}_N$ , and  $\mathbf{Q}_k$  are the covariance matrix of the error in the state estimate, the measurement noise, and the process noise, respectively. To linearize the measurement equations, we compute the gradient of  $\mathbf{h}(\cdot)$  as  $\mathbf{H} = \nabla \mathbf{h}(\mathbf{s}) = \left[ \frac{\partial h_1}{\partial \mathbf{s}}, \dots, \frac{\partial h_N}{\partial \mathbf{s}} \right]^T$  where  $\frac{\partial h_i}{\partial \mathbf{s}} = \left[ \frac{\partial h_i}{\partial \mathbf{x}}^T, \frac{\partial h_i}{\partial \mathbf{z}}^T \right]^T$ , and  $\frac{\partial h_i}{\partial \mathbf{z}} = \mathbf{0}_{3 \times 1}$  for  $i = 1$  to  $N$ .

The gradient must be evaluated for time instant  $k$  as  $\mathbf{H}_k = \nabla \mathbf{h}(\hat{\mathbf{s}}_k^-)$ , where  $\mathbf{h}(\hat{\mathbf{s}}_k^-)$  is the a posteriori location estimate at the time instant  $k$ -th. This matrix can again be computed using (2.15) and (2.18). Following the derivation of the EKF, if depth measurements are available,  $\mathbf{H}_k$  and  $\mathbf{R}$  should be modified to  $\check{\mathbf{H}}_k$  and  $\check{\mathbf{R}}$  as

$$\check{\mathbf{H}}_k = \begin{bmatrix} \mathbf{H}_k \\ [0, 0, 1, 0, 0, 0] \end{bmatrix},$$

$$\check{\mathbf{R}} = \begin{bmatrix} \mathbf{R} & \mathbf{0}_{N \times 1} \\ \mathbf{0}_{N \times 1}^T & \sigma_z^2 \end{bmatrix} = \begin{bmatrix} \sigma_t^2 \mathbf{I} & \mathbf{0}_{N \times 1} \\ \mathbf{0}_{N \times 1}^T & \sigma_z^2 \end{bmatrix},$$

where  $\mathbf{0}_{m \times n}$  denotes an  $m \times n$  all zero matrix.

---

**Algorithm 2** EKF
 

---

```

Start with an initial location guess.
for  $k = 1$  to  $K$  do
  Next state:
   $\hat{\mathbf{s}}_k^- = \Phi \hat{\mathbf{s}}_{k-1}$ 
  Next error covariance:
   $\mathbf{P}_k^- = \Phi \mathbf{P}_{k-1} \Phi^T + \mathbf{Q}_k$ 
  if  $z$  info. is not available: then
    Compute the Kalman gain:
     $\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R})^{-1}$ 
    Update the state:
     $\hat{\mathbf{s}}_k = \hat{\mathbf{s}}_k^- + \mathbf{K}_k (\tilde{\mathbf{t}}_k - \mathbf{h}(\hat{\mathbf{s}}_k^-))$ 
    Update the error covariance:
     $\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^-$ 
  else
    Compute the Kalman gain:
     $\check{\mathbf{K}}_k = \mathbf{P}_k^- \check{\mathbf{H}}_k^T (\check{\mathbf{H}}_k \mathbf{P}_k^- \check{\mathbf{H}}_k^T + \check{\mathbf{R}})^{-1}$ 
    Update the state:
     $\hat{\mathbf{s}}_k = \hat{\mathbf{s}}_k^- + \check{\mathbf{K}}_k ([\tilde{\mathbf{t}}_k^T, \tilde{z}_k]^T - [\mathbf{h}(\hat{\mathbf{s}}_k^-)^T, \tilde{z}_k]^T)$ 
    Update the error covariance:
     $\mathbf{P}_k = (\mathbf{I} - \check{\mathbf{K}}_k \check{\mathbf{H}}_k) \mathbf{P}_k^-$ 
  end if
end for

```

---

### 2.4.3 Posterior Cramér-Rao bound

The lower bound on the mean squared error (MSE) of estimation for any discrete-time filtering problem, like the proposed EKF, can be computed via the posterior Cramér-Rao bound (PCRB) [83]. The recursive PCRB derived in [84] provides a formula for updating the posterior FIM from one time instant to the next. The posterior FIM sequence  $\mathbf{J}_k$  for a linear process and a non-linear measurement model can be computed as

$$\mathbf{J}_k = (\mathbf{Q}_k + \Phi \mathbf{J}_{k-1}^{-1} \Phi^T)^{-1} + \bar{\mathbf{H}}_k^T \mathbf{R}_k^{-1} \bar{\mathbf{H}}_k \quad (2.41)$$

where all the parameters have been defined earlier, except for  $\bar{\mathbf{H}}_k$ , which is the measurement gradient evaluated at the true location of the mobile target at the  $k$ -th time instant. It is noteworthy that, since we basically estimate the location of the mobile target and not its velocity, the PCRB of our location estimates will

correspond to the sum of the first three diagonal elements of  $\mathbf{J}_k^{-1}$

$$\text{PCRB}_k = \sum_{i=1}^3 [\mathbf{J}_k^{-1}]_{ii}. \quad (2.42)$$

Note that, the PCRB of an  $i$ -th element of  $\mathbf{s}$  corresponds to the  $i$ -th diagonal element of  $\mathbf{J}_k^{-1}$ .

## 2.5 Numerical results

In this section, we will conduct several simulations to evaluate the performance of our proposed algorithms in an environment with an isogradient SSP. We assume that the sound speed at the surface is  $b = 1480$  m/s, and it increases as a linear function of depth with a steepness of  $a = 0.1$ .

As a first simulation result, we compute the range error resulting from the straight-line wave propagation model with a constant velocity. This velocity can simply be assumed to be the sound speed at the anchor location or the target location, at the average depth between these two points, or the best linear approximation as given by (2.24). In Fig. 2.2, it is shown that as the target node gets further away from a surface anchor node, the error increases. Furthermore, it can be seen that among the different given constant speeds, the best linear approximation and the average one perform the best. However, these methods need the depth information of the target and anchor node which may not be available all the time.

At the network level, we consider four anchors that are located on the vertices of a cube with edge length 100 m, in which one vertex is located at the origin of the Cartesian coordinate system as depicted in Fig. 2.3. Here we consider the proposed localization algorithm (GNA-ESSP), and for the computation of each point in the following figures, we average the solution over  $10^4$  independent Monte Carlo runs. In addition, for the sake of comparison, we also consider an ordinary range-based localization algorithm which considers a constant sound speed defined as the average sound speed between two given nodes (GNA-ASSP)<sup>1</sup>. In the GNA-ASSP, the distance between two nodes is estimated via the measured ToFs, i.e., as  $t\bar{v}$ , where  $\bar{v}$  is a given constant wave velocity. In our simulations, we simply take  $\bar{v}$  as the average speed over the region where the deepest and the shallowest anchors are located. Hence, we simply set  $\bar{v} = [C(\max_n z_n^A) + C(\min_n z_n^A)]/2$ . In each

<sup>1</sup>For the GNA-ASSP, the same GNA as in Algorithm 1 is used, but the gradient is computed according to the linear dependency of the ToF to the range in a homogeneous medium.

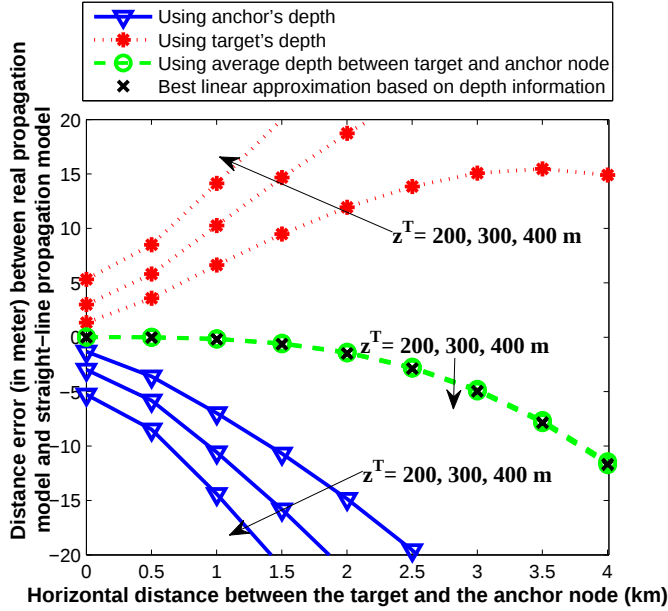


Figure 2.2: Error in range calculation resulting from the assumption of a straight-line propagation with a constant speed.

Monte Carlo run, the mobile target is located  $d$  meters away from a reference point in the 3-D environment, where  $d$  has a normal distribution with zero mean and standard deviation  $10\sqrt{3}$  m. For instance, in Fig. 2.3, the reference point of the target location is set at the anchors' center of gravity,  $[50, 50, 50]^T$ , and for each Monte Carlo run the mobile target has a random position around this reference point. Note that we plot the mean CRB (for the localization scenarios) and mean PCRB (for the tracking scenarios) as we average over different realizations of target locations and trajectories, respectively.

Based on the target position, the actual ToFs between the target and the anchors are computed. These actual ToFs can be obtained either from the analytical formulas or by ray-tracing simulators [85]. In order to compute the ToFs via ray-tracing simulators, a bunch of rays with different angles (so that the whole area is scanned) departs the transmitter and the trajectories of all the rays are computed. Among all these rays we pick the ones which have two properties: first, they are close enough to the receiving point, and second, they arrive sooner than the other rays to the receiving point. Then, we restart the above procedure with other set of rays, but with a finer angular resolution. This time, the initial angles of these rays

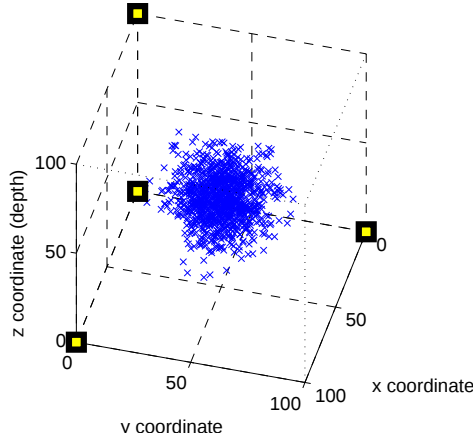


Figure 2.3: Random target node position around the reference point (here the anchors' center of gravity).

lay between the angles of the rays studied in the previous run. We continue this procedure until we get to the desired accuracy. Among the available simulators we have chosen the BELLHOP. The BELLHOP is a beam tracing model for predicting acoustic pressure fields in ocean environments and it can produce a variety of useful outputs including transmission loss, eigenrays, arrivals, and received time-series [86]. Notably, for the numerical results presented in this section, no reverberation due to scattering from fish or other biodata is considered.

After ToFs computation, noise is added to these ToFs and these noisy ToFs are used as an input to the considered localization algorithms. For our proposed GNA-ESSP the initial point is set to the anchors' center of gravity, and the stopping criteria are set to  $K = 7$  and  $\epsilon = 10^{-3}$ .

In Fig. 2.4, we investigate the effect of the measurement noise on the algorithms under consideration. For this simulation, the reference point is located at the anchors' center of gravity, and the measurement noise variance for all measurements is considered to be the same and distance-independent. Here, the horizontal axis represents the noise standard deviation (std.) on the ToF measurements, and the vertical axis is the root mean squared error (RMSE) of the location estimate which is given by

$$\text{RMSE} = \sqrt{\mathbb{E} [\|\hat{\mathbf{x}} - \mathbf{x}\|^2]}, \quad (2.43)$$

where  $\mathbb{E}\{\cdot\}$  represents the expectation operation.

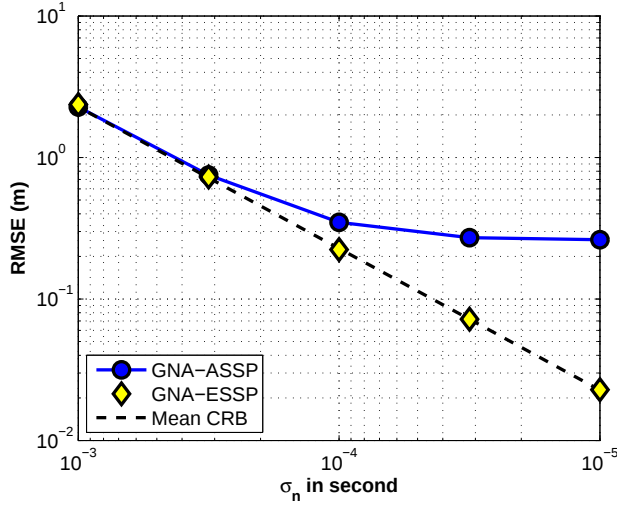


Figure 2.4: Localization performance with distance-independent measurement noise.

As is clear from Fig. 2.4, the performance of the proposed GNA-ESSP constantly improves by increasing the ToF measurement accuracy (decreasing the noise std.), and it falls on top of the mean CRB. On the contrary, the GNA-ASSP does not show any improvement after a given noise std. For large noise stds, both algorithms have the same performance. In that case, the proposed algorithm has no advantage, and the GNA-ASSP is preferred due to its lower complexity.

In Fig. 2.5, we investigate the performance when the variance of the measurement noise is distance-dependent. To evaluate the algorithm, we introduce a parameter which is the ratio of the expected squared travel time to the noise power:

$$\gamma = \frac{1}{N} \sum_{n=1}^N \frac{\mathbb{E}\{t_n^2\}}{\mathbb{E}\{v_n^2\}}. \quad (2.44)$$

The horizontal axis in Fig. 2.5 represents  $\gamma$  in dB, and the vertical axis is the RMSE of the location estimate. Here, we assume  $l_r = 1000$  m,  $\beta = 2$ ,  $K_E = -10$  dB, and  $L(f) = 1$  dB/km in (2.34) and (2.35) which is valid for frequencies below 20kHz. In this case, the proposed GNA-ESSP falls on top of the mean CRB while the GNA-ASSP again does not follow the mean CRB after a given  $\gamma$ . This shows that the GNA-ASSP which assumes a constant sound propagation speed is limited and cannot perform as efficient as the proposed algorithm.

In Fig. 2.6 and Fig. 2.7, we increase the  $x$ -coordinate of the reference point

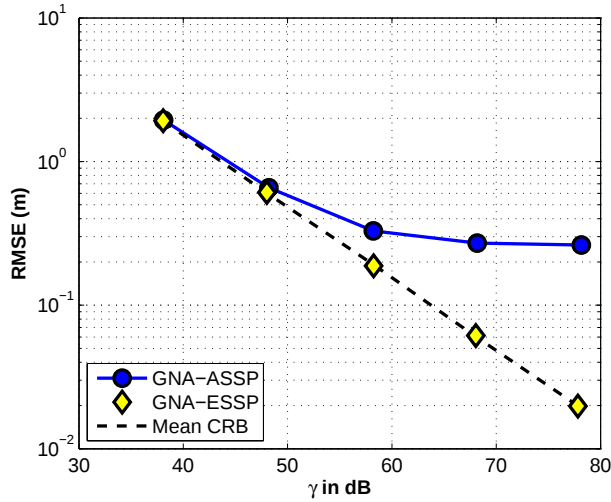


Figure 2.5: Localization performance with distance-dependent measurement noise.

around which the target node is located, while the  $y$  and  $z$ -coordinate of the reference point are  $[50, 50]$  as the  $y$  and  $z$ -coordinate of the anchors' center of gravity. In other words, the horizontal axis in these two figures represents how far the reference point is from the anchor locations. We set  $\sigma_n = 1$  ms for the DIN scenario, and for the DDN case, all the parameters are as defined before. From Fig. 2.6, it can be observed that as the horizontal distance between the target node and the anchor positions increases, the performance of the proposed GNA-ESSP degrades, but it still falls on top of the mean CRB. The reason for this phenomenon can be explained by the non-linear dependency of the ToF measurements to the target location, and the non-equal distribution of the error variance on the estimated location (it depends on the gradient of  $\mathbf{f}$  at that point). As a rule of thumb, the coordinate which has the lowest corresponding gradient suffers more from the noise. One more thing that can be extracted from this figure is that, although the performance of the proposed algorithm degrades as the distance between the target node and the anchors increases, the GNA-ASSP is affected more by this phenomenon and separates rapidly from the mean CRB as the distance increases.

It can be concluded from Fig. 2.7 that with a distance-dependent noise variance, not only the non-equal distribution of the error variance on the estimated location affects the performance of the algorithm, but also the increased noise power at larger distances leads to a further degradation.

Fig. 2.8 depicts the effect of the steepness of the SSP on the performance of the algorithms. Here, we consider a distance-independent noise variance, and we

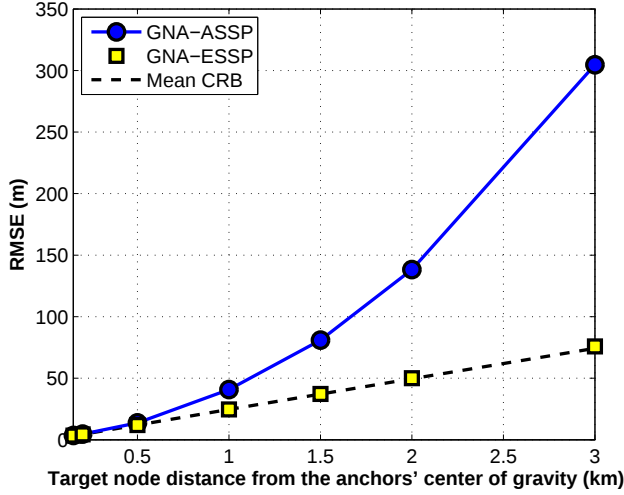


Figure 2.6: RMSE vs. the distance of the target node from the anchors' center of gravity, considering DIN.

set  $\sigma_n = 1$  ms. As can be seen from the figure, with an increase of the steepness of the SSP, the performance of the GNA-ASSP gets worse, but it has no effect on the proposed algorithm. Moreover, this effect is more clear for the case where the target node is further away from the center of gravity of the anchors.

Up to now, we did not consider any depth measurement. In order to compare the proposed localization algorithm with other existing state-of-the-art methods, we assume that the target measures its depth with a measurement noise std. of  $\sigma_z = 1$  m and this information can be used in the localization algorithm. In this comparison, the measurement noise of the ToFs is assumed to be DIN with  $\sigma_n = 0.5$  ms. In Fig. 2.9, we compare the performance of the proposed algorithm with the ones introduced in [5] and [6]. These algorithms estimate the horizontal range between two nodes based on the measured ToF and depth. Based on these range estimates, ML localization is performed as in [82].

The work in [5] uses LUTs to compute the mutual horizontal distance between two nodes. For our scenario, two LUTs have to be built. Each LUT has two entries, namely ToF and depth, and one output, namely the horizontal distance. Here, one LUT is responsible to estimate the horizontal distance between the target and the anchors which are located at  $z = 0$ , and the other LUT estimates the target's horizontal distance from the anchors located at  $z = 100$ . Each LUT covers a rectangular area of length 2500 m and width 100m with a resolution of 10 cm. Therefore, each LUT has 25 million points. Fig. 2.9 shows that the localization

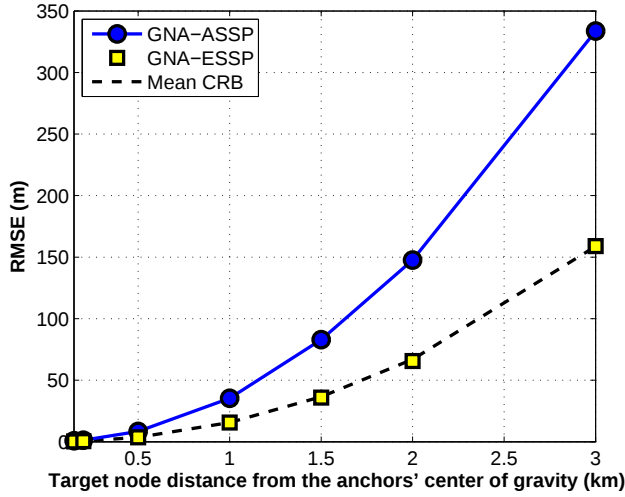


Figure 2.7: RMSE vs. the distance of the target node from the anchors' center of gravity, considering DDN.

algorithm based on [5] performs well, and its performance falls on top of the mean CRB like the proposed algorithm. However, as mentioned before, this algorithm works well only if the SSP of the environment remains constant. Otherwise, the computed values in the LUT are not valid anymore, or are less accurate than expected. The estimation of the horizontal distance in [6] has two phases; first, by measuring the depth and ToF information, the value of  $k_0$  in (2.3) is computed numerically, and second, by using  $k_0$  and taking the integral w.r.t. the depth of a point on a ray trajectory the value of the horizontal distance can be computed. However, in an inhomogeneous medium, a ray trajectory is not always a monotonic function of the depth, and as a result, whenever a path between two nodes crosses a specific depth more than once, which is quite common, the above algorithm is not valid anymore. This explains why the localization based on [6] works only for regions where the target is close to the anchors. Note though that, this algorithm performs optimal when the ray trajectories to all anchors are a monotonic function of the depth.

For the evaluation of the proposed tracking algorithm, the movement model is chosen to be a random walk with a sampling time step of  $T_s = 10$  s. The matrix  $\Phi$  as defined in (2.38) is then given by

$$\Phi = \begin{bmatrix} \mathbf{I}_3 & T_s \mathbf{I}_3 \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_3 \end{bmatrix},$$

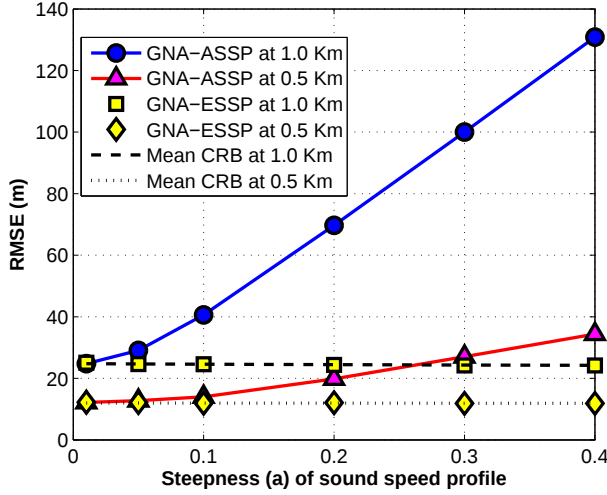


Figure 2.8: RMSE vs. SSP steepness.

and the process noise covariance matrix, which is assumed to be time-independent and only affecting the velocity, is given by

$$\mathbf{Q} = \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \text{diag}(\sigma_{\dot{x}}^2, \sigma_{\dot{y}}^2, \sigma_{\dot{z}}^2) \end{bmatrix},$$

where we assume that  $\sigma_{\dot{x}} = \sigma_{\dot{y}} = 10^{-2}$ , and  $\sigma_{\dot{z}} = 10^{-3}$ .

For all simulations, we set the initial location guess of the EKF to a point where it is  $[30 \text{ m}, 30 \text{ m}, 30 \text{ m}]^T$  away from the actual starting location of the target node. For each run, we consider  $K = 500$  movement steps, and we compute the positioning root mean squared error (RMSE) between the actual and estimated trajectories at the  $k$ -th time instant according to the following formula

$$\text{RMSE}_k = \sqrt{\frac{1}{K - K_1 + 1} \sum_{k=K_1}^{k=K} \mathbb{E} [\|\hat{\mathbf{x}}_k - \mathbf{x}_k\|^2]}, \quad (2.45)$$

where we try to avoid transient effects by setting  $K_1$  to a large number, e.g.  $K_1 = 300$ .

As a benchmark for our proposed tracking algorithm (EKF-ESSP), we again show the performance of an ordinary EKF which considers a straight-line wave propagation with a constant sound speed defined as the average sound speed between the depth of the deepest and the shallowest anchors (EKF-ASSP). In the

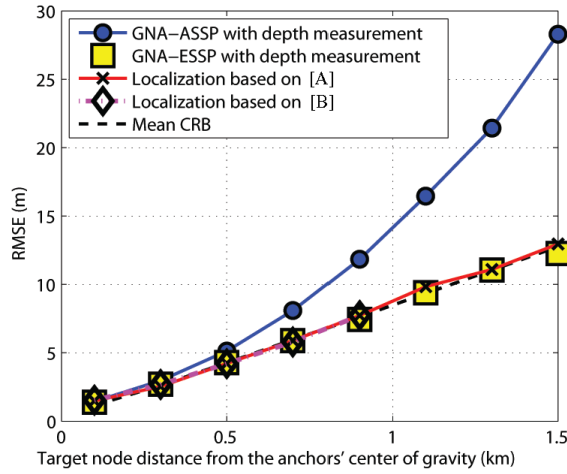


Figure 2.9: RMSE vs. the distance of the target node from the anchors' center of gravity, considering DIN, and depth measurement. Note that [A] and [B] refer to [5] and [6], respectively.

following simulations, we average over 5000 independent Monte Carlo trials and we set  $\sigma_t = 1\text{ms}$ ,  $\mathbf{x}_0 = [500\text{ m}, 50\text{ m}, 50\text{ m}]^T$ ,  $\rho = 10$  and  $\sigma_z = 1\text{ m}$ , unless otherwise mentioned.

In Fig. 2.10, we depict a tracking result example (a single Monte Carlo run) of the proposed EKF-ESSP and the EKF-ASSP algorithm, where the mobile target starts its journey from  $[3500\text{ m}, 50\text{ m}, 50\text{ m}]$ . It is shown that the proposed algorithm converges well to the real trajectory. However, the EKF-ASSP algorithm always has an offset from the real trajectory, and this offset increases as the mobile target gets further away from the center of gravity of the anchors.

In Fig. 2.11, we investigate the effect of the measurement noise on the algorithms under consideration. Here, the horizontal axis represents the noise std. on the ToF measurements. As is clear from the figure, the performance of the EKF-ESSP constantly improves when increasing the ToF measurement accuracy (decreasing the noise std.), while the EKF-ASSP does not show any improvement after a given noise std. Further, the performance of the EKF-ASSP gets worse when the distance of the mobile target (in its initial location) from the center of gravity of the anchors increases. For large noise stds, both algorithms have approximately the same performance.

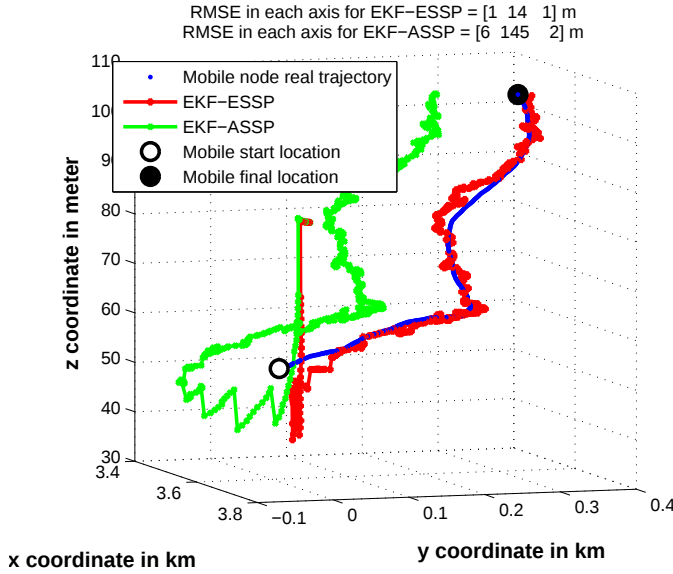


Figure 2.10: Tracking comparison.

Fig. 2.12 shows the effect of the availability of depth measurements on the RMSE performance of the proposed EKF-ESSP algorithm. Increasing the index shown on the horizontal axis ( $\rho$ ) means that we can less often measure the depth. From the figure, increasing  $\rho$  degrades the performance of the EKF-ESSP, although this degradation stops for large values of  $\rho$ . This means that the algorithm can work even if it relies only on ToF measurements.

Fig. 2.13 illustrates the effect of the depth measurement error (denoted by  $\sigma_z$ ) on the location estimation errors in each of the axes separately. As can be seen, the depth measurement error mainly affects the location estimates w.r.t. the vertical axis. The lower the depth measurement error, the better the  $z$  estimate. On the other hand, increasing the depth measurement error has no effect on the EKF-ESSP after a given value, since at these values, the EKF can acquire a better estimate from the ToFs than from the measured depth, and consequently ignores the depth information by decreasing its corresponding weight in  $\tilde{\mathbf{K}}$ .

Finally, Fig. 2.14 shows the effect of the number of anchors on the performance of the algorithm. The anchors are added one by one and are located on the vertices of the cube as defined before. Although increasing the number of anchors improves the performance of the algorithm slightly, it is not preferred due to the increase in computational complexity.

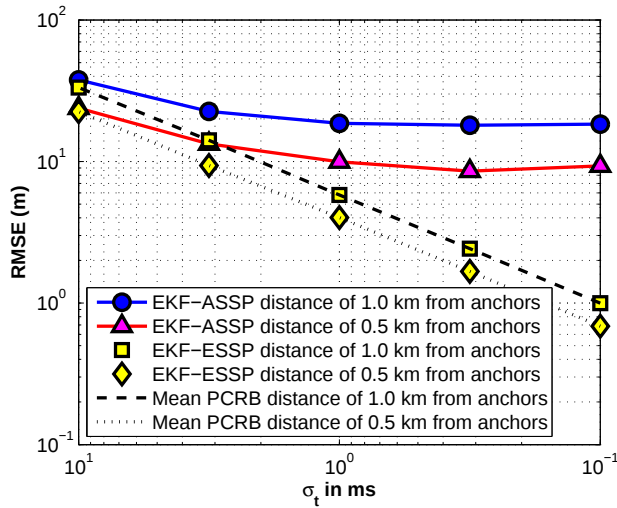


Figure 2.11: Effect of the time measurement error.

## 2.6 Conclusions

In this paper, we have considered the problem of target node localization and tracking in an underwater environment with an isogradient SSP. We have shown that the traditional terrestrial approaches for localization which assume a constant sound speed for the whole underwater environment are not so accurate. It is also shown that as the distance between two underwater nodes increases, the straight-line wave propagation model performs worse, since it does not follow the real propagation model. To solve this issue, we relate the ToF between two underwater nodes to their locations for an isogradient SSP, and formulate the localization problem as a time-based problem instead of a range-based one. Then, we use the Gauss-Newton algorithm and the extended Kalman filter with a proper formulation to solve the localization and tracking problem, respectively. It is shown that our proposed algorithms perform better than the algorithms based on a straight-line wave propagation model, especially for large distances. Although an isogradient SSP is not valid for all practical situations, the results can be used as an initial step towards more elaborate SSPs, since any given SSP can be modeled by several isogradient layers. This is a direction of further research.

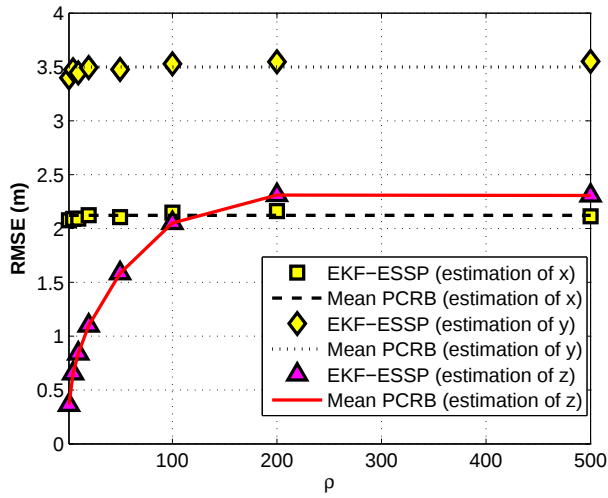


Figure 2.12: Effect of the measured depth report on the proposed tracking algorithm.

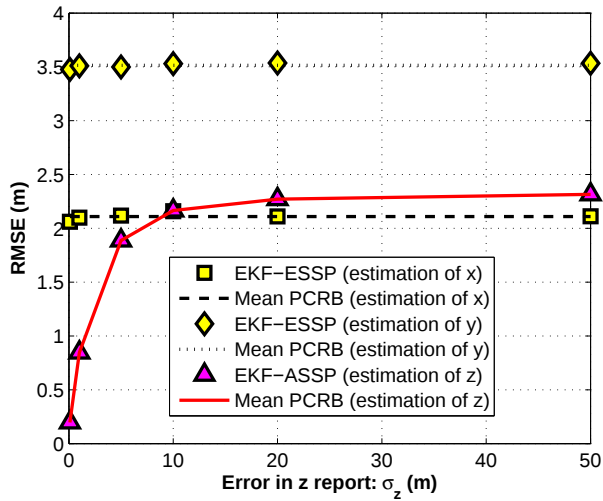


Figure 2.13: Effect of the depth measurement error on the proposed tracking algorithm.

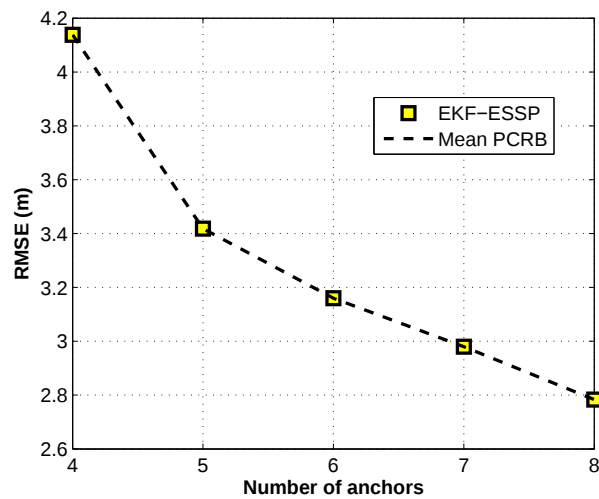


Figure 2.14: Effect of the number of anchors on the proposed tracking algorithm.



# Chapter 3

## Ranging in an Underwater Medium with Multiple Isogradient Sound Speed Profile Layers

Hamid Ramezani<sup>1</sup>

---

<sup>1</sup>This chapter is a verbatim copy of the journal paper published as: Hamid Ramezani, and Geert Leus. “Ranging in an underwater medium with multiple isogradient sound speed profile layers.” A special issue of *Sensors; Underwater Sensor Nodes and Underwater Sensor Networks*, Feb. 2012.

## Abstract

In this paper, we analyze the problem of acoustic ranging between sensor nodes in an underwater environment. The underwater medium is assumed to be composed of multiple isogradient sound speed profile (SSP) layers where in each layer the sound speed is linearly related to the depth. Furthermore, each sensor node is able to measure its depth and can exchange this information with other nodes. Under these assumptions, we first show how the problem of underwater localization can be converted to the traditional range-based terrestrial localization problem when the depth information of the nodes is known a priori. Second, we relate the pair-wise time of flight (ToF) measurements between the nodes to their positions. Next, based on this relation, we propose a novel ranging algorithm for an underwater medium. The proposed ranging algorithm considers reflections from the seabed and sea surface. We will show that even without any reflections, the transmitted signal may travel through more than one path between two given nodes. The proposed algorithm analyzes them and selects the fastest one (first arrival path) based on the measured ToF and the nodes' depth measurements. Finally, in order to evaluate the performance of the proposed algorithm we run several simulations and compare the results with other existing algorithms.

## 3.1 Introduction

Available wireless sensor network (WSN) localization techniques rely on mutual distances between sensors [87], which are for instance estimated from time of flight (ToF) measurements. In homogeneous medium, where the propagating wave speed is constant, the mutual distances between the nodes are linearly related to the ToFs. In contrast, the propagating wave speed inside an inhomogeneous medium is not constant, and depends on the location. In such a medium, the ToF between two nodes depends not only on the sound speed profile (SSP) of that medium but also on the position of the two nodes [88]. Therefore, the ToF is not linearly proportional to the Euclidean distance between the nodes, and the distance-dependent localization techniques are not appropriate for inhomogeneous media. As a result, they should be modified to ToF-based techniques.

In [74], the problem of localizing a node in an underwater environment with a known depth-dependent SSP is considered. As the target node measures the ToF from an anchor node, the corresponding constant range interval surface for the measured ToF is constructed. To construct a single constant range interval

surface, the path trajectory for each departing ray from an anchor node is calculated. Then on each path trajectory, a point is selected based on the measured ToF. All these points together form the constant range interval surface. After sufficient ToF measurements are taken, the position of the target node is estimated as a point whose sum of squared distances from all these surfaces is minimum. The main drawback of this approach is the computational complexity which is dependent on the network size and the required accuracy.

In our earlier work [89], we consider the problem of localizing a target node in an underwater environment with an isogradient SSP. There, we directly work with the ToF measurements, and we localize a target node based on the ToF equations. Since the algorithm is based on an analytical approach, the computational complexity is acceptable. Although the assumption of a single isogradient SSP is appropriate for a deep underwater medium, it is not valid for the entire environment. The sound speed at a given point in an underwater medium is affected by the salinity, water temperature and pressure of that point [90], and in general this causes the SSP to vary nonlinearly and even non-monotonically with respect to (w.r.t.) the depth.

In [91] and [72], the localization of an underwater WSN is investigated, in a 3D environment assuming knowledge of the nodes' depth. These algorithms can help us to convert the inhomogeneous underwater localization problem into a traditional 2D homogeneous distance-dependent localization problem, which is well-studied in the literature [49, 69]. We will show that as the depth information is known a priori, the ToF between two nodes only depends on the horizontal distance and thus also on the distance between the nodes. Hence, having the ToF measurement between two nodes, the pair-wise distance can be computed. As an example, based on the depth information and the SSP, a look up table (LUT) is built in [5], which relates the ToF measurement to the horizontal distance between two nodes. The algorithm of [5] is very fast, but to scan the whole inhomogeneous environment, a huge LUT is required which may not be practical. Furthermore, the SSP in an underwater medium is subject to changes in temperature and conductivity, and any change in SSP degrades the LUT accuracy and upsets the localization performance.

In [6], the problem of ranging in an inhomogeneous underwater environment is considered. A numerical range estimator is proposed which is based on reconstructing the slanted path using Fermat's principle and calculus of variations. Basically, after depth and time measurements, the authors form an integral equality which is taken over the depth between nodes. Then, they try to numerically calculate the constant defined by Snell's law. Afterwards, by using the computed

constant, they calculate the horizontal distance between the nodes through another integral equality. Their work is really comprehensive since with any given SSP, the horizontal distance is computable, but their algorithm to compute the constant value (defined by Snell's law) has an ambiguity, because in an inhomogeneous medium it is common that a traveling ray from one node to another node passes a given depth more than once. Since the depth of a point on a traveling ray trajectory is not a monotonic function of the depth, this phenomenon casts an ambiguity on the value of any integral taken w.r.t. the depth along the traveling path.

In this work, we analyze the acoustic signal propagation between two sensor nodes in an underwater environment. We use a ray-tracing approach to model the propagation which is a valid approximation for high-frequency signal transmission [5]. We assume that the underwater medium is composed of different layers with an isogradient SSP, which is a practical model for the actual SSP of the entire environment [2, 77]. We will show that in such an environment, if the depth information of two nodes located on a specific ray is known, then the positions of the crossing points, where the ray trajectory and the layer boundaries meet each other, can be obtained through a set of polynomial root finding equations. Based on these equations we are able to distinguish among different possible transmission paths between the nodes, and determine the fastest one. The proposed method for finding the fastest transmission path between the nodes can handle reflections from the surface and the seabed by adding more polynomial equations to the set. Another contribution of this paper is a novel method for accurate ranging between the nodes. The proposed algorithm computes the horizontal distance between two nodes based on the ToF and depth measurements. The algorithm estimates the range of a target by minimizing the difference between the measured ToF and the constructed ToF estimated from the known map in an iterative manner.

The rest of the paper is organized as follows. We describe the network model in Section 3.2, and we compute the ToF *versus* the node positions in Section 3.3. Next, in Section 3.4, we propose our ranging algorithm, and we extract its CRB. Then in Section 3.5, we evaluate the performance of the proposed algorithm through several simulations. Finally, we conclude the paper in Section 3.6.

## 3.2 Network model

Consider  $K$  anchor nodes with known locations and one target node in an underwater acoustic sensor network (UASN). The goal of the system is to estimate the position of the target node with ToF and depth measurements. To relate the

wave ToF to the node position inside an underwater medium, we are faced with the classical problem of how an individual ray behaves in the medium, and how a ray departing one node arrives at the other node.

Assume that the wave speed in a Cartesian coordinate system is a function of the position and is defined by  $c(x, y, z)$ . Then, we can compute the ToF between two nodes,  $\mathbf{x}^S$  and  $\mathbf{x}^E$  as

$$t = \int_{s(\mathbf{x}^S, \mathbf{x}^E)} \frac{ds}{c(x, y, z)} \quad (3.1)$$

where  $s$  is the arc length, which is related to the ray path according to the standard ray equation

$$\frac{d}{ds} \left( \frac{1}{c(x, y, z)} \frac{d\mathbf{x}}{ds} \right) = -\frac{1}{c^2(x, y, z)} \nabla c(x, y, z) \quad (3.2)$$

with  $\mathbf{x} = \mathbf{x}(s) = [x(s), y(s), z(s)]^T$  the position on the ray determined by the arc length  $s$ . It is clear from Equation (3.1) that in an inhomogeneous medium the ToF between two nodes is not linearly dependent on the distance between them, and is a function of the two node positions. Using the ToF measurements to  $K$  anchors, the location of a node can be obtained as,

$$\hat{\mathbf{x}} = \min_{\mathbf{x}} |\mathbf{t}(\mathbf{x}) - \mathbf{t}_m|^2 \quad (3.3)$$

where  $\mathbf{t}(\mathbf{x})$  is a  $K \times 1$  vector with the  $k$ -th element representing the ToF between the target node and the  $k$ -th anchor node, and  $\mathbf{t}_m$  is a vector denoting the noisy ToF measurements to all  $K$  anchors. To localize a node in a 3D inhomogeneous environment without depth information at least four anchors are needed, and the localization process is known as quadrilateration. If the depth information is available, it is possible to localize a node with just three anchors, but still we have to work with ToFs. For more information about how the anchor nodes inside the network are selected and how they can communicate with each other, the reader is referred to [91].

In an underwater medium where the sound speed varies only with depth, the availability of depth information does not only allow us to work with only three anchors, it also opens the door to convert the time-based localization problem into a traditional range-based one as explained next.

Since the SSP is only a function of depth, the problem of ray tracing between two nodes has a cylindrical symmetry around the line parallel to the  $z$ -axis and passing through one of the nodes. Hence, we can map the problem of ray tracing

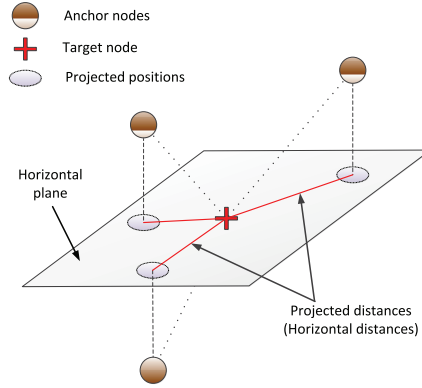


Figure 3.1: Projection of pair-wise distances on the horizontal plane crossing the target.

into a vertical plane that crosses the two nodes. As the depth of each node inside the plane is known, the ToF only depends on the horizontal range between the nodes. In other words, the horizontal distance between the nodes is the only variable that determines the ToF. Suppose now that the ToF of the first arrival path as a function of the horizontal distance is an invertible function, meaning that the ToF of the first arrival path between two points located at specific depths is a monotonic (actually increasing) function of the horizontal distance between the nodes. Then, one parameter can be computed from the other, and we are basically able to estimate the distance between the nodes using the corresponding ToFs. If the above assumption does not hold for a given environment, then there would be ambiguities in the ranging problem. However, this monotonicity is generally observed.

The conversion of a 3D underwater localization problem to a 2D one can now be explained as depicted in Figure 3.1. Using the ToFs and depth information, the unknown node computes the pair-wise distances to the other nodes. Then it projects the estimated distances on a horizontal plane at its depth. Finally, based on the projected distances, 2D multilateration is performed to compute its position [72]. Therefore, localizing a node in this case only requires the knowledge of the projected distances to the anchors or the estimated horizontal distances.

### 3.3 ToF Versus node positions

In order to relate the ToF to the node positions, we first require to find which ray departing the source reaches a specific destination. In this section, we

analytically find the rays that can travel between two nodes with known positions, and we compute their corresponding ToFs based on their trajectories. It is worth mentioning that in an underwater medium with fixed SSP, each ray departing a source can be uniquely characterized by its departing angle.

Here the SSP is considered as a piece-wise linear function of the depth which is a valid approximation according to the measured data [10]:

$$c^{(j)}(z) = a^{(j)}z + b^{(j)}, z^{(j-1)} < z < z^{(j)}, j \in 1, \dots, N \quad (3.4)$$

where  $z$  represents the depth,  $a^{(j)}$  and  $b^{(j)}$  are related to the chemical and physical characteristics of the  $j$ -th isogradient SSP layer, and  $N$  is the number of layers. In our previous work [89], we show how a ray can travel between two nodes located inside an isogradient SSP underwater environment. We review this work first for completeness.

### 3.3.1 ToF Versus node positions in a single layer

#### Exact propagation

In a single layer, each truncated ray (indexed by  $p$ ) between two points, *i.e.*,  $S_p$  and  $E_p$  in Figure 3.2, can be uniquely characterized if the position of the starting point, position of the end point, and SSP are known. In order to simplify the notation, we index the SSP from now on by the truncated ray index instead of the layer index. For instance, for the  $p$ -th truncated ray located at the  $j$ -th layer, we introduce the new notation

$$c_p(z) = a_p z + b_p = c^{(j)}(z) = a^{(j)}z + b^{(j)} \quad (3.5)$$

The relation between the ToF and the node positions can then be extracted from a set of differential equations characterized by Snell's law [5],

$$\frac{\cos \theta}{c_p(z)} = \frac{\cos \theta_p^S}{c_p(z_p^S)} = \frac{\cos \theta_p^E}{c_p(z_p^E)} = k_0, \text{ and } \theta \in \left( -\frac{\pi}{2}, \frac{\pi}{2} \right) \quad (3.6)$$

where  $\theta_p^S$  and  $\theta_p^E$  are the ray angles at the starting and end points, respectively,  $z_p^S$  and  $z_p^E$  represent the depth of the starting and end node, respectively, and  $k_0$  is constant along a ray traveling between the nodes (see Figure 3.2). Moreover, the parameters  $\theta$  and  $z$  represent the angle and depth of a given point along the ray.

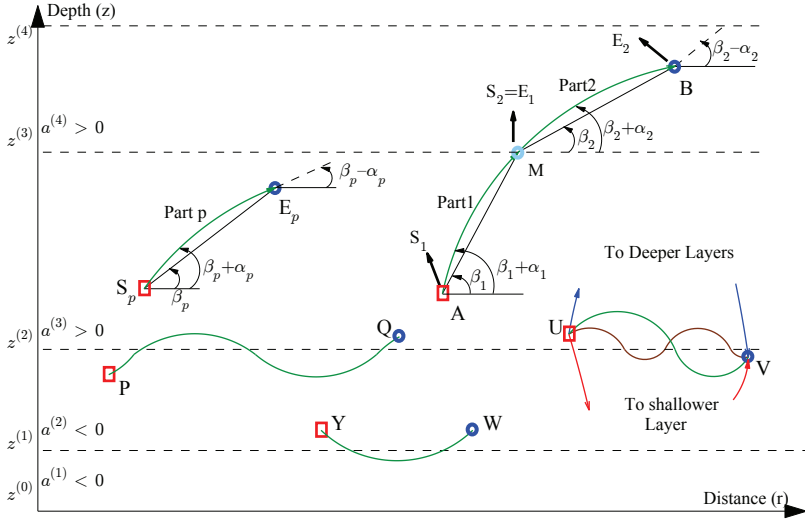


Figure 3.2: Samples of ray trajectories as they travel through different layers.

Now, we can write

$$\partial r = \frac{\partial z}{\tan \theta} \quad (3.7a)$$

$$\partial s = \frac{\partial z}{\sin \theta} \quad (3.7b)$$

$$\partial t = \frac{\partial s}{c_p(z)} \quad (3.7c)$$

where  $s$  is the arc length of a ray traveling between the two nodes, and  $t$  is its corresponding travel time. From Equations (3.5) and (3.6), by taking derivatives w.r.t.  $z$  and  $\theta$ , we can write

$$\partial z = -\frac{1}{a_p k_0} \sin \theta \partial \theta \quad (3.8)$$

Using the above differential equations, for the two points we have [89]

$$r_p^E - r_p^S = \sqrt{(x_p^E - x_p^S)^2 + (y_p^E - y_p^S)^2}, \quad (3.9a)$$

$$X_p = \frac{z_p^E - z_p^S}{r_p^E - r_p^S} \quad (3.9b)$$

$$Y_p = \begin{cases} \frac{K_p}{X_p}, & z_p^S \neq z_p^E \\ \frac{0.5a_p(r_p^E - r_p^S)}{b_p + a_p z_p^S}, & z_p^S = z_p^E \end{cases} \quad (3.9c)$$

$$K_p = \frac{0.5a_p(z_p^E - z_p^S)}{b_p + 0.5a_p(z_p^E + z_p^S)}, \quad (3.9d)$$

$$\tan \beta_p = X_p \quad (3.9e)$$

$$\tan \alpha_p = Y_p \quad (3.9f)$$

$$\theta_p^S = \beta_p + \alpha_p \quad (3.9g)$$

$$\theta_p^E = \beta_p - \alpha_p \quad (3.9h)$$

$$t_p = \frac{-1}{a_p} \left[ \ln \frac{1 + \sin \theta_p^E}{\cos \theta_p^E} - \ln \frac{1 + \sin \theta_p^S}{\cos \theta_p^S} \right] \quad (3.9i)$$

where  $r_p^E - r_p^S$  is the horizontal distance between the points,  $\mathbf{x}_p^S = [x_p^S, y_p^S, z_p^S]^T$  and  $\mathbf{x}_p^E = [x_p^E, y_p^E, z_p^E]^T$  are the coordinates of the starting and end points, respectively, and  $t_p$  is the traveling time of a truncated ray between these two nodes. Note that  $\beta_p$  represents the angle of the straight line connecting the nodes w.r.t. the horizontal axis, and  $\alpha_p$  is the angle between the actual ray and this straight line. From Equations (3.9g) and (3.9h), it can be seen that the ray deviations from the straight line at the starting and end point are the same but have opposite signs.

### Linear ToF approximation

The simple linear ToF approximation based on the depth information between two nodes in the same layer, which assumes a straight-line ray propagation, can be derived as

$$t_{\text{app},p} = \int_{z_p^S}^{z_p^E} \frac{1}{\sin(\beta_p)} \frac{dz}{c_p(z)} = \frac{d^{S_p E_p}}{z_p^E - z_p^S} \frac{1}{a_p} \ln \left( \frac{c_p^E}{c_p^S} \right) = \frac{d^{S_p E_p}}{c_p^E - c_p^S} \ln \left( \frac{c_p^E}{c_p^S} \right) \quad (3.10)$$

where  $d^{S_p E_p}$  is the straight-line distance between the starting point and end point of the truncated ray, and  $c_p^S$  and  $c_p^E$  are the sound speeds at the starting and end point, respectively. From Equation (3.10), it can be seen that  $t_{app,p}$  has a linear dependency on the range which is analogous to homogeneous medium. A similar approximation for a multi-layer medium which assumes a straight-line wave propagation between two nodes,  $\mathbf{x}^S$  and  $\mathbf{x}^E$ , can be obtained as

$$t_{app} = \frac{d^{SE}}{z^E - z^S} \sum_{p=1}^P \frac{1}{a_p} \ln \left( \frac{c_p^E}{c_p^S} \right) \quad (3.11)$$

where  $d^{SE}$  is the distance between the nodes, and  $P$  is the number of single-layer parts of the straight ray. For a straight ray, the number of single layer parts between two nodes is exactly the same as the number of layers it passes. The accuracy of the straight-line approximation will be evaluated in the numerical section.

Fermat's principal, which also leads to Snell's law, states that the path traveled by a ray between two points is the path that can be traversed in the least amount of time. Therefore, the approximated time based on a straight-line ray propagation,  $t_{app}$ , is always greater than the actual ToF. For instance, in Figure 3.3, it is shown that the difference between  $t_{app,p}$  and the actual ToF,  $t_p$ , is always positive. Here, we assume that the sound speed at  $z = 0$  is  $b_p = 1,480$  m/s, and we compute the ToF error for different values of depth, distance, and sound speed steepness. It is shown that as the absolute value of the SSP steepness,  $|a_p|$ , increases this error grows exponentially. In addition, for the low values of SSP steepness, the change in depth has less effect on this error in comparison with the change in pair-wise distance.

### 3.3.2 ToF Versus node positions for two adjacent layers

We start our analysis by considering a ray traveling between two points in adjacent layers. As illustrated in Figure 3.2, there are many ways for a ray to travel from U to V. An analysis of all possible rays is feasible, and we will later on discuss this. Now, we only focus on a ray which crosses the intermediate boundary only once, and does not propagate into other layers except these two adjacent layers such as the ray from A to B in Figure 3.2. In this scenario, when the  $r$  coordinate of the crossing point, M, is computed, we are able to relate the positions of the two nodes to the ToF. It can be seen that the ray has two parts, one indexed by  $p = 1$ , and the other by  $p + 1 = 2$ . The ending point of the first part is the starting point of the

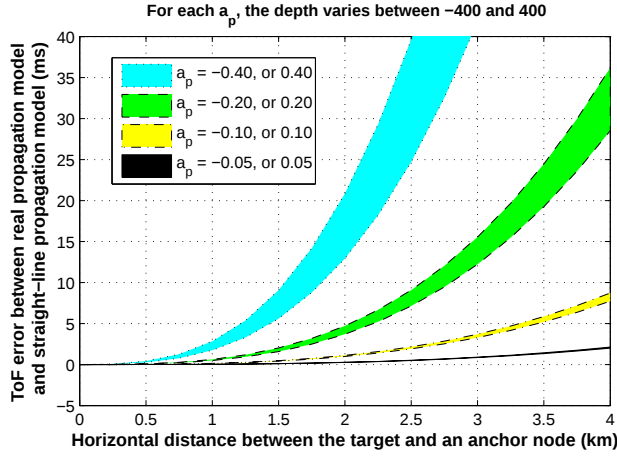


Figure 3.3: ToF error of the straight-line propagation model in a single layer for different values of range and depth.

second part. Thus, the two parts of the ray can be related to each other according to

$$\theta_2^S = \theta_1^E \quad (3.12)$$

Another representation for Equation (3.12) can be obtained by taking the tangent from both sides of the equation. Using Equations (3.9e) to (3.9h), the boundary equation can then be modified to

$$\frac{X_2 + Y_2}{1 - K_2} = \frac{X_1 - Y_1}{1 + K_1} \quad (3.13)$$

For a two-part ray, the combination of Sub-Equations (3.9b), (3.9c), and (3.9d) for each part, together with boundary Equation (3.13) forms a third-order polynomial root finding problem where the roots represent the possible  $r$  coordinates of the crossing point M. Notice that the node positions and the depth of M are known and as a result, the parameters  $K_1$  and  $K_2$  can be computed easily,  $X_1$  ( $X_2$ ) is inversely related to  $Y_1$  ( $Y_2$ ), and the only unknown parameter is  $r^M$  which determines  $X_1$  and  $X_2$ . Since there are at most three roots for a third order polynomial, there are at most three ways for a ray departing at A to reach B (note that we are still assuming that a ray crosses the boundary once, and propagates only in these two adjacent layers). For each of these possible rays the ToF can be calculated as,

$$t = \frac{-1}{a_1} \left[ \ln \frac{1 + \sin \theta_1^E}{\cos \theta_1^E} - \ln \frac{1 + \sin \theta_1^S}{\cos \theta_1^S} \right] + \frac{-1}{a_2} \left[ \ln \frac{1 + \sin \theta_2^E}{\cos \theta_2^E} - \ln \frac{1 + \sin \theta_2^S}{\cos \theta_2^S} \right] \quad (3.14)$$

### 3.3.3 Pattern definition for multi-layer ray propagation

To simplify the multi-layer analysis, we define the concept of *ray pattern*. A ray pattern is a set consisting of all possible rays that can travel between two points. For example, a ray pattern of 2.1.1.2 means that, the ray departs the starting point from the second layer, goes to the first layer, hits the surface, and arrives to the second node in the second layer. Therefore, a *ray pattern* has several properties. First, the number of digits used in the *ray pattern* indicates the number of single-layer parts a ray consists of. Second, it shows in which layer each part of a ray is located. Third, the reflection from the sea surface and the seabed can easily be modeled by this concept. Using the *ray pattern* concept, we are able to show how a ray can travel in a given medium, and which pattern may host the fastest ray.

### 3.3.4 ToF Versus node positions according to a given pattern

The procedure of ToF computation, as a function of the node positions for a ray which has multiple single-layer parts is the same as the two-part ray, but with more boundary equations. The combination of all these equations may form a higher order polynomial root finding problem, which consequently may increase the number of ways that a ray can travel between the two points. To predict how a ray may travel inside a multi-layer underwater area we introduce several lemmas below.

**Lemma 1:** The sound speed in a layer with zero steepness SSP is constant. Thus, the wave propagation inside that layer is along a straight line, and for that reason the parameter  $\alpha_p$  for each truncated ray at that layer is zero. In this case, the relation between the ToF and the node positions is modified into

$$t_p = \frac{1}{b_p} \sqrt{(r_p^E - r_p^S)^2 + (z_p^E - z_p^S)^2} \quad (3.15)$$

**Lemma 2:** Rays are bent toward the region where the sound speed is lower.

**Proof:** If we place Equation (3.9b) into Equation (3.9c), for two nodes at different

depths we obtain

$$Y_p = \frac{0.5a_p(r_p^E - r_p^S)}{b_p + 0.5a_p(z_p^E + z_p^S)} \quad (3.16)$$

The denominator of Equation (3.16) is the average of the sound speed at the two points and is always positive, and hence using Equation (3.9f) we understand that the angle  $\alpha_p$  has the same sign as  $a_p$ . When  $a_p$  is larger than zero, the ray angle at the starting point is greater than the ray angle at the end point or any other point along the ray trajectory. Therefore, the ray angle has a tendency to become smaller and to bend up where the sound speed is lower. On the other hand, when  $a_p$  is smaller than zero, the ray angle has a tendency to become larger and to bend down where again the sound speed is lower. Indeed, as depicted in Figure 3.2, within a layer, a ray bends toward the region where the sound speed is lower.

**Lemma 3:** In a layer, the depth along a truncated ray between two points,  $S_p$  and  $E_p$ , can exceed the region  $[z_p^S, z_p^E]$  if and only if (iff) the sign of  $\theta_p^S \theta_p^E$  is negative. The excess value can be computed as,

$$\Delta z = \begin{cases} \frac{a_p z_p^S + b_p}{a_p \cos \theta_p^S} (1 - \cos \theta_p^S), & \text{if } |\theta_p^S| < |\theta_p^E| \\ \frac{a_p z_p^E + b_p}{a_p \cos \theta_p^E} (1 - \cos \theta_p^E), & \text{if } |\theta_p^S| > |\theta_p^E| \end{cases} \quad (3.17)$$

**Proof:** In a single layer, the depth of a node along a given part as a function of the angle can be derived from Equation (3.6)

$$z = \frac{1}{a_p} \left( \frac{\cos \theta}{k_0} - b_p \right) \quad (3.18)$$

where  $k_0 = \frac{\cos \theta_p^S}{a_p z_p^S + b_p} = \frac{\cos \theta_p^E}{a_p z_p^E + b_p}$  is a positive constant for a given ray. It is obvious that  $z$  follows the behavior of  $\cos \theta$ , and its extremum occurs when  $\theta$  is zero. In other words, the depth of a node on a truncated ray exceeds the region  $[z_p^S, z_p^E]$  iff the signs of the angles at the starting and end points differ from each other. As a result, the excess value can be computed as,

$$\Delta z = \begin{cases} \max_{\theta} z(\theta) - \max \{z_p^S, z_p^E\} & \text{if } a_p > 0 \\ \min_{\theta} z(\theta) - \min \{z_p^S, z_p^E\} & \text{if } a_p < 0 \end{cases} \quad (3.19)$$

which leads to Equation (3.17).

In this way, it can be understood that if  $\Delta z$  for a one-part ray in a single layer is so large that a ray part crosses another layer, the assumption of a one-part ray propagation has to be changed into a three-part ray propagation, and the equations have to be reorganized accordingly (see for instance the ray in Figure 3.2 traveling between points Y and W).

**Lemma 4:** A ray can travel multiple times between two layers if the SSP has a local minimum between them.

**Proof:** A ray which is capable of traveling periodically between two layers has both maximum and minimum points of depth on its traveling path, as illustrated in Figure 3.2 between the points P and Q. Therefore, based on Lemma 2, it must bend from the first layer to second one, and from the second layer to the first one. Since a ray has a tendency to bend toward the low speed region, periodic traveling between two layers happens if the lower speed region is located between the two layers. In addition, the assumption of an isogradient SSP for each layer forces the total SSP to have a local minimum value at the boundary of these two layers.

**Lemma 5:** Reflections from the seabed and sea surface can be formulated as a boundary equation. In a shallow underwater environment, it is very common that a traveling ray hits the sea surface or the seabed before reaching the end point. Based on the physical properties of the seabed and the sea surface, the reflection can be formulated as a boundary equation. For instance, if we consider a perfect reflection from the seabed or the sea surface, the boundary equations can be obtained as

$$\begin{aligned}\beta_{p+1} + \alpha_{p+1} &= -(\beta_p - \alpha_p) \\ \beta_{p+1} &= -\beta_p\end{aligned}\tag{3.20}$$

Under the assumption of a perfect reflection, the reflected parts in different layers have the same properties as the corresponding non-reflected parts but with an axial symmetry around the line parallel to the  $z$ -axis crossing the reflection point as illustrated in Figure 3.4. Due to this symmetry which is resulted by the cylindrical symmetry of the ray propagation, the  $r$ -coordinate of the two crossing points around a reflection point are linearly related to each other, and one can be formulated by the  $r$ -coordinate of the other. Therefore, this does not change the degree of the polynomial resulting from such kinds of ray patterns.

Thanks to the piece-wise linear behavior of the SSP, we are now able to predict how a ray, which starts from a given point, can travel through different layers to arrive at a specific point. Having built a *ray pattern* using the above lemmas we can

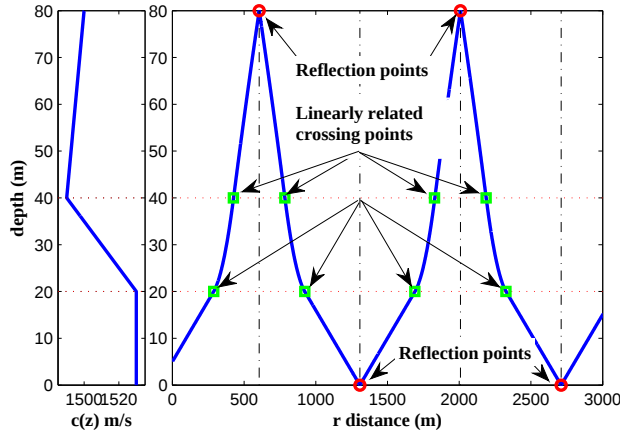


Figure 3.4: Linear dependency of the reflection and crossing points under the assumption of a perfect reflection.

then for every  $P$  part ray relate the ToF to the node's position using Equation (3.9) for each single-layer part and the following boundary equations

$$\frac{X_{p+1} + Y_{p+1}}{1 - K_{p+1}} = \frac{X_p - Y_p}{1 + K_p}, \text{ for } p = 1 \text{ to } P - 1 \quad (3.21)$$

The ToF of any possible  $P$  part ray can then be computed as

$$t = \sum_{p=1}^P \frac{-1}{a_p} \left[ \ln \frac{1 + \sin \theta_p^E}{\cos \theta_p^E} - \ln \frac{1 + \sin \theta_p^S}{\cos \theta_p^S} \right] \quad (3.22)$$

For instance, as illustrated in Figure 3.2, four kinds of rays can be predicted between the two points U and V which are described below.

- A ray may travel directly from point U, crossing the boundary once, and not leaving the two adjacent layers. Of course, as formulated before, even with such a limitation, there are at most three paths that can be included in this category.
- A ray may travel directly from point U, crossing the boundary multiple times, not leaving the two adjacent layers. For three crossings of the ray with the boundary, a four-part ray will be obtained. Based on Equation (3.9c), the  $r$ -coordinate of the second crossing path is related to the previous one

according to

$$r_2^E = r_1^E + 2 \frac{b_2 + a_2 z_{j-1}}{a_2} Y_2 \quad (3.23)$$

and similarly,

$$r_3^E = r_2^E + 2 \frac{b_3 + a_3 z_{j-1}}{a_3} Y_3 \quad (3.24)$$

and based on the boundary condition we have  $Y_3 = -Y_2$  and  $X_3 = 0$ . From the above equations, it can be derived that the crossing points in this scenario are linearly dependent, and therefore the combination of all equations forms a fourth-order polynomial root finding problem owing to the single independent added boundary equation. For the other scenarios with more than three crossing points we have the same analysis, and the problem again reduces to a fourth-order polynomial root finding problem.

- It is also possible that a ray from U to V passes other layers. For instance, in the scenario depicted in Figure 3.2, the fourth layer has a positive steepness and a ray may bend over at that layer thereby entering a shallower layer. According to the introduced lemmas this is possible, and we should consider it in our analysis.
- Similar to the fourth layer, this phenomenon may also happen in the first layer which forms the fourth category of traveling paths.

In Figure 3.5(a), we show how many rays can travel between two points located in an unbounded two-layer underwater medium. Here, we imagine that the two layers have the same steepness but with different signs, e.g.,  $a_1 = -a_2 = 0.1$ , and consequently the SSP has a minimum value at the boundary of the two layers. The sound speed at the boundary ( $z = 0$ ) is assumed to be 1,480 m/s. To compute the number of possible rays between two points, we fix the position of  $\mathbf{x}^S$ , and change the position of  $\mathbf{x}^E$  to cover a 100 m by 3.5 km area in the vertical plane as depicted in the figure. According to the discussed lemmas, the possible patterns that can propagate between these two points are 1.2, 1.2.1.2, 1.2.(1.2) ..., or 1.1, 1.2.1, 1.(2.1) ..., depending on where the two points are located. It is shown that as the pair-wise distance between the two nodes which are located close to the boundary of the two layers increases, the number of paths between them increases too. Around the region where SSP has a minimum value, a pattern with lower number of digits has a lower ToF, but a greater overshoot. Therefore, to compute the fastest ray in this region we can start searching with a simple pattern, and check

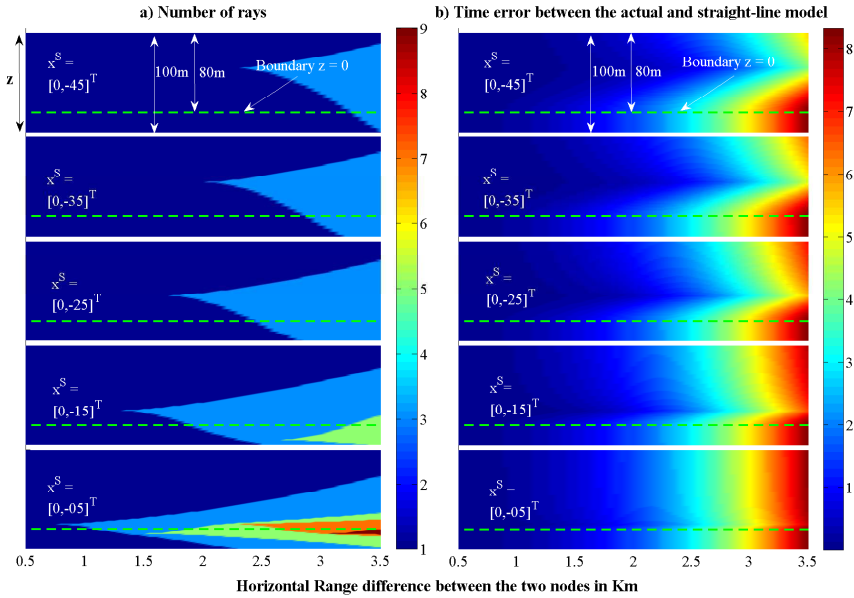


Figure 3.5: (a) Number of paths *versus* the location of the two nodes. (b) Range error due to the linear approximation of the ToF.

Lemma 3. If Lemma 3 holds, then we can stop, otherwise we should continue with a more complicated pattern.

For the same scenario as described above, Figure 3.5(b) shows the error due to a linear approximation of the ToF. From the figure, it can be observed that as the distance between the two nodes gets larger, the linear approximation performs worse. In addition, another effect that influences the accuracy of the straight-line propagation is the angle of this line with the horizontal axis. The larger the angle of the straight line with the horizontal axis, the better the accuracy of the straight-line propagation model.

The above analysis indicates that if a UASN is forced to utilize a straight-line propagation model due to any reason (e.g., system complexity) to achieve more accurate localization results in a noise-free environment, it is suggested that sensors are deployed in such a way that the angles of the straight lines between the nodes become large. However, when noisy measurements exist, we should also consider their influence on the mapped horizontal distances.

### 3.4 Pair-wise underwater ranging

To our best knowledge, the algorithm in [6] is the only available mathematical approach for underwater range estimation. In order to better understand this algorithm we shortly review it here. The horizontal range between two nodes at different depths in an underwater medium can be obtained from Equations (3.6) and (3.7)

$$t = \int_{z^S}^{z^E} \frac{1}{c(z)} \frac{1}{\sqrt{1 - [k_0 c(z)]^2}} dz, \quad 0 < k_0 < \min_z \frac{1}{c(z)} \quad (3.25a)$$

$$r^E - r^S = \int_{z^S}^{z^E} \frac{k_0 c(z)}{\sqrt{1 - [k_0 c(z)]^2}} dz \quad (3.25b)$$

where  $k_0$  is a constant defined by Snell's law,  $t$  is the ToF between two nodes, and  $r^E - r^S$  represents the horizontal distance between them. The estimation of the horizontal distance has two phases; first, by measuring the depth and ToF information, the value of  $k_0$  can be computed numerically from Equation (3.25a), and second, by substitution of  $k_0$  into Equation (3.25b), and taking the integral, the value of  $r^E - r^S$  can be obtained. However, in an inhomogeneous medium, a ray trajectory is not always a monotonic function of the depth, and as a result, whenever a path between two nodes crosses a depth more than once, which is quite common, the above formulas are not valid anymore. In this case, either Equation (3.25a) has no answer for  $k_0$  in the specified range, or the obtained answer is not valid.

#### 3.4.1 Proposed algorithm

Assume that, at a specific depth, the ToF of the fastest ray is a monotonic function of the horizontal range. In other words, a propagating wave at a specific depth reaches the destination with a smaller horizontal distance faster. Then, using the ToF and depth measurements, we can find the horizontal distance through a root finding algorithm such as Newton's method or bisection. Newton's method is very fast, but it requires the derivative of the ToF w.r.t. the horizontal distance which is hard to compute. The bisection method is robust, and it eventually finds the solution. However, it requires an upper and a lower bound on the horizontal distance. The lower bound can be set to zero, and the upper bound can be computed through multiplying the measured ToF by the maximum sound speed of the entire environment. In spite of the fact that other efficient numerical root-finding

algorithms can also be used, we utilize the simple bisection algorithm for the results in the simulation section.

Algorithm 3 shows the steps of the proposed algorithm. In this algorithm,  $K$  and  $E$  are the user-defined limits on the stopping criteria that determine when the algorithm exits from the loop,  $r_{\text{low}}$  and  $r_{\text{up}}$  are the lower and the upper bound, respectively. The algorithm starts by initializing the upper and the lower bound on the range, and then it computes the fastest ToF for the midpoint of the bounds. In order to calculate the fastest ToF, given the depth of the two points, different ray-patterns that may host the fastest ray are formed, and all the rays between the points are found and their corresponding ToFs are computed, *i.e.*, in Algorithm 3,  $t^{[l]}$  represents the ToF of the  $l$ -th found ray between the points. Then, among all these ToFs, the smallest one is selected. Next, based on the computed ToF, the lower, the upper, or both bounds are modified accordingly, and the procedure continues until one of the stopping criteria is met. The important factor that influences the complexity of the proposed algorithm is the number of ray patterns that may host the fastest ray. The ray patterns can be built very efficiently using the proposed Lemmas, still one can add more Lemmas (for a specific SPP) to reduce the number of ray patterns that may host the fastest ray between two nodes.

### 3.4.2 Cramér–Rao bound

The Cramér–Rao bound (CRB) expresses a lower bound on the variance of any unbiased estimator of a deterministic parameter. As mentioned before, since the depth information is known and the projection method can be used for localization, a given distance-based traditional localization algorithm works only with horizontal distances. Therefore, in this section we only derive the CRB for the horizontal distance estimation between two nodes. For the computation of the horizontal distance, three measurements are required: two depth measurements which are not directly related to the horizontal distance, and one ToF measurement. It is assumed that all the measurements are affected by Gaussian distributed noise as

$$\begin{aligned}\hat{t} &= t + n_t \\ \hat{z}^S &= z^S + n_z^S \\ \hat{z}^E &= z^E + n_z^E\end{aligned}\tag{3.26}$$

where  $n_t$ ,  $n_z^S$  and  $n_z^E$  are independent Gaussian distributed with variance  $\sigma_t^2$ ,  $\sigma_z^2$  and  $\sigma_z^2$ , respectively. The Fisher information matrix for estimating the horizontal

---

**Algorithm 3** Proposed Algorithm.

---

Compute horizontal distance upper and lower bounds,

$$r_{\text{low}} = 0,$$

$$r_{\text{up}} = \hat{t}c_{\text{max}}, \text{ where } c_{\text{max}} = \max_z c^{(j)}(z), j \in 1, \dots, N.$$

Initialize loop parameters,

$$e = r_{\text{up}} - r_{\text{low}},$$

$$k = 1.$$

**while**  $e \geq E$  and  $k \leq K$  **do**

  Compute the average value of the upper and the lower bound,

$$r = \frac{r_{\text{low}} + r_{\text{up}}}{2}.$$

  Find the smallest ToF for this horizontal distance

- Form all possible ray patterns hosting the fastest ray (see lemmas).
  - Compute ToF for each possible ray  $t^{[l]}(r, \hat{z}^S, \hat{z}^E)$ , (see Equation (3.22)).
  - Select the ray with the smallest ToF.
- $$t = \min_l t^{[l]}(r, \hat{z}^S, \hat{z}^E).$$

  Update the lower or the upper bound,

**if**  $t < \hat{t}$  **then**

$$r_{\text{low}} = r.$$

**else if**  $t > \hat{t}$  **then**

$$r_{\text{up}} = r.$$

**else**

$$r_{\text{low}} = r,$$

$$r_{\text{up}} = r.$$

**end if**

  Update loop parameters,

$$e = r_{\text{up}} - r_{\text{low}},$$

$$k = k + 1.$$

**end while**

Compute the estimated horizontal distance between the nodes.  $\hat{r}^E - \hat{r}^S = \frac{r_{\text{low}} + r_{\text{up}}}{2}$

---

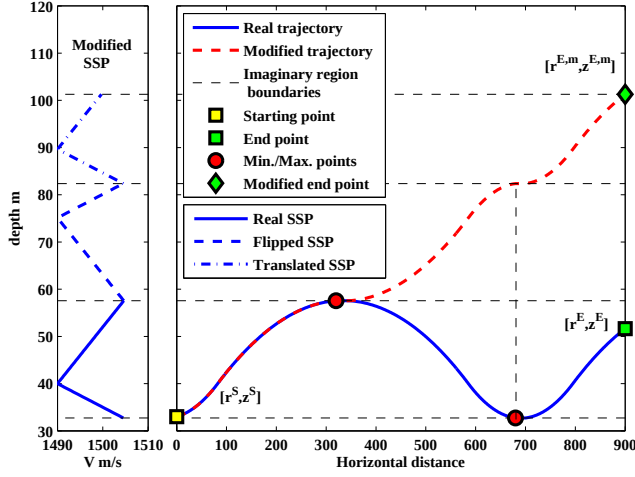


Figure 3.6: Changing the real ray trajectory into a trajectory which is a monotonic function of the depth.

distance  $(r^E - r^S)$ ,  $z^S$ , and  $z^E$  can be obtained as

$$\mathbf{I}(r^E - r^S, z^S, z^E) = \frac{1}{\sigma_t^2} \begin{bmatrix} \frac{\partial t}{\partial(r^E - r^S)} \\ \frac{\partial t}{\partial z^S} \\ \frac{\partial t}{\partial z^E} \end{bmatrix} \begin{bmatrix} \frac{\partial t}{\partial(r^E - r^S)} & \frac{\partial t}{\partial z^S} & \frac{\partial t}{\partial z^E} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & \sigma_z^{-2} & 0 \\ 0 & 0 & \sigma_z^{-2} \end{bmatrix} \quad (3.27)$$

In order to compute the partial derivative  $\frac{\partial t}{\partial(r^E - r^S)}$ , we modify the environment in such a way that we can compute the horizontal distance as an integral w.r.t. depth. In order to achieve this, we have to convert the horizontal distance and the ToF to monotonic functions of the depth. Therefore, a ray can not have maximum or minimum points on its trajectory w.r.t. the depth. Let us illustrate the proposed idea with an example. Assume that a ray has a maximum point on its trajectory. The ray angle is zero at this maximum point, and after that it changes sign. But, this sign change does not affect Snell's law, as it is related to the cosine of the ray angle. As a result, we can assume that the ray travels upward instead of downward as depicted in Figure 3.6, but in a new environment. In this new environment the SSP of each imaginary region must be changed accordingly. For instance, Figure 3.6 shows that the real SSP is flipped and translated in the first and second imaginary regions, respectively. In other words, the SSPs of the imaginary regions follow the behavior of the modified ray trajectory.

Note that the above conversion can only be done after we compute the fastest

ray, because only then we are able to locate the maximum and/or minimum points on the trajectory and build the new environment. Under this assumption, and using Equations (3.7b), (3.7b) and (3.6) we have

$$t = \int_{z^S}^{z^{E,m}} \frac{1}{c^m(z)} \frac{1}{\sqrt{1 - [k_0 c^m(z)]^2}} dz, \quad 0 < k_0 < \min_z \frac{1}{c^m(z)} \quad (3.28a)$$

$$r^E - r^S = \int_{z^S}^{z^{E,m}} \frac{k_0 c^m(z)}{\sqrt{1 - [k_0 c^m(z)]^2}} dz \quad (3.28b)$$

where m indicates that the variable is related to the modified environment. The above equations are similar to Equations (3.25a) and (3.25b), hence we can utilize the same approach used in [6] to compute the CRB, which results into

$$\text{var}(\hat{r}^E - \hat{r}^S) \geq \sigma_t^2 \frac{1}{k_0^2} + \sigma_z^2 \frac{1 - (k_0 c(z^E))^2}{(k_0 c(z^E))^2} + \sigma_z^2 \frac{1 - (k_0 c(z^S))^2}{(k_0 c(z^S))^2} \quad (3.29)$$

## 3.5 Numerical results

In this section we study the performance of finding the fastest path, as well as the proposed ranging algorithm in a multi-layer underwater environment. We consider two kinds of SSPs for our simulations as shown in Figure 3.7; the former is derived from the sound speed measurements in shallow water [1], and the latter is extracted from the sound speed of the Pacific Ocean and represents a deep water environment [2].

### 3.5.1 Ray propagation for shallow water

In this part of the report, based on the aforementioned lemmas, we analyze how a ray can propagate between two points inside the shallow water medium. Using the *ray pattern* concept, we are able to show how a ray can travel in a given medium, and which pattern may host the fastest ray. In Table 3.1, we show the family of patterns a ray may travel between two points through different layers. Since the depth of each node is known, we can select the proper patterns from the table, and form the corresponding polynomial formulas. By finding the roots of the polynomials, the ToF of each ray can be computed, and the fastest one will be recognized.

In Figure 3.8, we illustrate the ray propagation in a shallow underwater

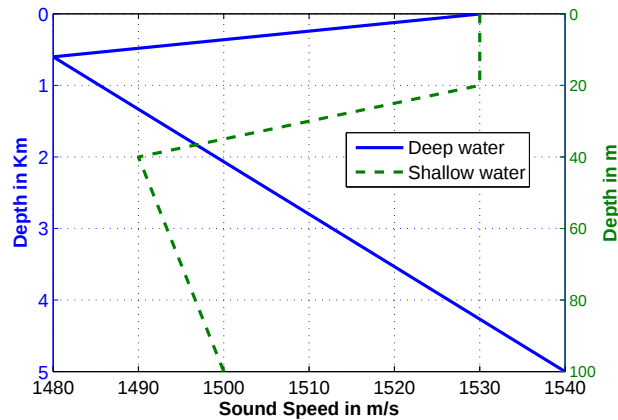


Figure 3.7: Sound speed profile for deep and shallow water.

|              | to layer 1 | to layer 2  | to layer 3  |
|--------------|------------|-------------|-------------|
| from layer 1 | 1          | 1.2         | 1.2.3       |
|              |            | 1.1.2       | 1.1.2.3     |
|              |            | 1.2.3.2     | 1.2.3.3     |
|              |            | 1.2.3.3.2   | 1.2.3.2.3   |
| from layer 2 | 2.1        | 2           | 2.3         |
|              |            | 2.1.1       | 2.1.1.2.3   |
|              |            | 2.3.2       | 2.3.3       |
|              |            | 2.3.2.3.2   | 2.3.2.3     |
|              |            | ⋮           | ⋮           |
|              |            | ⋮           | ⋮           |
| from layer 3 | 3.2.1      | 3.2         | 3           |
|              |            | 3.3.2.1     | 3.3         |
|              |            | 3.3.2.1.1   | 3.2.3       |
|              |            | 3.2.3.3.2   | 3.2.1.1.2.3 |
|              |            | 3.2.3.2.3.2 | 3.2.3.2.3   |
|              |            | ⋮           | ⋮           |

Table 3.1: All possible patterns that a fastest ray in a shallow underwater environment can follow.

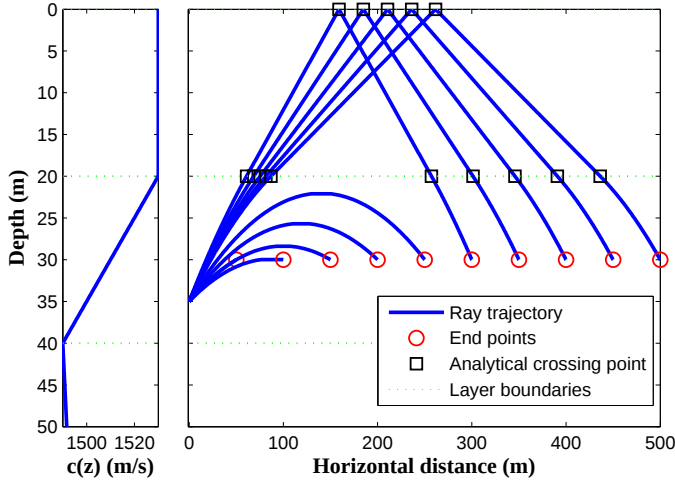


Figure 3.8: Sample of ray propagation between two nodes.

environment between two given points in the second layer. A ray can depart the first point and reach the second point in several ways; (a) it can directly propagate in the second layer without entering the other layers; (b) it can go to the upper layer, hit the surface and go back to the second layer; (c) it can go to the third layer and go back to the second layer; (d) it can go to the third layer, hit the bottom and go back to the second layer. Among all of these possibilities we choose the ray which has the lowest ToF. It is worth mentioning that the algorithm of [6] can not compute the correct horizontal range for any of the drawn blue-colored rays in Figure 3.8 except for the first one, since all other ray trajectories are not monotonic functions of depth.

In Figure 3.9, we show different possible rays that can travel between two points located in the second layer with a horizontal distance of 1,800 m. Based on the formulation, only three *ray patterns* can exist in this scenario, *i.e.*, 2.3.2, 2.3.3.2, and 2.1.1.2 (here we only consider one reflection from the surface, and only one reflection from the seabed in the existing ray patterns). Since the sound speed has higher values in the first and second layers, the fastest path belongs to the 2.1.1.2 pattern. It can be noted that if the horizontal distance between the two points increases, one *ray pattern* will be eliminated, namely 2.3.2.

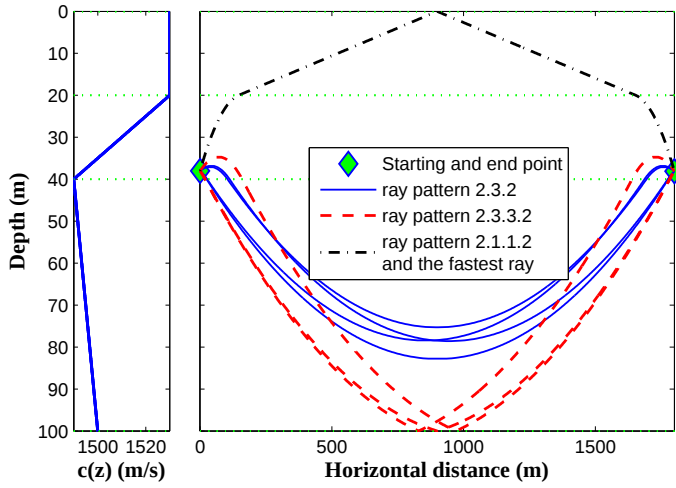


Figure 3.9: Different possible rays between two points in the second layer.

### 3.5.2 Ranging for deep water

In Figure 3.10, we compare the performance of the proposed range algorithm with the one introduced in [6], and with the algorithms which approximate the inhomogeneous underwater medium as a homogeneous one with a presumably constant sound speed, *i.e.*, we use a straight-line range computation based on the depth information. In this simulation, we consider Gaussian noise for the ToF and depth measurements with a standard deviation (std) of  $\sigma_t = 1$  ms and  $\sigma_z = 1$  m, respectively. In addition, we choose the deep water environment as a communication medium. The communication is between two points from different layers which are located at depth 550 m and 650 m, respectively. The root mean squared error (RMSE) for the horizontal distance estimation is computed by averaging over 1000 Monte Carlo simulations. As illustrated in this figure, the proposed algorithm performs well for all ranges while the algorithm of [6] has no definite solution from a given point as the horizontal range exceeds a given value. Furthermore, the straight-line algorithm degrades as the distance between the points increases.

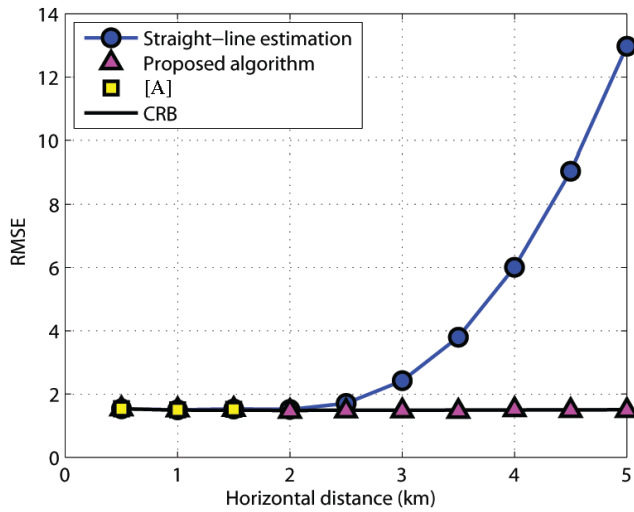


Figure 3.10: Performance of the proposed algorithm for deep water. Note that [A] refers to [6].

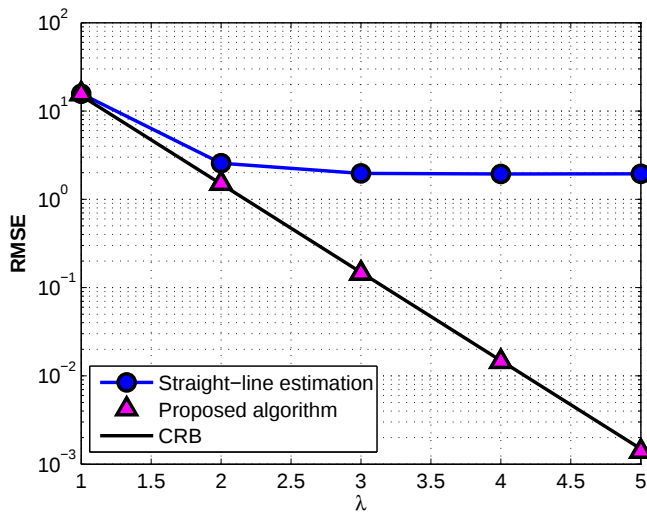


Figure 3.11: Performance of the proposed algorithm for difference values of noise power.

In Figure 3.11, we investigate the effect of the measurement noise on the algorithms under consideration. Here the depth of the two nodes is as before, and their horizontal range is fixed at 3 km. The horizontal axis represents how noisy the measured data are. The depth and ToF measurement noise powers are exponentially related to the parameter  $\lambda$ , *i.e.*,  $\sigma_z = 10^3 \sigma_t = 10^{2-\lambda}$ . It can be seen that, the performance of the proposed ranging algorithm constantly improves and attains the CRB when we increase the measurement accuracy, while the straight-line estimation does not show any improvement after a given noise power.

## 3.6 Conclusions

We have analyzed the problem of localizing a target node in an underwater environment. The inhomogeneous underwater medium upsets the linear dependency of the pair-wise distances to the time of flight. We have shown that, if the depth information of the unlocalized node is available, then the problem of underwater localization can be converted to the traditional range-based one. Dividing the underwater medium into several isogradient sound speed profile layers, we have completely analyzed how a ray can travel between two given points through using different Lemmas. Further, we have proposed an iterative algorithm for the range estimation between two nodes, and we have demonstrated that the proposed algorithm attains the CRB and performs superb in comparison with other existing algorithms. In the future, we want to extend this work for more elaborate SSPs (not necessarily multiple isogradient), especially the ones with one local minimum, for ranging and channel modeling applications.



# Chapter 4

## Cramér Rao Lower Bound for Underwater Range Estimation with Noisy Sound Speed Profile

Hamid Ramezani<sup>1</sup>

---

<sup>1</sup>This chapter is a verbatim copy of the journal paper published as: Hamid Ramezani, Raj Thilak Rajan, Milica Stojanovic, and Geert Leus. “Cramer Rao Lower Bound for Underwater Range Estimation with Noisy Sound Speed Profile.” 49th Annual Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, California November 8-11, 2015.

## Abstract

In this paper, the Cramér Rao bound (CRB) for range estimation between two underwater nodes is calculated under a Gaussian noise assumption on the measurements. The nodes can measure their depths, their mutual time of flight, and they have access to noisy sound speed samples at different depths. The effect of each measurement on the CRB will be analyzed, and it will be shown that for long distances, the effect of the sound speed measurement noise is dominant, and its impact depends on the positions of the nodes, actual sound speed profile, the number of sound speed samples, and the depths at which the sound speed samples are gathered.

## 4.1 Introduction

Range estimation is required mostly for sensor network localization and navigation. In this paper, we present an insight into underwater ranging via the Cramér Rao bound (CRB). The CRB expresses a lower bound on the variance of unbiased estimators of a deterministic parameter. The CRB for range estimation in a terrestrial environment where sensor nodes communicate with each other through radio frequency links has been investigated in [92]. There, time of flight (ToF) measurements are used for range estimation, and it is shown that the corresponding CRB depends on the signal bandwidth, wave propagation speed, and received signal to noise ratio. Apart from range information in the time delay, [93] has reached a more accurate formulation by also extracting range information from the amplitude of the received signal power. Although the results of the above papers give us a valid inception of the range estimation accuracy in free space, they do not justify why practical underwater range estimation (specifically for long distances) suffers from a higher inaccuracy than anticipated by the developed bounds.

Acoustic underwater communications is quite different from its terrestrial counterparts [10]. The propagation speed is not constant and it varies with temperature, salinity and pressure [90]. On the other hand, the underwater sensor nodes have the privilege to measure their depth via a pressure sensor. In [94], it is shown that knowing the depth information and the sound speed profile (SSP), a mutual distance between two nodes can be obtained from a single ToF measurement. Under these conditions, the CRB for range estimation has been derived for a multiple-isogradient sound speed profile in [94], and for a more general sound speed profile in [6]. However, in practice, the SSP has to be

measured. Consequently, a noisy sound speed measurement will indirectly affect the accuracy of range estimation.

In general, the SSP can be represented as a linear combination of  $N$  basis functions obtained from empirical data as [95]

$$c(z) = \bar{c}(z) + \sum_{n=1}^N a_n f_n(z), \quad (4.1)$$

where  $z$  represents depth,  $\bar{c}(z)$  is the nominal sound speed profile which is known a priori (obtained from historical data), and  $f_n(z)$  for  $n = 1, 2, \dots, N$  denotes the basis functions. In order to measure the sound speed at a certain depth, a CTD (conductivity, temperature, and depth) sensor is used. Gathering all the noisy sound speed measurements at  $M$  different depths leads to

$$\hat{\mathbf{c}} = \bar{\mathbf{c}} + \mathbf{F}\mathbf{a} + \mathbf{v}, \quad (4.2)$$

where  $\hat{\mathbf{c}} = [\hat{c}(z_1), \hat{c}(z_2), \dots, \hat{c}(z_M)]^T$  is a vector of noisy sound speed samples at different depths,  $\bar{\mathbf{c}} = [\bar{c}(z_1), \bar{c}(z_2), \dots, \bar{c}(z_M)]^T$ ,  $\mathbf{a} = [a_1, a_2, \dots, a_N]^T$ ,  $\mathbf{F}$  is an  $M \times N$  matrix with  $n$ -th column  $\mathbf{f}_n = [f_n(z_1), f_n(z_2), \dots, f_n(z_M)]^T$ ,  $\forall n = 1, 2, \dots, N$ , and  $\mathbf{v}$  is the zero-mean Gaussian distributed measurement noise with covariance  $\mathbf{R}_v = \sigma_c^2 \mathbf{I}_{M \times M}$ .

## 4.2 Ray tracing

The relation between the ToF and the node positions can be extracted from a set of differential equations characterized by Snell's law:

$$\frac{\cos \theta_s}{c(z_s)} = \frac{\cos \theta_d}{c(z_d)} = \frac{\cos \theta}{c(z)} = k_0, \quad \theta \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right), \quad (4.3)$$

where  $k_0$  is constant for a given ray,  $\theta_s$  and  $\theta_d$  are the ray angle at the source and the destination, respectively,  $\theta$  is the ray angle at any point between the source and the destination, and  $z_s$  and  $z_d$  are the depths of the source and destination, respectively, as shown in Fig. 4.1. The basic relationship between the ToF  $t$ , horizontal distance  $h$ , and depth  $z$  can be represented by

$$\partial h = \frac{\partial z}{\sin \theta},$$

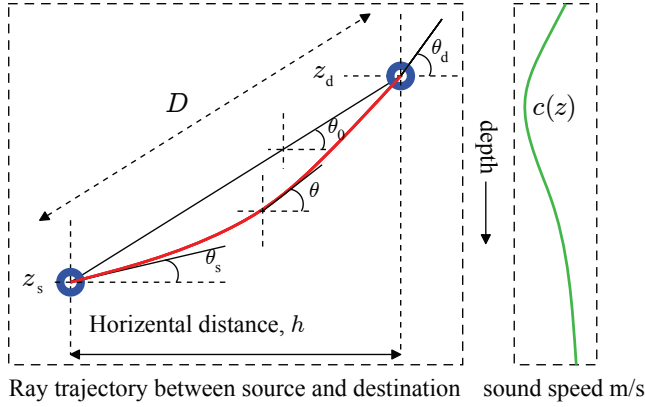


Figure 4.1: Ray propagation between the source and destination.

$$\partial t = \frac{\partial h}{c(z)}, \quad (4.4)$$

which will be used in calculating the ToF and horizontal distance between two points.

Although in an underwater medium with a general SSP, a ray between two points can have different patterns [94], here we assume that the ray crosses any depth between the two points only once. Using this assumption, the ToF and the horizontal distance can be formulated as (see (4.3) and (4.4))

$$t = \int_{z_s}^{z_d} \frac{1}{c(z) \sqrt{1 - [k_0 c(z)]^2}} dz, \quad (4.5)$$

$$h = \int_{z_s}^{z_d} \frac{k_0 c(z)}{\sqrt{1 - [k_0 c(z)]^2}} dz. \quad (4.6)$$

With the knowledge of the ToF, the SSP, and the depths, one can calculate  $k_0$  from (4.5), and use that in (4.6) to find the horizontal distance and eventually the range  $D$  between the two points. Unfortunately, the measurements are always noisy and that makes the estimation inaccurate. In the next section, we investigate the lowest achievable bound by any unbiased range estimator.

### 4.3 Cramér Rao lower bound

As explained before, the measurements are the ToF  $t$ , the depth of the source  $z_s$ , the depth of the destination  $z_d$ , and the samples of the sound speed at different depths  $\hat{\mathbf{c}}$ , each contaminated by respective Gaussian-distributed noise with zero mean and variance  $\sigma_t^2$ ,  $\sigma_z^2$ ,  $\sigma_z^2$ , and covariance  $\sigma_c^2 \mathbf{I}_{M \times M}$ , respectively. Stacking all the measurements in a vector we have

$$\mathbf{f}^T = [t, z_s, z_d, \hat{\mathbf{c}}^T]_{1 \times (M+3)},$$

with related noise  $\mathbf{w}$  whose elements are assumed to be independent of each other, and therefore

$$\mathbf{R}_w = \mathbb{E}[\mathbf{w}\mathbf{w}^T] = \text{diag}([\sigma_t^2, \sigma_z^2, \sigma_z^2, \sigma_c^2 \mathbf{1}_{1 \times M}]).$$

The estimated parameters are stated in

$$\mathbf{x}^T = [k_0, z_s, z_d, \mathbf{a}^T]_{1 \times (N+3)}.$$

Later, with a change of variables, the CRB for the horizontal distance and the mutual distance can be obtained.

For Gaussian distributed noise, the elements of the Fisher information matrix (FIM) can be obtained as [96]

$$[\mathbf{I}_\mathbf{x}]_{i,j} = \frac{\partial \mathbf{f}}{\partial x_i}^T \mathbf{R}_w^{-1} \frac{\partial \mathbf{f}}{\partial x_j} + \frac{1}{2} \text{tr} \left[ \mathbf{R}_w^{-1} \frac{\partial \mathbf{R}_w}{\partial x_i} \mathbf{R}_w^{-1} \frac{\partial \mathbf{R}_w}{\partial x_j} \right]. \quad (4.7)$$

Among the list of measurements, the variance of the ToF is distance dependent [97], and hence the second term of (4.7) is not zero for  $[\mathbf{I}_\mathbf{x}]_{1,1}$ . However, it can be ignored for high values of the SNR. The diagonal elements of the inverse of the Fisher information matrix give us the lowest bound on the variance of any unbiased estimator for  $\mathbf{x}$ . The elements of the inverse FIM (which is symmetric) can be calculated as (see Appendix A)

$$\begin{aligned} [\mathbf{I}_\mathbf{x}^{-1}]_{1,1} = & \sigma_t^2 \frac{1}{(\partial t / \partial k_0)^2} + \\ & \sigma_z^2 \frac{(\partial t / \partial z_s)^2}{(\partial t / \partial k_0)^2} + \sigma_z^2 \frac{(\partial t / \partial z_d)^2}{(\partial t / \partial k_0)^2} + \end{aligned}$$

$$\begin{aligned}
& \sigma_c^2 \frac{1}{(\partial t / \partial k_0)^2} \frac{\partial t}{\partial \mathbf{a}} (\mathbf{F}^T \mathbf{F})^{-1} \left[ \frac{\partial t}{\partial \mathbf{a}} \right]^T, \\
& [\mathbf{I}_{\mathbf{x}}^{-1}]_{1,2} = -\sigma_z^2 \frac{\partial t / \partial z_s}{\partial t / \partial k_0}, \\
& [\mathbf{I}_{\mathbf{x}}^{-1}]_{1,3} = -\sigma_z^2 \frac{\partial t / \partial z_d}{\partial t / \partial k_0}, \\
& [\mathbf{I}_{\mathbf{x}}^{-1}]_{1,4:N+3} = -\sigma_c^2 \frac{1}{\partial t / \partial k_0} \frac{\partial t}{\partial \mathbf{a}} (F^T F)^{-1}, \\
& [\mathbf{I}_{\mathbf{x}}^{-1}]_{2,2} = \sigma_z^2, \\
& [\mathbf{I}_{\mathbf{x}}^{-1}]_{3,3} = \sigma_z^2, \\
& [\mathbf{I}_{\mathbf{x}}^{-1}]_{2:4,4:N+3} = \mathbf{0}, \\
& [\mathbf{I}_{\mathbf{x}}^{-1}]_{4:N+3,4:N+3} = \sigma_c^2 (\mathbf{F}^T \mathbf{F})^{-1}, \tag{4.8}
\end{aligned}$$

where

$$\begin{aligned}
\frac{\partial t}{\partial k_0} &= \int_{z_s}^{z_d} \frac{k_0 c(z)}{(1 - [k_0 c(z)]^2)^{\frac{3}{2}}} dz, \\
\frac{\partial t}{\partial z_s} &= \frac{-1}{c(z_s) \sqrt{1 - [k_0 c(z_s)]^2}}, \\
\frac{\partial t}{\partial z_d} &= \frac{1}{c(z_d) \sqrt{1 - [k_0 c(z_d)]^2}},
\end{aligned}$$

and  $\frac{\partial t}{\partial \mathbf{a}}$  is a  $1 \times N$  vector  $[\frac{\partial t}{\partial a_1}, \frac{\partial t}{\partial a_2}, \dots, \frac{\partial t}{\partial a_N}]$  including the derivatives of the ToF to the coefficients of the basis functions in (4.1) and can be obtained as

$$\frac{\partial t}{\partial a_n} = \int_{z_s}^{z_d} \frac{2[k_0 c(z)]^2 - 1}{c^2(z)(1 - [k_0 c(z)]^2)^{\frac{3}{2}}} f_n(z) dz.$$

In 3D underwater localization [72], it is shown that knowing the depths of the sensor nodes, only the horizontal distance between each pair of nodes can be used for self-localization. Here, our parameters of interest are  $\mathbf{y}^T = [h, z_s, z_d, \mathbf{a}^T]$  whose CRB is given by a transform of (4.8) as

$$\mathbf{I}_{\mathbf{y}}^{-1} = \mathbf{H}^T \mathbf{I}_{\mathbf{x}}^{-1} \mathbf{H}, \tag{4.9}$$

where  $\mathbf{H}$  is the Jacobian matrix of  $\mathbf{y}$  with respect to  $\mathbf{x}$  and can be formulated as

$$\mathbf{H} = \left[ \begin{array}{c|ccc} \frac{\partial h}{\partial k_0} & \frac{\partial h}{\partial z_s} & \frac{\partial h}{\partial z_d} & \frac{\partial h}{\partial \mathbf{a}^T} \\ \mathbf{0}_{(N+2) \times 1} & \mathbf{I}_{(N+2) \times (N+2)} & & \end{array} \right]^T, \quad (4.10)$$

where

$$\begin{aligned} \frac{\partial h}{\partial z_d} &= k_0 c^2(z_d) \frac{\partial t}{\partial z_d}, \\ \frac{\partial h}{\partial a_n} &= \int_{z_s}^{z_d} \frac{k_0}{(1 - [k_0 c(z)]^2)^{\frac{3}{2}}} f_n(z) dz \\ \frac{\partial h}{\partial z_s} &= k_0 c^2(z_s) \frac{\partial t}{\partial z_s}, \\ \frac{\partial h}{\partial k_0} &= \frac{1}{k_0} \frac{\partial t}{\partial k_0}, \end{aligned} \quad (4.11)$$

and  $\mathbf{I}$  is the identity matrix. Using (4.10) in (4.9), and computing  $[\mathbf{I}_{\mathbf{y}}^{-1}]_{11}$  as the CRB of  $h$  results in

$$\begin{aligned} \text{CRB}_h &= \sigma_t^2 \frac{1}{k_0^2} + \\ &\quad \sigma_z^2 \frac{1 - [k_0 c(z_s)]^2}{[k_0 c(z_s)]^2} + \sigma_z^2 \frac{1 - [k_0 c(z_d)]^2}{[k_0 c(z_d)]^2} + \\ &\quad \sigma_c^2 \left\| \frac{\partial h}{\partial \mathbf{a}} - \frac{1}{k_0} \frac{\partial t}{\partial \mathbf{a}} \right\|_{(\mathbf{F}^T \mathbf{F})^{-1}}^2 \end{aligned} \quad (4.12)$$

where  $\|\mathbf{x}\|_A^2 = \mathbf{x}^T \mathbf{A} \mathbf{x}$ , and  $\frac{\partial h}{\partial \mathbf{a}} - \frac{1}{k_0} \frac{\partial t}{\partial \mathbf{a}}$  can be viewed as the inner product of  $g(z)$  and the basis functions for  $z \in [z_s, z_d]$  where

$$g(z) = \frac{1}{k_0 c^2(z) \sqrt{1 - [k_0 c(z)]^2}}. \quad (4.13)$$

Assume that we have sampled the sound-speed profile linearly for  $z \in [z_s, z_d]$ . It can then be shown that for a large number of samples ( $M \rightarrow \infty$ ) we have

$$\left\| \frac{\partial h}{\partial \mathbf{a}} - \frac{1}{k_0} \frac{\partial t}{\partial \mathbf{a}} \right\|_{(\mathbf{F}^T \mathbf{F})^{-1}}^2 \approx (\Delta z \mathbf{g}^T \mathbf{F})(\mathbf{F}^T \mathbf{F})^{-1}(\mathbf{F}^T \mathbf{g} \Delta z) \quad (4.14)$$

where  $\Delta z = \frac{|z_d - z_s|}{M}$ , and  $\mathbf{g}$  is a  $M \times 1$  vector whose elements are the samples of  $g(z)$  at different depths. Since  $\mathbf{F}(\mathbf{F}^T \mathbf{F})^{-1} \mathbf{F}^T$  can be seen as the projection matrix

on the columns of  $\mathbf{F}$ , (4.14) can be represented by

$$\left\| \frac{\partial h}{\partial \mathbf{a}} - \frac{1}{k_0} \frac{\partial t}{\partial \mathbf{a}} \right\|_{(\mathbf{F}^T \mathbf{F})^{-1}}^2 \approx \frac{|z_d - z_s|}{M} \int_{z_s}^{z_d} \tilde{g}^2(z) dz, \quad (4.15)$$

where  $\tilde{g}(z)$  is the projection of  $g(z)$  on the space which is spanned by the sound-speed basis functions which are defined in the range  $[z_s, z_d]$ . Since the energy of the signal is always greater than or equal to the energy of its projection, the right hand side of (4.15) can be approximately upper-bounded by  $\frac{|z_d - z_s|}{M} E_g$  where  $E_g$  is the energy of  $g(z)$  within the range  $[z_s, z_d]$ . Note that as  $M$  increases the effect of a noisy sound speed measurement on the CRB in (4.12) decreases.

The CRB of the range (denoted by  $D$ ) estimation as a function of  $h$ ,  $z_s$ , and  $z_d$ , i.e.,  $D = \sqrt{h^2 + (z_s - z_d)^2}$ , can now be formulated as

$$\text{CRB}_D = \mathbf{s}^T [\mathbf{I}_{\mathbf{y}}^{-1}]_{1:3,1:3} \mathbf{s}, \quad (4.16)$$

where  $\mathbf{s} = \partial D / \partial [h, z_s, z_d] = [\cos \theta_0, -\sin \theta_0, \sin \theta_0]^T$ , and  $\theta_0$  is the angle between the straight line from the source to the destination and the horizontal axis. The CRB of  $D$  can be simplified as

$$\begin{aligned} \text{CRB}_D = & \sigma_t^2 c(z_s)^2 \left( \frac{\cos \theta_0}{\cos \theta_s} \right)^2 + \\ & \sigma_z^2 \left( \frac{\sin[\theta_0 - \theta_d]}{\cos \theta_d} \right)^2 + \sigma_z^2 \left( \frac{\sin[\theta_0 - \theta_s]}{\cos \theta_s} \right)^2 + \\ & \sigma_c^2 (\cos \theta_0)^2 \left\| \frac{\partial h}{\partial \mathbf{a}} - \frac{1}{k_0} \frac{\partial t}{\partial \mathbf{a}} \right\|_{(\mathbf{F}^T \mathbf{F})^{-1}}^2, \end{aligned} \quad (4.17)$$

where the first two terms (related to a noisy ToF and depth measurement at the destination) are similar to what is extracted in [6]. Regarding the noisy sound speed samples, it can be observed that five factors affect the CRB: the measurement noise power, how the ray propagates (the actual sound speed profile), the number of samples  $M$ , the depth at which the samples are taken, and the inner product of  $g(z)$  with the truncated form of the basis functions.

The effects of noisy depth and ToF measurements on the range estimation have been analyzed before in [6] and [94] for a known SSP. In the numerical section, we focus more on the effect of noisy sound speed samples.

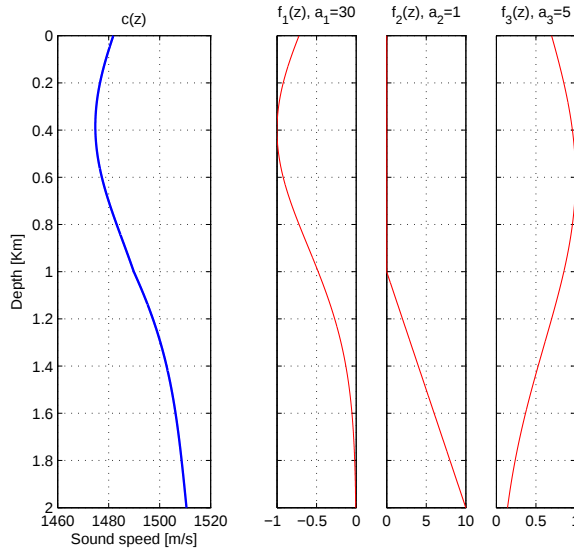


Figure 4.2: The sound speed profile as a function of depth is presented, along with corresponding coefficients of each basis function.

## 4.4 Numerical results

In this section, we evaluate the CRB of the range estimation for a given set up. A 2D environment with length  $D_h = 10$  km and depth  $D_z = 2$  km is considered. The nominal sound speed is set to  $\bar{c}(z) = 1500$  m/s, and it is assumed that the SSP is composed of three basis functions as depicted in Fig. 4.2. The depth of the source node is set to  $z_s = 0$ , and the coordinate of the destination point  $[h, z_d]$  varies over the area where the path trajectory crosses any depth only once. In addition, there are  $M = 10$  sound speed samples obtained at depth  $z_m = m \frac{D_z}{M}$  for  $m = \{1, 2, \dots, M\}$ . The variance of the measurements are  $\sigma_t^2 = 10^{-8}$ ,  $\sigma_z^2 = 1$ , and  $\sigma_c^2 = 1$ .

Fig. 4.3 illustrates the effect of each phenomenon on the CRB of  $D$  in part a, b, and c for a normalized noise power, and the overall CRB for  $D$  in part d. The white regions in these figures indicate that the initial assumption of path trajectory does not hold. It can be observed that for long distances a noisy ToF measurement deteriorates the performance almost similarly for any position of the destination point, and for a small ToF measurement noise power, its effect is trivial. Furthermore, the depth measurement error is not influential on the CRB for actual values of the noise power (e.g.,  $\sigma_z^2 = 1$ ). In contrast, the effect of a noisy SSP

is dominant here, and it is more dominant when the vertical distance between the source and the destination is lower than the horizontal distance.

## 4.5 Conclusions

The CRB of range estimation in an underwater sensor network has been derived under a depth-dependent sound-speed profile wherein noisy time of flight, depth, and sound speed measurements are available. The effect of each measurement noise on the CRB of range estimation has been evaluated analytically. For long distances, the noise power of the depth measurements does not play a significant role in the CRB, while those of the ToF and the sound speed samples are dominant. Over long distances, even with perfect ToF measurements, the range estimation cannot be perfect. We have shown that for a few sound speed samples at different depths, several factors play a vital role in the CRB such as the basis functions that the sound speed profile (SSP) is composed of, the actual SSP, the number of sound speed samples, and the positions of the source and the destination.

## 4.6 Appendix

The Fisher information matrix can be represented as

$$\mathbf{I}_x = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{D} \end{bmatrix} \quad (4.18)$$

where

$$\begin{aligned} \mathbf{A} &= \frac{1}{\sigma_t^2} \begin{bmatrix} \frac{\partial t}{\partial k_0} \\ \frac{\partial t}{\partial z_s} \\ \frac{\partial t}{\partial z_d} \end{bmatrix} \begin{bmatrix} \frac{\partial t}{\partial k_0} & \frac{\partial t}{\partial z_s} & \frac{\partial t}{\partial z_d} \end{bmatrix} + \frac{1}{\sigma_z^2} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \\ \mathbf{B} &= \frac{1}{\sigma_t^2} \begin{bmatrix} \frac{\partial t}{\partial k_0}, \frac{\partial t}{\partial z_s}, \frac{\partial t}{\partial z_d} \end{bmatrix}^T \frac{\partial t}{\partial \mathbf{a}}, \\ \mathbf{D} &= \frac{1}{\sigma_t^2} \begin{bmatrix} \frac{\partial t}{\partial \mathbf{a}} \end{bmatrix}^T \frac{\partial t}{\partial \mathbf{a}} + \frac{1}{\sigma_c^2} \mathbf{F}^T \mathbf{F}, \end{aligned} \quad (4.19)$$

and  $\mathbf{A}$  is a  $3 \times 3$  symmetric positive definite,  $\mathbf{B}$  is a  $3 \times N$ , and  $\mathbf{D}$  is a  $N \times N$  matrices.

Using the general formula of matrix inversion in block form, the inverse FIM

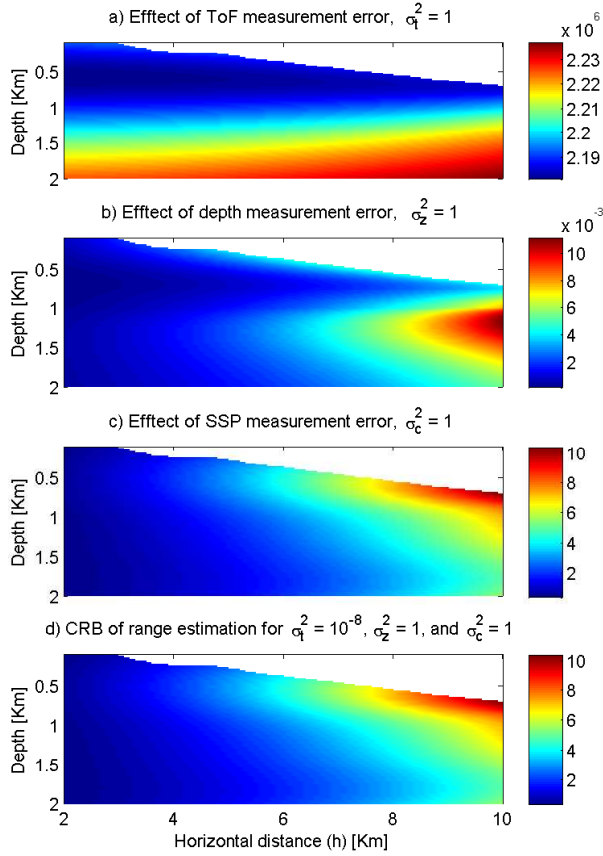


Figure 4.3: CRB of range estimation, a) effect of noisy ToF measurement, b) effect of noisy depth measurement, c) effect of noisy sound speed sample, d) the overall CRB of range estimation.

in (4.18) can be obtained as

$$\begin{aligned} [\mathbf{I}_x^{-1}] = & \\ \begin{bmatrix} (\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{B}^T)^{-1} & -\mathbf{A}^{-1}\mathbf{B}(\mathbf{D} - \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B})^{-1} \\ -(\mathbf{D} - \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{B}^T\mathbf{A}^{-1} & (\mathbf{D} - \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B})^{-1} \end{bmatrix}. \end{aligned} \quad (4.20)$$

The first block of (4.20) can be further expanded according to the Woodbury identity as [98]

$$\begin{aligned} (\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{B}^T)^{-1} = & \\ \mathbf{A}^{-1} + \mathbf{A}^{-1}\mathbf{B}(\mathbf{D} - \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{B}^T\mathbf{A}^{-1}, \end{aligned} \quad (4.21)$$

where the elements of the  $\mathbf{A}^{-1}$  are

$$\begin{aligned} [\mathbf{A}^{-1}]_{11} &= \frac{\sigma_t^2}{(\partial t / \partial k_0)^2} + \sigma_z^2 \frac{(\partial t / \partial z_s)^2}{(\partial t / \partial k_0)^2} + \sigma_z^2 \frac{(\partial t / \partial z_d)^2}{(\partial t / \partial k_0)^2} \\ [\mathbf{A}^{-1}]_{12} &= -\sigma_z^2 \frac{\partial t / \partial z_s}{\partial t / \partial k_0} \\ [\mathbf{A}^{-1}]_{13} &= -\sigma_z^2 \frac{\partial t / \partial z_d}{\partial t / \partial k_0} \\ [\mathbf{A}^{-1}]_{23} &= 0, \quad [\mathbf{A}^{-1}]_{22} = \sigma_z^2. \end{aligned} \quad (4.22)$$

Using (4.22) in  $\mathbf{A}^{-1}\mathbf{B}$  leads to

$$\begin{aligned} \mathbf{A}^{-1}\mathbf{B} &= \begin{bmatrix} \frac{1}{\partial t / \partial k_0} \\ 0 \\ 0 \end{bmatrix} \frac{\partial t}{\partial \mathbf{a}} \\ \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B} &= \frac{1}{\sigma_t^2} \begin{bmatrix} \partial t \\ \partial \mathbf{a} \end{bmatrix}^T \frac{\partial t}{\partial \mathbf{a}}, \end{aligned} \quad (4.23)$$

therefore the second term of (4.21) is

$$\begin{aligned} \mathbf{A}^{-1}\mathbf{B}(\mathbf{D} - \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{B}^T\mathbf{A}^{-1} = & \\ \frac{1}{(\partial t / \partial k_0)^2} \begin{bmatrix} \frac{\partial t}{\partial \mathbf{a}} \left( \frac{1}{\sigma_c^2} \mathbf{F}^T \mathbf{F} \right)^{-1} \left[ \frac{\partial t}{\partial \mathbf{a}} \right]^T & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \end{aligned} \quad (4.24)$$

which means that the second term of (4.21) only affects the lower bound of the

---

variance of the  $k_0$  estimator. Using a similar approach for the other matrix blocks in (4.20) we obtain (4.8).



---

---

## **PART III**

---

# **UNDERWATER LOCALIZATION PACKET SCHEDULING**



# Chapter 5

## Localization Packet Scheduling for Underwater Acoustic Sensor networks

Hamid Ramezani<sup>1</sup>

---

<sup>1</sup>This chapter is a verbatim copy of the journal paper published as: Hamid Ramezani, and Geert Leus. "Localization Packet Scheduling for Underwater Acoustic Sensor networks." IEEE Journal on Selected Areas in Communications, vol. 33 , no. 7, May 2015.

## Abstract

Medium access control (MAC) determines how sensor nodes share the channel for packet exchanging. To obtain the maximum network efficiency for accomplishing a specific task, the network has to adapt its parameters accordingly. In other words, different MAC protocols are required for different tasks. Localization is a crucial task of an underwater acoustic sensor network (UASN) which requires multiple packet exchanges. This article concerns the problem of designing a MAC protocol for a UASN which efficiently schedules the localization packets of the anchors. Knowing the relative positions of the anchors and their maximum transmission range, the scheduling protocol takes advantage of the long propagation delay of underwater communications to minimize the duration of the localization task. First, we formulate the concept of collision-free packet transmission for localization, and we show how the optimum solution can be obtained. Furthermore, we model the problem as a mixed integer linear program both in single-channel and multi-channel scenarios. Then, we propose two low-complexity algorithms, and through comprehensive simulations we compare their performances with the optimal solution as well as with other existing methods. Numerical results show that the proposed algorithms perform near optimum and better than alternative solutions.

## 5.1 Introduction

Due to the high attenuation of radio frequency signals or magnetic induction [19], [17] over large distances and high operating frequencies, underwater sensor networks usually employ acoustic signals for communications. Despite the fact that the underwater acoustic channel is one of the most challenging wireless propagation media, a large number of applications such as early warning systems (e.g., for tsunamis), ecosystem monitoring, oil drilling, military surveillance and so on, leaves us no choice but utilizing underwater acoustic sensor networks (UASNs). One of the requirements of a UASN is packet exchange among different nodes of the network which is handled by the medium access control (MAC) layer. Although extensive research has been done on the design of MAC protocols for wireless sensor networks (WSNs), the unique characteristics of the underwater acoustic communication environment, e.g., a very low and distance-dependent bandwidth [10], high power consumption in transmit mode [99], and long propagation delay, make the existing WSN algorithms and protocols inefficient for UASNs. To

overcome these issues, researchers have suggested several modifications to WSN MAC protocols or proposed different alternatives. For instance, in a time division multiple access (TDMA) system, in order to decrease the collision probability, the slotted floor acquisition multiple access (FAMA) [100] sets the time slot duration equal to the packet length plus the maximum network propagation delay. The distance aware collision avoidance protocol (DACAP) [101] uses request to send (RTS) and clear to send (CTS) handshaking to reserve the channel. To increase the network efficiency in point-to-point communications, DACAP estimates the mutual distance between two nodes and uses this information to minimize the duration of a handshake. With the knowledge of mutual distances among the sensor nodes, further steps have been taken in [102] where a transmitting node adjusts the time guard of its TDMA slot according to its distance to the other nodes. In addition to these modifications, many other works recently tried to improve the UASN performance by introducing new features such as a reservation period [103], back-off [104], parallel reservation strategy [105], scheduling [106] [107], and spatial fairness [108]. However, all of these mentioned protocols are dedicated to source-to-destination packet exchanges. In contrast, some tasks in a network may require packet broadcasting. Underwater localization [56] is an example of such an inevitable task where anchors broadcast their localization packets to other nodes. Generally, localization packets only have a few bits of information, mainly about the anchor's position and the time when the packet is transmitted. As shown in Fig. 5.1, the localization packet may also include other information such as a preamble, the anchor's ID, the guard time, and channel coding [109].

Kim et al. [110] evaluate the impact of the MAC on localization in a large-scale UASN. They show that the performance of a simple MAC protocol, namely carrier sense multiple access (CSMA), is better than T-Lohi [103] (a recently designed underwater MAC protocol). Ordered CSMA (OCSMA) [111] is a scheduling protocol which has been introduced for packet transmission in a fully connected network. In OCSMA, a coordinator finds the scheduling sequence based on the full knowledge of the relative positions of the anchors, and informs them of the

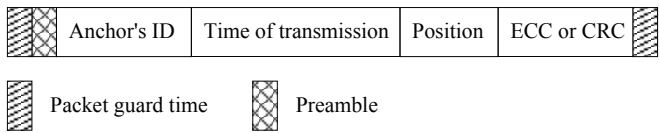


Figure 5.1: Structure of a localization packet.

resulting sequence. Then, the anchors start their packet transmission one after another according to the given scheduling sequence. Nevertheless, this kind of broadcasting protocol is not optimum, because it does not support simultaneous transmission for a broadcasting task. To overcome this problem, a single-hop all-to-all broadcasting transmission scheduling (AAB-MAC) is proposed in [112]. Knowing the propagation delay matrix, i.e., the propagation delays between all nodes, the goal of this protocol is to minimize the all-to-all transmission period in a way that no collisions occur. Although AAB-MAC performs better than OCSMA, it cannot be used for the localization task, because first we do not know the positions of all the underwater sensor nodes, and second, using the AAB-MAC only for the anchor nodes causes collisions at the sensor nodes. There are also a few other broadcasting MAC protocols [113] [114] which are not suitable for the localization task, because they do not consider collision-free broadcasting by the localization beacon.

In [115], the problems of position estimation and synchronization are combined with a recently proposed localization packet scheduling [116]. The authors consider the anchors that are within the communication range of each other, and schedule them to transmit their packet in a such a way that none of the sensor nodes experiences a collision. They have also extended their proposed idea for a large scale network in [117]. In contrast to these works, we consider multi-channel and dynamic multi-channel packet scheduling. We also evaluate the collision-free conditions when the anchors are not in the communication range of each other. Furthermore, we show how the optimization problem can be represented as a mixed integer linear program (MILP) problem.

Two efficient tone broadcast MAC protocols (TB-MACs) are proposed in [118] which are an adaptation of slotted-ALOHA and slotted-FAMA [100], but modified to work with broadcast traffic. Before broadcasting, TB-MACs use different handshaking mechanisms (NACK and NCTS instead of ACK and CTS) to handle the ‘reply storm’ problem. Albeit that these broadcasting protocols aim to enhance the network efficiency through reducing the handshaking overhead, they still rely on TDMA-like signaling which is not advisable for underwater networks when the node positions are known.

Beside the relative position information of the nodes, another factor that increases the system efficiency is the use of several independent channels for packet exchanges [119]. According to [120], multi-channel MAC protocols help to improve the network efficiency. The paper [121] analytically evaluates the idea of multi-channel MAC protocols, and shows that the theoretical analysis

closely follows the estimated system performance which is better in comparison with single-channel MAC protocols. The multi-channel packet exchange scheme basically reduces the possibility of collision in the network. However, it is not clear how it behaves in a collision-free packet scheduling which is tackled in this paper.

In contrast to the above works on underwater MAC protocols, in our previous papers, we have focused on designing a scheduling protocol for the specific task of anchor-based partially-connected single channel [122] and multi-channel underwater localization [122]. To do that, we have utilized the information about the relative positions of the anchors and their maximum transmission range to minimize the duration of the localization task. The localization procedure finishes when all the anchors have transmitted their packets. In this work, we combine the problem of localization packet scheduling for single-channel and multi-channel networks, and mention practical issues for this problem. Furthermore, we state that the problem is NP-hard, and we show how the optimal solution can be obtained, and how the problem can be converted to an standard MILP problem. Our contributions are listed below.

- The problem of minimizing the duration of the localization task is formulated in single-channel and multi-channel partially connected networks.
- For the localization task, an anchor is usually not interested in the transmitted packets from other anchors, unless it wants to estimate a physical phenomenon, or other information that is included in the localization packets. If that is the case we talk about the broadcast scenario. We show how the problem can be modified to support broadcast packets.
- The concept of dynamic multi-channel packet scheduling is introduced. In this approach, the system is able to split the existing channel into a few subchannels, and to schedule the localization packets not only in time but also in a specific subchannel.
- Several practical issues such as multi-path, sound speed variations, interference and moving anchors are considered, and it is shown how the formulation can be adapted accordingly.
- It is shown that the optimization problem can be converted to a combinatorial one which is NP-hard. Furthermore, it is shown how the optimum solution can be obtained through exhaustive search among all possible solutions.

- We also model the optimization problem as a MILP problem, and use tools such as CPLEX to solve the problem.

The rest of the paper is organized as follows. In Section 5.2, we explain the network model, define the concept of collision-free anchors, and formulate the problem. Next, in Section 5.3, we show how the optimum solution of the problem can be obtained, and in Section 5.4, we introduce the MILP model of the problem. Further, in Section 5.5, a number of novel algorithms to tackle the problem are proposed. Section 5.6 evaluates the performance of the proposed algorithms through several simulations, and finally, Section 5.7 concludes the paper and mentions some future works.

## 5.2 Network model

We consider an underwater sensor network with  $N$  surface-located anchor nodes (they can be located anywhere if their positions are known) with a maximum communication range of  $R$  meters. The following assumptions are made in this work.

- The anchors are equipped with GPS devices, as well as radio (or satellite) and half-duplex acoustic modems. It is further assumed that the anchors are synchronized with each other.
- The information about the positions of the anchors can be collected by a fusion center through their radio modems.
- There is no information about the position of the underwater sensor nodes, and they can be located anywhere in the operating area. In addition, they are not necessarily synchronized with the anchors.
- The sensor nodes are equipped with an inertial navigation system, and freely move in the environment, while the anchor nodes only drift around their nominal positions.

The localization task is required repeatedly in the network, and is carried out when decided by a fusion center, or upon requests from underwater nodes. The fusion center is responsible for scheduling the localization packet transmission of the anchors where each packet has a duration of  $t_p$ . Beside the localization data, other information can be encapsulated in the localization packets. Under this condition the packet size of each anchor can be different. Despite the fact

that this condition can be included in our problem, we do not consider that in this paper. The underwater sensor nodes receive the transmitted packets, and use them for self-localization. In a 3D environment, each node requires at least 4 (3 if the depth is known [94]) localization packets for self-localization. The localization accuracy and localization coverage depends on the density of the anchors per squared meter. If each sensor node is located under the coverage of four or more anchors, and receives the packets correctly, it can localize itself (full coverage). Still, the localization accuracy depends on the orientation of the anchors, and the position of the sensor node. The goal is to minimize the localization time, and to avoid any possible collision in the packet reception at all underwater sensor nodes. In order to accomplish this task, the fusion center gives each anchor  $i$  a waiting time  $w_i$  before it starts its packet transmission. In a multi-channel<sup>2</sup> scenario with  $M$  subchannels, the fusion center also determines which subchannel  $m_i \in 1, 2, \dots, M$  each anchor  $i$  has to transmit in.

So the problem we have to solve is to minimize the maximum waiting time, thereby avoiding any possible packet collision. To solve that problem, we have to analyze how collisions occur in the network. A collision will happen, if two or more transmitted packets overlap with each other at a sensor node. But since the sensor nodes can be located anywhere in the medium, there may be a collision if the transmitted packets from two anchors collide anywhere inside the intersection of the transmission ranges of these two anchors. Hence, as shown in Fig. 5.2, even if two anchor nodes are not located within their acoustic communication ranges they may cause a collision in the network. Briefly stated, two anchor nodes with a mutual distance smaller than twice the maximum transmission range are *collision-risk neighbors*, and therefore, they may cause collisions. The symbol  $\leftrightarrow$  is defined to show that two nodes are collision-risk, i.e.,  $i \leftrightarrow j$  means that anchors  $i$  and  $j$  are collision-risk anchors. In addition  $\Leftrightarrow$  is used to show that the anchors are within the communication range of each other. If  $i \Leftrightarrow j$  they are also collision-risk anchors.

In order to eliminate the collision problem, we introduce the concept of *collision-free anchors*, and we will show how waiting times can be modified to make anchors collision-free in order to eliminate collisions at the sensor nodes.

---

<sup>2</sup>Note that, this algorithm is independent of the technique (i.e., FDMA, CDMA, etc.) that is used to increase the number of channels as long as the created sub-channels are orthogonal to each other.

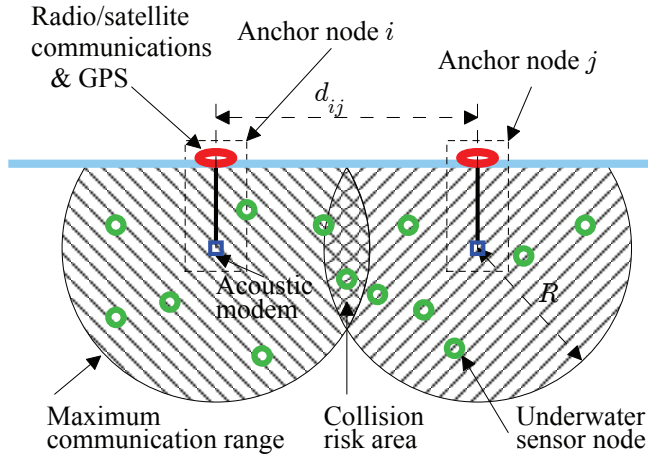


Figure 5.2: Example of two collision-risk anchors.

### 5.2.1 Collision-free anchors

Imagine that there are two anchors, namely  $i$  and  $j$ , at distance  $d_{ij}$  that are going to transmit their packets in the respective subchannels  $m_i$  and  $m_j$  and, with respective waiting times  $w_i$  and  $w_j$  where  $w_i > w_j$ . We then want to find out whether under these conditions the two anchor nodes are collision-free. Below, we define a few conditions that will help us to analyze this problem.

Condition 1: If anchor  $i$  and  $j$  transmit their packets at different subchannels, i.e.,  $m_i \neq m_j$ , they are collision-free anchors.

Condition 2: When the mutual distance between the two anchors is larger than  $2R$ , their transmission packets never collide for any pair of waiting times, because their communication ranges have no intersection. We call such two anchors strictly distance-related collision-free anchors.

Condition 3: Assume that the sound speed in the underwater medium is  $c$ . If the difference between the two waiting times is greater than  $\frac{R}{c} + t_p$ , the transmitted packets of these nodes will never collide with each other for any mutual distance. We call such two anchors strictly time-related collision-free anchors.

Condition 4: If anchors  $i$  and  $j$  transmit in the same channel, they are collision-free anchors if  $w_i - w_j > \frac{2R - d_{ij}}{c} + t_p$  as shown in Fig. 5.3 for the minimum value of  $w_i - w_j$ . It can be observed that the crossing area is swept by the first, and the second anchor without any collision. This condition is useful when  $d_{ij} > R$ , otherwise, the term  $\frac{2R - d_{ij}}{c} + t_p$  is greater than  $\frac{R}{c} + t_p$ , and Condition 3 covers this

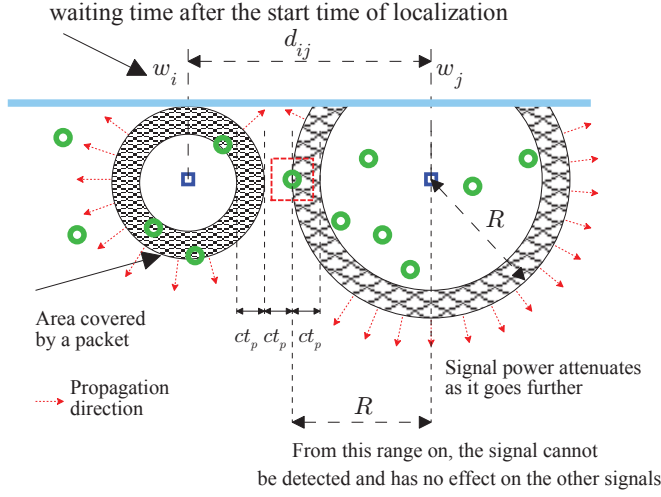


Figure 5.3: Snapshot of the packet transmissions of two collision-free anchors at time  $t = w_j + \frac{R}{c}$ , and at distance  $d_{ij}$  where  $R < d_{ij} < 2R$ . The hatched parts show the area where the localization packets reside. Given  $w_j$ , anchor  $i$  has transmitted its packet at minimum  $w_i$  according to (5.1). At time  $t = w_j + t_p + \frac{R}{c}$ , the effect of the packet transmitted from  $j$ -th anchor vanishes and the sensor node which is located on the border (inside the red box in the figure) starts receiving the transmitted packet from the  $i$ -th anchor.

case. We can deduce that if we have  $R < d_{ij} < 2R$ , and  $w_j$  is already set, then the minimum value for  $w_i$  that makes these anchors collision-free can be obtained by

$$w_{i,\min} = w_j + \frac{2R - d_{ij}}{c} + t_p. \quad (5.1)$$

In general, when  $w_i$  is not necessarily greater than  $w_j$ , for a collision-free transmission of the localization packets when the waiting time of anchor  $j$  is already set,  $w_i$  has to meet the following inequality:

$$|w_i - w_j| \geq \frac{2R - d_{ij}}{c} + t_p, \quad (5.2)$$

Condition 5: If anchors  $i$  and  $j$  transmit in the same channel, they are collision-free anchors if  $w_i - w_j > t_p + \frac{d_{ij}}{c}$  as shown in Fig. 5.4 for the minimum value of  $w_i - w_j$ . This condition is useful if  $d_{ij} < R$ , otherwise, like Condition 4, it can be represented by Condition 3. In other words, if we have  $d_{ij} < R$ , and the waiting

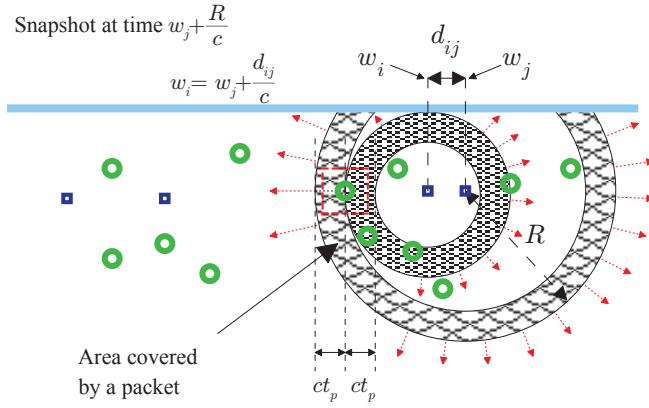


Figure 5.4: Snapshot of the packet transmissions of two collision-free anchors located  $d_{ij} < R$  meters away from each other at time  $t = w_j + \frac{R}{c}$ . Given  $w_j$ , anchor  $i$  transmits at the minimum waiting time according to (5.3). The transmitted packet from the  $i$ -th anchor follows that of the  $j$ -th anchor, and does not overlap with that.

time of anchor  $j$  is already set to  $w_j$ , then the minimum value for  $w_i$ , that makes these two anchors collision-free can be obtained as

$$w_{i,\min} = w_j + t_p + \frac{d_{ij}}{c}. \quad (5.3)$$

As before, when  $w_i$  is not necessarily greater than  $w_j$ , for a collision-free transmission of the localization packets when the waiting time of anchor  $j$  is already set,  $w_i$  has to be outside the following boundaries:

$$|w_i - w_j| \geq \frac{d_{ij}}{c} + t_p. \quad (5.4)$$

This condition is similar to what is explained in [115], [116].

### 5.2.2 L-MAC: Localization packet scheduling

Now that the concept of collision-free packet transmission has been clarified, we can formulate the optimization problem as

$$\min_{\{w_i\}, \{m_i\}} \max \{w_i\} \text{ s.t.}$$

$$w_i \geq 0, \text{ for } i \in \{1, 2, \dots, N\} \quad (5.5a)$$

$$|w_i - w_j| > t_p + \frac{\check{d}_{ij}}{c} \text{ if } m_i = m_j \text{ and } i \leftrightarrow j. \quad (5.5b)$$

where

$$\check{d}_{ij} = \min(d_{ij}, 2R - d_{ij}), \quad (5.6)$$

$w_i \geq 0$  states that we cannot have a packet transmission at negative times, and Conditions 1 to 5 are merged into (5.5b).

From Conditions 4 and 5, it can be observed that in a collision-free packet transmission, setting the waiting time of an anchor imposes limitations on the waiting times of its collision-risk neighbors. These limitations not only relate to the time after the packet transmission of the considered anchor, but also to the time before its packet transmission. This is really important for finding the optimal solution of (5.5). In the next subsection, we show how the problem of scheduling can be formulated for a broadcast scenario.

### 5.2.3 B-MAC: Broadcasting packet scheduling

In a broadcast scenario, the transmitted packet from an anchor has to be received not only by the sensor nodes in its communication range, but also by the anchors which are within the communication range of this anchor. Since, simultaneous reception and transmission is not allowed in half-duplex underwater nodes, to receive the transmitted packet another condition has to be added to (5.5), namely

$$|w_i - w_j| > t_p + \frac{d_{ij}}{c} \text{ if } i \leftrightarrow j, \text{ for any } m_i \text{ and } m_j. \quad (5.7)$$

In an actual scenario, the packet exchanges between anchors may be used for sound-speed estimation, or to check the functioning of their acoustic modems. If the network supports packet exchanges between neighboring anchors, we refer to the scheduling protocol as B-MAC.

### 5.2.4 Dynamic multi-channel packet scheduling

In dynamic multi-channel packet scheduling, the fusion center can decide to use the whole operating bandwidth as a single channel for packet transmission, or to divide it into several subchannels. Under this condition, the number of subchannels  $M$  would be a variable, and this can be include in (5.5). Note that since the number of

bits in a localization packet is constant, the packet duration varies with the number of subchannels as

$$t_p^M = \frac{M}{1 - \alpha_M} t_p^1, \quad (5.8)$$

where  $t_p^M$  is the packet duration if  $M$  subchannels are employed,  $t_p^1$  is the packet duration for the single-channel case (whole bandwidth is used), and  $\alpha_M$  is the penalty that the system suffers from splitting the channel into  $M$  subchannels [122]. In underwater acoustic communication, the signals which are transmitted at higher frequencies are attenuated more. Therefore, if the anchors transmit with the same power in each subchannel, the communication range of each subchannel would be different. However, with a simple power allocation strategy, they can maintain a similar communication range for each subchannel. The localization algorithms are usually based on range estimation which can be obtained via time of flight estimation between the sensor node and an anchor. The variance of error in time of flight (ToF) estimation grows linearly with the inverse of the received signal bandwidth [123] and the time of signal observation. Since the time of signal observation (packet duration) increases as the channel is splitting (signal bandwidth reduction), the accuracy of the ToF estimation does not change. However, we assume that the system cannot generate more than  $M_{\max}$  subchannels. In general, for dynamic multi-channel B-MAC we can formulate the problem as

$$\min_{M, \{m_i\}, \{w_i\}} \max \{w_i + t_p^M\}, \text{ s.t.}$$

$$w_i \geq 0, \text{ for } i \in \{1, 2, \dots, N\},$$

$$M_{\max} \geq M \geq 1,$$

$$|w_i - w_j| > t_p^M + \frac{\check{d}_{ij}}{c} \text{ if } i \leftrightarrow j \text{ and } m_i = m_j, \quad (5.9a)$$

$$|w_i - w_j| > t_p^M + \frac{d_{ij}}{c} \text{ if } i \nleftrightarrow j, \text{ for any } m_i \text{ and } m_j. \quad (5.9b)$$

where  $\check{d}_{ij}$  has been defined in (5.6). Note that for L-MAC condition (5.9b) is not required.

### 5.2.5 Problem formulation in a TDMA system

In a TDMA system, if the time duration of each slot is set to  $t_s = \frac{R}{c} + t_p$ , and we have  $\frac{R}{c} \rightarrow 0$ , then the optimization function in (5.5) is equivalent to minimizing the number of slots under a collision-free transmission of localization packets. With the above definitions, this problem can be modeled as TDMA broadcast scheduling which is well-studied in [124]. As mentioned in [124], scheduling the packets in the minimum number of slots is an NP-hard problem.

TDMA broadcast scheduling leads to the optimal solution for minimizing the localization task if  $\frac{R}{c} \rightarrow 0$ . For cases where  $\frac{R}{c} \neq 0$ , this solution is not optimal, but it can still be hired for localization packet scheduling, as we will discuss in Section 5.3.1. We label optimal and suboptimal algorithms that try to minimize the number of slots for the TDMA broadcast scheduling problem as TDMA-based algorithms. In WSNs, the wave speed is the speed of light and the propagation delay is negligible, so slotted algorithms are quite acceptable. In contrast, the propagation delays in underwater communications are large, and sometimes even greater than the packet length, especially for localization packets. In that case, TDMA-based algorithms are inefficient, and other schemes have to be devised.

### 5.2.6 Practical issues for the problem

Albeit that this paper tries to take advantage of the long propagation delay of underwater acoustic communications to minimize the time duration of the localization task, there are some other issues that would adversely affect the time duration of the localization task. In this subsection, we list these challenges, and we suggest how they can be considered in our optimization problem.

#### Problem formulation considering full coverage

To be able to localize its position in a three dimensional (3D) environment, each node requires at least four time of flight (ToF) measurements to the known anchors. If each sensor node is equipped with a pressure sensor, then it can measure its depth, and with this information, it only requires three ToF measurements to find its location. Three-dimensional localization based on the surface-located anchors and the depth information of the sensor nodes has been analyzed in [94] and [72]. Although the more ToFs a node acquires, the better the localization accuracy, one may say that for the localization task, each point in the operational area has to be covered by at least four (without pressure sensor) or three (with pressure sensor)

anchors. Hence, it is not needed to include all the anchors in the localization task if this condition is satisfied with a smaller number of anchors. In this situation, before running the MAC protocol, we can eliminate those unnecessary anchors and reduce the time of the localization task. This can also be taken into account in the optimization function (5.5) if the map of the area is known. However, in practical situations we are interested in utilizing all the available information from the anchors and perform the localization task with minimum error. Furthermore, when unwanted phenomena such as fading exist, the more anchors we employ in the network, the possibility of a full coverage of the operating area increases.

### **Multi-path**

Underwater acoustic communications are subject to severe multipath. Multiple reflections from the surface, seabed, other layers of water or other objects located inside the water cause the transmitted signal to be received from different paths with different delays. If we assume that the maximum channel delay spread is  $\tau_{\max}$ , then due to the convolutional property, the time duration of the received packet at a sensor node increases by this maximum delay. Hence, only adopting the conditions of Subsection 5.2.1, the received packets from two subsequent collision-risk anchors might collide in some parts of the operating area. This can simply be avoided by adding a guard time of length  $\tau_{\max}$  to the end of the packet. According to experimental data [125], the value of the maximum channel delay spread or consequently the value of the required guard time depends on the system parameters (bandwidth, operating frequency, transmitted power), location of the reflectors (shallow or deep water), depth of the receiving and transmitting nodes, and the distance between the transmitter and the receiver, and it may vary from 10 ms to 15 ms or more (typically between 10 ms to 25 ms).

### **Sound speed variations**

The sound speed in an underwater medium is not constant but varies with temperature, pressure, and salinity [73]. This causes a transmitted signal not to travel along a straight line as it propagates. On the other hand, when a wave travels along a curved path to the destination, which is longer than a straight path, it may still be faster when the sound speed along the curved path is sufficiently higher. This upsets the assumption of spherical wave propagation in an underwater medium, and it cannot be stated that the maximum propagation range in each direction is  $R$  meters. This issue can easily be handled by setting  $R$  to the maximum

value that a wave can propagate in different directions. The typical range of the underwater sound speed is from 1480 to 1520 m/s. In [97] and [89] it is shown that the influence of the sound speed on signal propagation becomes more severe if the horizontal distance between the transmitter and the receiver increases. Although considering a worst case radius  $R^3$  resolves this issue, the effect of a varying sound speed is negligible for short distances, particularly for just a few kilometers.

### Interference in packet reception

The maximum transmission range of an anchor can be defined as the distance at which the receiver is not able to decode the data anymore, for instance because of the sensitivity level of the receiver, or the low signal strength in comparison with the in-band ambient noise power. However, even if the maximum transmission range is limited due to the low signal to noise ratio, interference from other transmitters may still corrupt the packet reception at a sensor node. Interference occurs if two or more transmitted signals sweep the same point at the same time which may happen even for ranges beyond the maximum transmission range of an anchor. This explanation implies that two anchors with a mutual distance  $d_{ij} > 2R$  are not necessarily collision-free, and the probability of a collision due to interference is not zero. In order to include this phenomenon in our formulation, we define a maximum interference range,  $R_I$ , which is the distance to an anchor beyond which a sensor node does not experience interference from that anchor. This can be considered in our problem formulation (5.5) by substituting  $R$  with  $R_I$ .

### Moving anchors

In actual scenarios, the assumption of fixed anchors may not hold. The anchors may drift due to the waves in windy weather, move with water currents, or follow a predetermined trajectory. When the velocity vector of each anchor for each time is known, we can compute the relative distance between the anchors  $i$  and  $j$  in time as  $d_{ij}(t)$ , and as a result, the effect of network movement can be included in the optimization function when the future positions of the anchors are known. However, it is hard to take a random movement of the anchors into account, unless we predict them [126] which for a long time duration is not practical. If we assume that the maximum displacement from the anchor's original position during the

---

<sup>3</sup>Worst case radius means the largest distance from the transmitter, at which a signal can be detected by a sensor node. Note that the propagation model for variable sound speed is not spherical anymore.

localization task is  $d_{\text{drift}}$ , then we can add a guard time at the end of the packet to remove any possible collision. The value of this guard time can be computed as  $\frac{2d_{\text{drift}}}{c}$ . Note that, under the condition that a guard time,  $t_g$ , is added to the actual packet, the packet length in scheduling algorithms (see (5.8)) has to be modified to

$$t_p^M = \frac{M}{1 - \alpha_M} t_p^{\text{ng}} + t_g, \quad (5.10)$$

where  $t_p^{\text{ng}}$  is the packet length in single-channel scheduling approaches without considering the guard interval.

### Quantized waiting time

In practice, the fusion center quantizes the waiting-times before transferring this information to the anchors.<sup>4</sup> That would affect the precision of the localization task. In this work, we assume that enough bits are allocated to the waiting-times, and no quantization error exists.

## 5.3 Optimal solution

In this section, we first show how the optimal solution for the single-channel slotted scenario can be obtained, and based on that, we explain how this solution can be extended to the multi-channel problem and to finding the optimal solution of our problem (5.5).

### 5.3.1 Optimal solution in single-channel scenario

#### TDMA-based scheduling

As stated before, TDMA-based scheduling based on the definition of strictly distance-related collision-free anchors, and strictly time-related collision-free anchors is an NP-hard (non-deterministic polynomial-time hard) problem. The TDMA-based approach can be employed in UASNs where no information about the anchors' relative positions is available.

The optimal solution (which may not be unique) belongs to a set of  $N!$  possible candidate anchor sequences, and can be obtained by an exhaustive search. Based on a given anchor sequence, we start with the first anchor, and allocate it to the first time slot. Then, we move to the next anchor, and allocate it to the

<sup>4</sup>The calculated scheduling times have to be quantized and converted to bits and then transmitted.

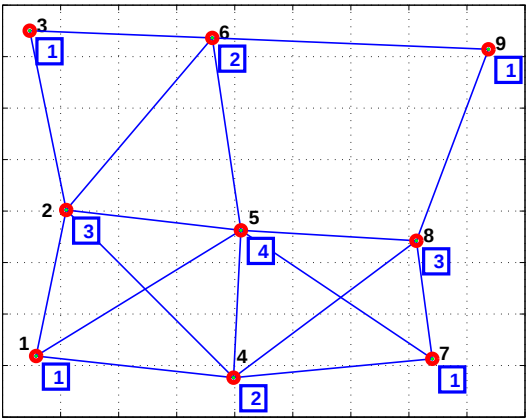


Figure 5.5: Network graph for a TDMA-based scheduling problem. Vertices represent the anchors where each anchor is labeled by its index, and an edge connecting two anchors shows there is a collision risk. The boxed numbers represent the time slot that each anchor can use to transmit its localization packet.

earliest possible time slot that causes no collision with considering the previously scheduled anchors. The same procedure continues until the last anchor gets scheduled. At the end, we count the number of used slots, and among all possible  $N!$  anchor sequences we choose the sequence with the lowest number of slots. An example of a possible solution for the network in Fig. 5.5 is shown in Fig. 5.6. The vertices in this graph represent the anchors, and each edge connecting two vertices indicates that there is a collision risk.

|              |      |      |      |      |      |      |      |      |      |
|--------------|------|------|------|------|------|------|------|------|------|
| anchor index | 1    | 3    | 7    | 9    | 4    | 6    | 2    | 8    | 5    |
| time slots   | 1    | 1    | 1    | 1    | 2    | 2    | 3    | 3    | 4    |
| waiting time | 0.00 | 0.00 | 0.00 | 0.00 | 1.05 | 1.05 | 2.10 | 2.10 | 3.15 |

Figure 5.6: Optimum solution when a TDMA-based algorithm is used. Note that the anchors 1, 3, 7, and 9 are allocated the same time slot. Anchor 4 cannot transmit in time slot 1 because it is a neighbor of some previously scheduled anchors, namely 1 and 7.

## Distance-aware scheduling

To find the optimal solution in a distance-aware network, we follow the same procedure as for TDMA-based scheduling. Here, we again start from the fact that the optimal solution (which is not unique) belongs to at most  $N!$  sequences of anchor indices. For each anchor sequence, the anchor that appears earlier has to transmit its packet sooner than the ones which appear later. Conditioned on a given anchor sequence, the minimum duration of the task,  $w_{\max}$ , can simply be computed based on Conditions 2, 4, and 5 of Section 5.2.1. In this procedure, the first anchor is assigned to transmit its packet first. Then, the limitations on the transmission time of the other anchors are computed, and the second anchor computes the earliest available time (which is greater than or equal to the transmission time of the previously scheduled anchors) it can transmit without causing collisions. Finally, by comparing the maximum waiting times ( $w_{\max}$ ) of all  $N!$  anchor sequences, we choose the sequence of anchor indices which has the minimum  $w_{\max}$ .

For instance, consider the network graph depicted in Fig. 5.7, where the maximum transmission range,  $R$ , is equal to  $c$  meters (normalized to one only in the figure), and the packet length is 0.05 s. In this graph, each edge weight with white background color shows the normalized distance between the two collision-risk anchors. The optimal anchor sequence and the related waiting times have been computed by the explained procedure, and the result is shown in Fig. 5.8.

### 5.3.2 Optimal solution in a multi-channel scenario

The optimal solution of a TDMA-based system in a multi-channel scenario can be obtained similarly by examining all possible ways that the time slots can be allocated to the nodes. It can be shown that when  $M < N$ , the number of possible solutions is smaller than  $\frac{(MN)!}{(MN-N)!}^5$ , and greater than  $N!$ , which makes finding the optimal solution a combinatorial problem. This is also true for the distance-aware algorithms. Given the order of packet transmission and the subchannels that they are going to use, we can find the minimum waiting-times of all nodes as explained in the single-channel case. After comparing the results of all possible solutions, we select the one which leads to the lowest number of time slots, or the lowest maximum waiting-time.

---

<sup>5</sup>This is equivalent to the selection of  $N$  candidates from  $MN$  items where ordering is important. This amount can hugely be reduced via heuristic approaches.

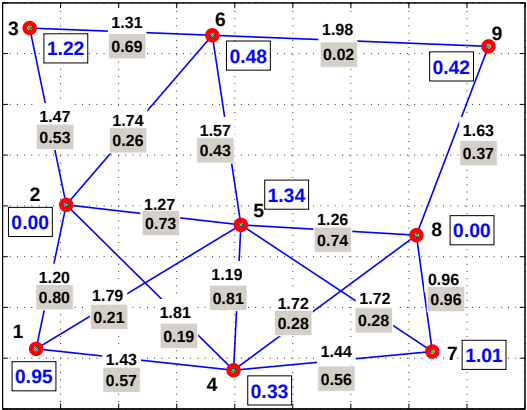


Figure 5.7: Network graph for the distance-aware scheduling problem. The edge weights with white background color represent the normalized distances between collision-risk anchors, and the edge weights with gray background color show the normalized modified distances,  $\min\{d_{ij}, 2R - d_{ij}\}$ , between collision-risk anchors. The boxed numbers display the waiting times of the anchors.

|              |      |      |      |      |      |      |      |      |      |
|--------------|------|------|------|------|------|------|------|------|------|
| anchor index | 2    | 8    | 4    | 9    | 6    | 1    | 7    | 3    | 5    |
| waiting time | 0.00 | 0.00 | 0.33 | 0.42 | 0.48 | 0.95 | 1.01 | 1.22 | 1.34 |

Figure 5.8: Optimum solution when the optimal distance-aware scheduling algorithm is used.

## 5.4 Mixed-integer linear programming

In this section we show how the optimization problem of (5.5) (and similarly (5.9)) can be modeled as a standard mixed-integer linear programming (MILP) problem. The basic form of a MILP problem is given by

$$\begin{aligned} \min \mathbf{d}^T \mathbf{x}, \text{ s.t.} \\ \mathbf{x}_L \leq \mathbf{x} \leq \mathbf{x}_U \\ \mathbf{b}_L \leq \mathbf{A}\mathbf{x} \leq \mathbf{b}_U \\ x_i \text{ integer for } i \in I \end{aligned} \quad (5.11)$$

where  $\mathbf{d} \in \mathbb{R}^{P \times 1}$  collects the linear objective cost function coefficients,  $P$  is the number of variables in the design parameter  $\mathbf{x}$ ,  $\mathbf{x}_L, \mathbf{x}_U \in \mathbb{R}^{P \times 1}$  are respectively the lower and upper bound on the design parameter,  $\mathbf{A} \in \mathbb{R}^{M \times P}$  is a linear constraint matrix stacking the linear constraints,  $\mathbf{b}_L, \mathbf{b}_U \in \mathbb{R}^{M \times 1}$  are respectively the lower and upper bound on the constraints, and  $I$  is a subset of  $\{1, 2, \dots, M\}$ , which contains the indices of integer variables.

It is well-known that the min max problem can be transformed into a linear one by introducing  $N$  additional constraints as

$$\min z, \text{ s.t. } w_i \leq z \text{ for } i \in \{1, 2, \dots, N\}. \quad (5.12)$$

In addition, by defining of a new Boolean variable,  $\delta_{ij}$ , (5.5b) in a single-channel scenario can be modeled by 2 linear inequalities as

$$-w_i + w_j - Q_1 \delta_{ij} \leq -t_p - \frac{\check{d}_{ij}}{c} \quad (5.13a)$$

$$+w_i - w_j + Q_1 \delta_{ij} \leq -t_p - \frac{\check{d}_{ij}}{c} + Q_1, \quad (5.13b)$$

where  $Q_1$  is a constant which has to be greater than  $\max \left( |w_i - w_j| + \frac{\check{d}_{ij}}{c} + t_p \right)$ , for  $i, j \in \{1, 2, \dots, N\}$  (or in the worst case  $Q_1 > N \left( \frac{D}{c} + t_p \right)$ ). This means that if  $\delta_{ij} = 0$ , then (5.13a) has to be satisfied and (5.13b) is always true, and vice versa. Under this condition the MILP problem has  $N + 1$  continuous variables ( $z$  and  $w_i, i \in \{1, 2, \dots, N\}$ ),  $N_c$  Boolean variables ( $\delta_{ij}$ ), and  $2N_c + N$  inequality constraints (see (5.12) and (5.13)), where  $N_c$  is the number of collision-risk connections.

In the multi-channel L-MAC scenario, (5.9a) can be modeled as

$$|w_i - w_j| + Q_2|m_i - m_j| > t_p^M + \frac{\check{d}_{ij}}{c}, \quad (5.14)$$

where  $Q_2 > \max\left(\frac{\check{d}_{ij}}{c} + t_p^M\right)$  for  $i, j \in \{1, 2, \dots, N\}$  is a constant, and can be considered as  $Q_2 > \frac{D}{c} + t_p^M$ . This means that if  $m_i \neq m_j$  then (5.14) is always true and  $w_i$  is independent of  $w_j$ . The non-linear inequality of (5.14) can be expanded into four non-linear inequalities as

$$\begin{aligned} -w_i + w_j - Q_2(+m_i - m_j) - Q_3(0 + \delta_{ij} + \gamma_{ij}) &\leq -\Delta_{ij}, \\ -w_i + w_j - Q_2(-m_i + m_j) - Q_3(1 - \delta_{ij} + \gamma_{ij}) &\leq -\Delta_{ij}, \\ +w_i - w_j - Q_2(+m_i - m_j) - Q_3(1 + \delta_{ij} - \gamma_{ij}) &\leq -\Delta_{ij}, \\ +w_i - w_j - Q_2(-m_i + m_j) - Q_3(2 - \delta_{ij} - \gamma_{ij}) &\leq -\Delta_{ij}, \end{aligned}$$

where  $\Delta_{ij} = t_p^M + \frac{\check{d}_{ij}}{c}$ ,  $\gamma_{ij}, \delta_{ij} \in \{0, 1\}$ ,  $m_i \in \{1, 2, \dots, M\}$  for  $i \in \{1, 2, \dots, N\}$ , and

$$Q_3 > \max(|w_i - w_j| + Q_2|m_i - m_j| + \Delta_{ij}),$$

for  $i, j \in \{1, 2, \dots, N\}$  is a constant, and can be considered as  $Q_3 > N\left(\frac{D}{c} + t_p\right) + (M - 1)Q_2$ .

The multi-channel B-MAC optimization problem can similarly be modeled as a MILP problem. There, for each pair of anchors  $i$  and  $j$ , if  $i \leftrightarrow j$  we have 4 inequalities like what extracted for (5.14), and if  $i \Leftrightarrow j$  we have two inequalities like (5.13). Note that the constraints we have in a single-channel L-MAC include the ones defined for the single-channel B-MAC, and therefore there is no difference between them in a single-channel scenario.

## 5.5 Proposed algorithms

The complexity of the optimal solution (without any heuristic approach) is equal to or greater than  $N!$ , which makes it impossible to be used when the number of anchors or subchannels are large. In this section, we propose two heuristic algorithms with a smaller complexity (of order  $N$  and  $N^2$ ) that can be adopted for practical applications. In the networking concept, the heuristic algorithms are similar to the standard greedy nearest neighbor with slight modifications. In the numerical section, we show that these suboptimal algorithms can perform near

optimal.

The first suboptimal algorithm is based on a greedy approach, and its steps are shown in Algorithm 4. In the initial phase, the waiting times of the transmitting nodes are set to zero, and a buffer of size  $N \times M$  is defined to store the limitations on the waiting time of the nodes in all subchannels. The algorithm starts with scheduling a pre-set arbitrary anchor (for instance the  $I$ -th anchor in which case we refer to the algorithm as the  $I$ -th starter or  $IS$ ) or a random anchor (RS), and assign the first subchannel to this node. Therefore, the waiting time of this anchor is fixed to zero, and it will transmit in the first subchannel. When the waiting time of an anchor gets fixed, it will be removed from the scheduling task. Based on this fixed waiting time, the collision-risk neighbors of the selected anchor are detected, and their corresponding waiting times are modified in such a way that no collisions will occur in the network (collision-free anchors based on Conditions 1 to 5). Then, from the unscheduled anchors, the one which has the lowest waiting time in all possible subchannels will be selected, and the above steps will be repeated until the waiting times of all anchors get fixed. It may happen that there are two or more anchors with the same minimal waiting time. In this case, we select the one which has the lowest index as well.

As can be seen from Algorithm 4, Condition 3 is not included. Condition 3 states that if  $|w_i - w_j|$  is greater than  $\frac{R}{c} + t_p$ , the two anchors are collision-free. Since in each step of the algorithm we choose the anchor with the minimal waiting time, it never happens that  $w_i < w_j$ , and we only have to check the condition  $w_i - w_j > \frac{R}{c} + t_p$ . If it is met, then the two anchors are collision-free and no modification on  $w_i$  is required. This condition is hidden behind the max operation of the algorithm. If this condition holds, the algorithm does not modify  $w_i$  which means that the algorithm excludes the corresponding anchor from a possible waiting time modification.

The best starter algorithm (BS) is an extension of  $IS$ . In BS, we run  $IS$  for all the anchors ( $I = 1$  to  $N$ ), and select the one (the best starter) which results in the minimal total scheduling time.

For the dynamic multi-channel (DMC) packet scheduling, we run the algorithm for different number of channels  $M \in \{1, \dots, M_{\max}\}$ , and select the  $M$  which results in the lowest scheduling time. Note that for DMC, for each value of  $M$ , the packet duration,  $t_p$ , has to be modified according to (5.8) or (5.10). The DMC can be used for both RS ( $IS$ ) and BS.

**Algorithm 4** *IS*: Start from the  $I$ -th anchor

Input: distances between collision-risk nodes,  $d_{ij}$ ,  
 maximum transmission range,  $R$ ,  
 Number of subchannels,  $M$ ,  
 Packet duration,  $t_p$ ,

Output: waiting times before packet transmission,  $w_k$  for  $k = 1, 2, \dots, K$ ,  
 channel in which each node has to transmit its packet,  $m_k$ ,  
 Task duration,  $T_{\text{broadcast}}$ .

Set all the waiting times to zero:  $w_k = 0$ , for  $k = 1, 2, \dots, K$ ,

Set all entries of  $\mathbf{W}_{K \times M}$  to zero.

Set  $m = 1$ .

Set  $\Omega = \{1, 2, \dots, K\}$ .

Start with the pre-defined anchor index  $I, j = I$ ,

**for**  $k = 2$  to  $K - 1$  **do**

Remove  $j$ -th anchor from the network:  $\Omega = \Omega - \{j\}$

Find the collision-risk neighbors of the  $j$ -th anchor, and modify their waiting time to eliminate possible collisions:

**for**  $i \in \Omega$  **do**

**if**  $d_{ij} \leq 2D$  **then**

**if** L-MAC **then**

**if** TDMA-based **then**

$[\mathbf{W}]_{i,m} = \max([\mathbf{W}]_{j,m} + t_p + \frac{D}{c}, [\mathbf{W}]_{i,m})$

**else**

$[\mathbf{W}]_{i,m} = \max([\mathbf{W}]_{j,m} + t_p + \frac{\tilde{d}_{ij}}{c}, [\mathbf{W}]_{i,m})$

**end if**

**else if** B-MAC **then**

**if**  $d_{ij} \leq D$  and  $\frac{d_{ij}}{c} - |[\mathbf{W}]_{j,m} - [\mathbf{W}]_{i,m}| < t_p$  **then**

**for**  $p = 1$  to  $M$  **do**

$[\mathbf{W}]_{i,p} = \max([\mathbf{W}]_{j,p} + t_p + \frac{\tilde{d}_{ij}}{c}, [\mathbf{W}]_{i,p})$

**end for**

**else**

$[\mathbf{W}]_{i,m} = \max([\mathbf{W}]_{j,m} + t_p + \frac{\tilde{d}_{ij}}{c}, [\mathbf{W}]_{i,m})$

**end if**

**end if**

**end if**

**end for**

Select the anchor with the minimum waiting time:

$[j, m] = \arg \min_{i \in \Omega, m \in \{1 \text{ to } M\}} [\mathbf{W}]_{i,m}$

**end for**

Compute the waiting times of each anchor and its channel

**for**  $k = 1$  to  $K$  **do**

$w_k = \min_{m=1 \text{ to } M} [\mathbf{W}]_{k,m}$

$m_k = \arg \min_{m=1 \text{ to } M} [\mathbf{W}]_{k,m}$

**end for**

Compute the broadcasting task duration:  $T_{\text{broadcast}} = \max_{i=1 \text{ to } N} w_i + t_p$

## 5.6 Numerical results

In this section, we evaluate the performance of the proposed low-complexity algorithms and compare them with MILP and optimal solutions. In order to find the solution of the MILP problem we have used the TOMLAB/CPLEX [127] and MATLAB MILP solver with their default setting. Moreover, we also compare their performance with the traditional TDMA-based ones.

Unless otherwise stated, the simulation set up is as follows. For the computation of each point in the following figures, we average the solution over  $10^3$  ( $10^2$  for MILP) independent Monte Carlo runs. The localization packet length is  $t_p = 150$  ms (150 bits for an acoustic modem with a data rate of 1kbps), which is long enough to convey the information about the anchor's ID, position and time of transmission. The maximum transmission range of each anchor is assumed to be  $2c$  m (3 km). The positions of the anchors are assumed to be uniformly distributed at random over a squared area with dimensions  $d_x = d_y = 5c$  m (7.5 km). The system can split the existing channel to at most  $M_{\max} = 3$  subchannels, and the penalty in channel splitting is formulated as  $\alpha_M = 0.1(M - 1)$ .

In our first simulation, we consider a single-channel (SC) network in which each pair of anchors are within the communication range of each other (fully connected network). The maximum transmission range of the anchors for this simulation is set to  $5c\sqrt{2}$  m. We compare the performance of different position-aware algorithms (L-MAC-SC-RS, L-MAC-SC-BS, L-MAC-SC-MILP, and L-MAC-SC-Optimal)<sup>6</sup> with an algorithm where no position information is assumed (position-unaware). In the position-unaware algorithm the first anchor transmits its packet, and after the complete reception of the packet, the second anchor starts its transmission. This continues until the last packet is transmitted from the  $N$ -th anchor. Here, the order of transmission is fixed, and no position information or a fusion center is required. Fig. 5.9 shows the performance of each algorithm for different numbers of anchors (in the x-axis). The y-axis shows the average time of the localization task, as defined by  $t_{\text{avg}} = \mathbb{E}[w_{\max} + t_p]$ . As Fig. 5.9 demonstrates, the increase in the number of anchors makes the duration of the localization task more lengthy. That is because a network with more anchors requires more packet transmissions. Another fact resulting from the figure is the performance superiority of position-aware algorithms in comparison with the position-unaware one. Not only do they perform much better than their opponents,

<sup>6</sup>Note that, SC means *single channel*, RS means *random starter*, BS means *best starter*, and DMC means *dynamic multi-channel*.

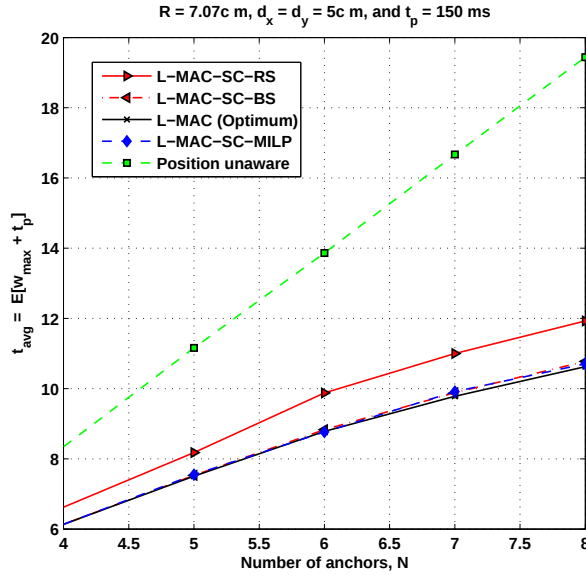


Figure 5.9: Average packet transmission time versus number of anchors.

their performances are also very close to each other and to the optimal solution. Therefore, when complexity is an issue in practical situations, L-MAC-RS can be adopted as the appropriate scheduling protocol for the localization task.

For the rest of the simulation results, the performance of the optimal solution is not computed because it takes a huge amount of time. Furthermore, the complexity of the TOMLAB/CPLEX increases greatly with the number of integer variables. It has been observed that the TOMLAB/CPLEX solution is highly close to the BS algorithms. Therefore, to reduce the simulation time, we have not included the results of the MILP problem when the network size is large.

The effect of the maximum transmission range on  $t_{avg}$ , where the dimension of the area is fixed, is depicted in Fig. 5.10. In this scenario, with an increase in  $R$ , the number of strictly distance-related collision-free anchors gets lower (see upper part of the figure), and as a result, the possibility of simultaneous packet transmission in the network decreases, and  $t_{avg}$  increases. This growth in  $t_{avg}$  for the distance-aware algorithms stops when the network is fully connected and  $\min\{d_{ij}, 2R - d_{ij}\} = d_{ij}$ . At this point, the performance of OCSMA is the same as that of L-MAC-SC, because in a fully connected network the L-MAC-SC does not support simultaneous transmissions (see Condition 5), and therefor anchors transmit one after each other (no simultaneous transmissions) similar tp OCSMA.

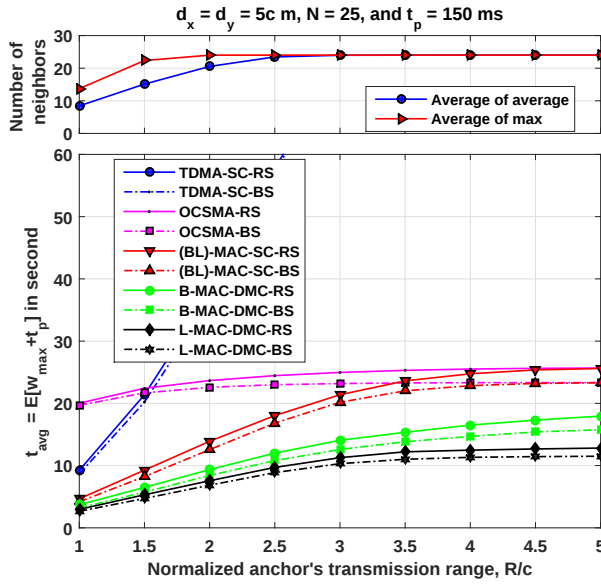


Figure 5.10: Average packet transmission time versus anchors' maximum transmission range.

In the TDMA-based algorithms, the average localization time increases because the time-slot length is proportional to  $\frac{R}{c}$ . The performances of the DMC algorithms are included in this graph as well. It can be observed that they work better than their single-channel counterparts. Furthermore, the performance of L-MAC-DMC is better than B-MAC-DMC, because in L-MAC the anchors are not interested in the packet reception from other anchors, and they experience less-strict limitations on their waiting times. Under the condition that a guard time has been added to the packet, the packet length can be obtained by (5.10), and the performance of the multi-channel schemes would be better than the ones depicted in Fig. 5.10 in comparison to the single-channel schemes.

In Fig. 5.11, the performance of the proposed algorithms versus network scalability is evaluated. For this simulation, as the dimension of the operating area increases, the number of anchor nodes increases too such that the average number of anchors per square meter is constant. Again, as the network gets larger, the probability that more nodes are strictly distance-related collision-free decreases (see upper part of the figure) and the nodes experience a larger waiting time. However, at a specific network size, the average number of collision-risk neighbors converges to a fixed value as depicted in the upper part of the figure, and as a result,

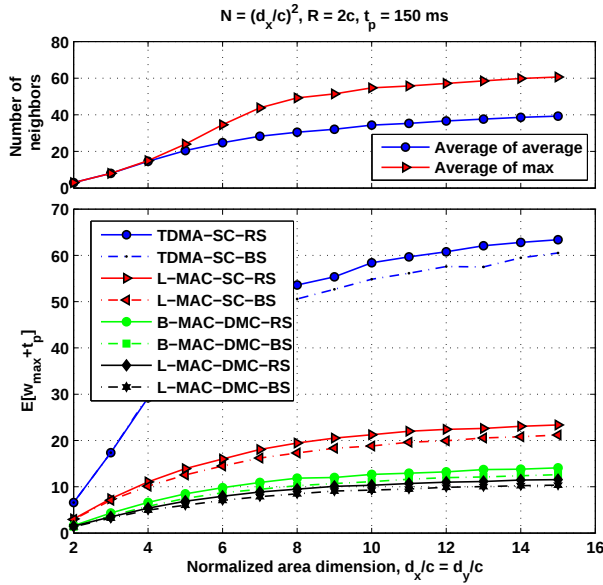


Figure 5.11: Performance of the algorithms versus network scalability.

the performance of both the TDMA-Based and the proposed algorithms saturate. Still we can see that the proposed algorithms perform better than TDMA-based one, and the ones which use DMC greatly reduce the average time that is required for localization.

A map of the waiting times for a specific single-channel network with a large number of anchors ( $N = 400$ ) is shown in Fig. 5.12. The colormap shows the range of waiting time values, the color behind each anchor node represents its waiting time, and the colors in between convey no particular meaning. It can be noticed that different disjoint clusters of the network transmit their localization packets simultaneously. This is the reason why the network scalability does not influence the performance of the proposed algorithms. On the other side, the performance of the algorithms is directly related to the average number of collision-risk neighbors around an anchor, and their average modified distances,  $\min\{d_{ij}, 2R - d_{ij}\}$ , (see Fig. 5.7).

In Fig. 5.13, as the dimension of the area increases, we increase the maximum transmission range of the anchors, but keep the number of anchors constant. It can be observed that, when  $d_x/R$  is constant, for large values of  $\frac{R}{c}$ ,  $t_{avg}$  is linearly related to  $R$  for all algorithms. On the contrary, when the value of  $\frac{R}{c}$  is small relative to the packet length, the performance of the

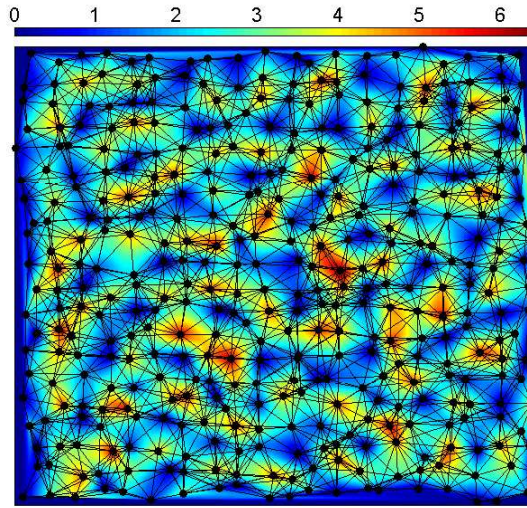


Figure 5.12: Waiting time map for a specific network of  $N=400$  anchors nodes,  $R = 1.1c$  m, and  $d_x = d_y = 20c$  m. The vertices show the anchors' locations and the edges show which anchors have a collision risk. The waiting times are computed based on the L-MAC-BS algorithm.

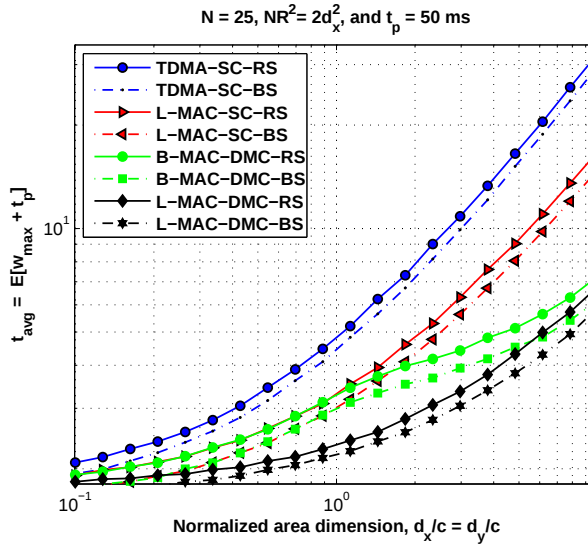


Figure 5.13: Average packet transmission time versus anchors density.

algorithms tends to be constant as a function of  $\frac{R}{c}$ . As also anticipated from the formulation of the objective function, as the ratio  $\frac{R}{c}$  becomes smaller and smaller, the performance of TDMA-based algorithms approaches the performance of distance-aware algorithms or equivalently the optimal solution. A similar analysis can be carried out when the packet size increases and other parameters are fixed. As the packet size gets larger, the required guard time,  $\frac{R}{c}$ , with respect to the slot duration, becomes negligible for TDMA-based algorithms and again both single-channel distance-aware and TDMA-based ones perform similar. This concept is illustrated in Fig. 5.14. In the reverse direction, when the ratio of the packet length and  $\frac{R}{c}$  approach zero, its value is not dominant anymore in the performance of the algorithms.

From Fig. 5.13 and Fig. 5.14, it can be seen that L-MAC-DMC always requires less localization time in comparison with its single-channel counterparts. As the number of subchannels increases the number of collision-risk anchors at the same subchannel decreases and the limitations on the waiting time of the anchors become less. In contrast, as the ratio of  $\frac{t_p c}{R}$  gets larger in B-MAC-DMC, the inequality (5.9b) becomes dominant, and because of the penalty we have in channel splitting in order to reduce the localization time the system will work with less subchannels. As the ratio of  $\frac{t_p c}{R}$  gets very large, the B-MAC-DMC performs the same as B-MAC-SC.

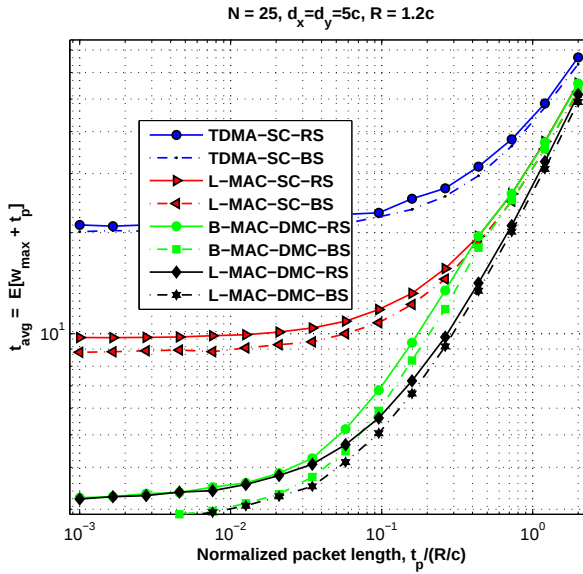


Figure 5.14: Average packet transmission time versus packet length.

In order to compare the performance of the algorithms with the one of MILP, we have considered a small network of 6 anchors depicted in Fig. 5.15 with different transmission ranges. The MATLAB MILP solver is used to find the solution. The localization time for each scheme is show in Table 5.1.

Table 5.1: Localization time vs. different transmission range and different algorithms. The  $^{(M)}$  shows the number of used subchannels in DMC.

| Algorithm      | $R = 2c$                | $R = 3c$                | $R = 4c$                |
|----------------|-------------------------|-------------------------|-------------------------|
| TDMA-SC-RS     | 8.7500 s                | 15.9000 s               | 20.9000 s               |
| TDMA-SC-BS     | 8.7500 s                | 15.9000 s               | 20.9000 s               |
| L-MAC-SC-RS    | 5.5239 s                | 9.5535 s                | 10.0881 s               |
| L-MAC-SC-BS    | 5.0113 s                | 9.4192 s                | 9.8322 s                |
| L-MAC-DMC-RS   | 2.5831 s <sup>(2)</sup> | 3.0818 s <sup>(3)</sup> | 4.6049 s <sup>(3)</sup> |
| L-MAC-DMC-BS   | 2.3717 s <sup>(2)</sup> | 3.0818 s <sup>(3)</sup> | 3.7337 s <sup>(3)</sup> |
| L-MAC-SC-MILP  | 5.0113 s                | 9.4192 s                | 9.8322 s                |
| L-MAC-DMC-MILP | 2.3717 s <sup>(2)</sup> | 3.0818 s <sup>(3)</sup> | 3.4779 s <sup>(3)</sup> |

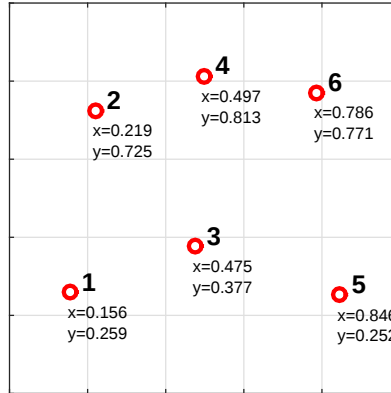


Figure 5.15: Anchors positions normalized to  $5c$ .

## 5.7 Conclusion

We have formulated the problem of scheduling the localization packets of the anchors in single-channel and multi-channel partially-connected underwater sensor networks. We have introduced the concept of dynamic multi-channel packet scheduling. In this approach the network splits the existing channel into several subchannels adaptively in order to reduce the scheduling time. Furthermore, we have proposed two low-complexity algorithms in order to minimize the duration of the localization task. We have shown that the proposed algorithms perform near optimal, and much better than other alternative solutions such as TDMA-based or position-unaware approaches. Furthermore, through comprehensive simulations, it has been revealed that the mean of the localization task duration depends on the number of subchannels, localization packet length, the anchors' maximum transmission range, the number of collision-risk neighbors and their modified average distances. We have found that, multi-channel scheduling approaches perform better than their single-channel ones especially when the ratio of the packet length to the average pair-wise distance is low. Moreover, we observed that a system that adjusts the number of subchannels dynamically has the highest performance among other position-aware algorithms. The proposed scheduling algorithms cannot directly be used for cooperative localizations, unless a localized sensor node participates in the localization process as an anchor before scheduling. In the future, we want to address the problem of localization when most of

the underwater nodes are not under the coverage of the anchors. The optimal scheduling protocol for such networks can be considered as an extension of the work carried out in this paper.

# Chapter 6

## Collision Tolerant and Collision Free Packet Scheduling for Underwater Acoustic Localization

Hamid Ramezani<sup>1</sup>

---

<sup>1</sup>This chapter is a verbatim copy of the journal paper published as: Hamid Ramezani, Fatemeh Fazel, Milica Stojanovic, and Geert Leus. "Collision Tolerant and Collision Free Packet Scheduling for Underwater Acoustic Localization." IEEE Transactions on Wireless Communications, vol. 14, no. 5, pp. 2584-2595, Jan 2015.

### Abstract

This article considers the joint problem of packet scheduling and self-localization in an underwater acoustic sensor network with randomly distributed nodes. In terms of packet scheduling, our goal is to minimize the localization time, and to do so we consider two packet transmission schemes, namely a collision-free scheme (CFS), and a collision-tolerant scheme (CTS). The required localization time is formulated for these schemes, and through analytical results and numerical examples their performances are shown to be dependent on the circumstances. When the packet duration is short (as is the case for a localization packet), the operating area is large (above 3km in at least one dimension), and the average probability of packet-loss is not close to zero, the collision-tolerant scheme is found to require a shorter localization time. At the same time, its implementation complexity is lower than that of the collision-free scheme, because in CTS, the anchors work independently. CTS consumes slightly more energy to make up for packet collisions, but it is shown to provide a better localization accuracy. An iterative Gauss-Newton algorithm is employed by each sensor node for self-localization, and the Cramér Rao lower bound is evaluated as a benchmark.

## 6.1 Introduction

Developments in computer systems and networking have been paving a way towards fully autonomous underwater acoustic sensor networks (UASNs) [54], [36]. Modern underwater networks are expected to handle many tasks automatically. To enable applications such as tsunami monitoring, oil field inspection, bathymetry mapping, or shoreline surveillance, the sensor nodes measure various environmental parameters, encode them into data packets, and exchange the packets with other sensor nodes or send them to a fusion center. In many underwater applications, the sensed data has to be labeled with the time and the location of their origin to provide meaningful information. Therefore, sensor nodes that explore the environment and gather data have to know their position, and this makes localization an important task for the network.

Due to the challenges of underwater acoustic communications such as low data rates and long propagation delays with variable sound speed [10], a variety of localization algorithms have been introduced and analyzed in the literature [56] [49]. In contrast to underwater systems, sensor nodes in terrestrial wireless sensor networks (WSNs) can be equipped with a GPS module to determine location.

GPS signals (radio-frequency signals), however, cannot propagate more than a few meters in (salty) water, and underwater acoustic signals are used instead. In addition, radio signals experience negligible propagation delays as compared to the sound (acoustic) waves.

An underwater sensor node can determine its location by measuring the time of flight (ToF) to several anchors with known positions, and performing multilateration. Other approaches may be employed for self-localization, such as finger-printing [48] or angle of arrival estimation [128]. All these approaches require packet transmission from anchors.

Many factors determine the accuracy of self-localization. Other than noise, the number of anchors, their constellation and relative position of the sensor node [129], propagation losses and fading also affect the localization accuracy. Some of these parameters can be adjusted to improve the localization accuracy, but others cannot.

Although a great deal of research exists on underwater localization algorithms [54], little work has been done to determine how the anchors should transmit their packets to the sensor nodes. In long base-line (LBL) systems where transponders are fixed on the sea floor, an underwater node interrogates the transponders for round-trip delay estimation [130]. In the underwater positioning scheme of [131], a master anchor sends a beacon signal periodically, and other anchors transmit their packets in a given order after the reception of the beacon from the previous anchor. The localization algorithm in [61] addresses the problem of joint node discovery and collaborative localization without the aid of GPS. The algorithm starts with a few anchors as primary seed nodes, and as it progresses, suitable sensor nodes are converted to seed nodes to help in discovering more sensor nodes. The algorithm works by broadcasting command packets which the nodes use for time-of-flight measurements. The authors evaluate the performance of the algorithm in terms of the average network set-up time and coverage. However, physical factors such as packet loss due to fading or shadowing and collisions are not included, and it is not established whether this algorithm is optimal for localization. In reactive localization [132], an underwater node initiates the process by transmitting a “hello” message to the anchors in its vicinity, and those anchors that receive the message transmit their packets. An existing medium access control (MAC) protocol may be used for packet exchanging [133]; however, there is no guarantee that it will perform satisfactorily for the localization task. The performance of localization under different MAC protocols is evaluated in [110], where it is shown that a simple carrier sense multiple access (CSMA) protocol performs better than

the recently introduced underwater MAC protocols such as T-Lohi [104].

In our previous work, we considered optimal collision-free packet scheduling in a UASN for the localization task in single-channel (L-MAC) [134] and multi-channel [122] scenarios (DMC-MAC). In these algorithms, the position information of the anchors is used to minimize the localization time. In spite of the remarkable performance of L-MAC and DMC-MAC over other algorithms (or MAC protocols), they are highly demanding. The main drawback of L-MAC or DMC-MAC is that they require a fusion center which gathers the positions of all the anchors, and decides on the time of packet transmission from each anchor. In addition, these two collision-free algorithms need the anchors to be synchronized and equipped with radio modems in order to exchange information fast.

In this paper, we consider packet scheduling algorithms that do not need a fusion center. Although the synchronization of the anchors which are equipped with GPS is not difficult, the proposed algorithms can work with asynchronous anchors if there is a request from a sensor node.

We assume a single-hop UASN where anchors are equipped with half-duplex acoustic modems, and can broadcast their packets based on two classes of scheduling: a collision-free scheme (CFS), where the transmitted packets never collide with each other at the receiver, and a collision-tolerant scheme (CTS), where the collision probability is controlled by the packet transmission rate in such a way that each sensor node can receive sufficiently many error-free packets for self-localization. Our contributions are listed below.

- Assuming packet loss and collisions, the localization time is formulated for each scheme, and its minimum is obtained analytically for a predetermined probability of successful localization for each sensor node. A shorter localization time allows for a more dynamic network, and leads to a better network efficiency in terms of throughput.
- It is shown how the minimum number of anchors can be determined in order to reach the desired probability of self-localization.
- An iterative Gauss-Newton self-localization algorithm is introduced for a sensor node which experiences packet loss or collision. Furthermore, the way in which this algorithm can be used for each packet scheduling scheme is outlined.
- The Cramér Rao lower bound (CRB) on localization is derived for each scheme. Other than the distance-dependent signal to noise ratio, the effects

of packet loss due to fading or shadowing, collisions, and the probability of successful self-localization are included in this derivation.

The structure of the paper is as follows. Section 6.2 describes the system model, and outlines the self-localization process. The problem of minimizing the localization time in the collision-free and collision-tolerant packet transmission schemes is formulated and analyzed in Section 6.3.1 and Section 6.3.2, respectively. The self-localization algorithm is introduced in Section 6.4. The average energy consumption is analyzed in Section 6.5, and Section 6.6 compares the two classes of localization packet scheduling through several numerical examples. Finally, we conclude the paper in Section 6.7, and outline the topics of future works.

## 6.2 System model

We consider a UASN consisting of  $M$  sensor nodes and  $N$  anchors. The anchor index starts from 1, whereas the sensor node index starts from  $N + 1$ . Each anchor in the network encapsulates its ID, its location, time of packet transmission, and a predetermined training sequence for the time of flight estimation. The so-obtained localization packet is broadcast to the network based on a given protocol, e.g., periodically, or upon the reception of a request from a sensor node. The system structure is specified as follows.

- Anchors and sensor nodes are equipped with half-duplex acoustic modems, i.e., they cannot transmit and receive simultaneously.
- Anchors are placed randomly on the surface, and have the ability to move within the operating area. The anchors are equipped with GPS and can determine their positions which will be broadcast to the sensor nodes. It is assumed that the probability density function (pdf) of the distance between the anchors is known,  $f_D(z)$ . It is further assumed that the sensor nodes are located randomly in an operating area according to some probability density function. The sensor nodes can move in the area, but within the localization process, their position is assumed to be constant. The pdf of the distance between a sensor node and an anchor is  $g_D(z)$ . These pdfs can be estimated from the empirical data gathered during past network operations.
- We consider a single-hop network where all the nodes are within the communication range of each other.

- The received signal strength (which is influenced by path-loss, fading and shadowing) is a function of transmission distance. Consequently, the probability of a packet loss is a function of distance between any pair of nodes in the network.

The considered localization algorithms are assumed to be based on ranging, whereby a sensor node determines its distance to several anchors via ToF or round-trip-time (RTT). Each sensor node can determine its location if it receives at least  $K$  different localization packets from  $K$  different anchors. The value of  $K$  depends on the geometry (2D or 3D), and other factors such as whether depth of the sensor node is available, or whether sound speed estimation is required. The value of  $K$  is usually 3 for a 2D operating environment with known sound speed and 4 for a 3D one. In a situation where the underwater nodes are equipped with pressure sensors, three different successful packets would be enough for a 3D localization algorithm [94].

The localization procedure starts either periodically for a predetermined duration (in a synchronized network), or upon receiving a request from a sensor node (in any kind of network, synchronous or asynchronous) as explained below.

**Periodic localization:** If all the nodes in the network including anchors and sensor nodes are synchronized with each other, a periodic localization approach may be employed. In this approach, after the arrival of a packet from the  $j$ -th anchor, the  $m$ -th sensor node estimates its distance to that anchor as  $\hat{d}_{m,j} = c(\hat{t}_{m,j}^R - t_j^T)$  where  $c$  is the sound speed,  $t_j^T$  is the time at which the anchor transmits its packet, and  $\hat{t}_{m,j}^R$  is the estimated time at which the sensor node receives this packet. The departure time  $t_j^T$  is obtained by decoding the received packet (the anchor inserts this information into the localization packet), and the arrival time  $\hat{t}_{m,j}^R$  can be calculated by correlating the received signal with the known training sequence (or similar procedures). The estimated time of arrival is related to the actual arrival time through  $\hat{t}_{m,j}^R = t_{m,j}^R + n_{m,j}$ , where  $n_{m,j}$  is zero-mean Gaussian noise with power  $\sigma_{m,j}^2$  which varies with distance and can be modeled as [92]

$$\sigma_{m,j}^2 = k_E d_{m,j}^{n_0}, \quad (6.1)$$

with  $d_{m,j}$  the distance between the  $j$ -th anchor and the sensor node,  $n_0$  the path-loss exponent (spreading factor), and  $k_E$  a constant that depends on system parameters (such as signal bandwidth, sampling frequency, channel characteristics, and noise level). In periodic localization, sensor nodes are not required to be synchronized with the anchors. If they are not synchronized, they can calculate

the time-differences of arrival (TDoAs) from the measured ToFs; however, we will not consider this situation in our calculation.

**On-demand localization:** In this procedure (which can be applied to a synchronous or an asynchronous network) a sensor node initiates the localization process. It transmits a high-power frequency tone immediately before the request packet. The tone wakes the anchors up from their idle mode, and puts them into the listening mode. The request packet may also be used for a more accurate estimation of the arrival time. We assume that all the anchors have been correctly notified by this frequency tone. After the anchors have received the wake up tone, they reply with localization packets. The time when the request has been received by an anchor,  $t_{j,m}^R$ , and the time  $t_j^T$  at which a localization packet is transmitted are included in the localization packet. This information will be used by the sensor node to estimate its round-trip-time (which is proportional to twice the distance) to the anchor. The round-trip-time can be modeled as

$$\hat{t}_{m,j}^{\text{RTT}} = (t_{m,j}^R - t_m^T) - (t_{j,m}^R - t_j^T) + n_{j,m} + n_{m,j}, \quad (6.2)$$

where  $t_m^T$  is the transmission time of the request signal from the sensor node. Therefore, the estimated distance to anchor  $j$  is

$$\hat{d}_{m,j} = \frac{1}{2} c \hat{t}_{m,j}^{\text{RTT}}. \quad (6.3)$$

After the sensor node estimates its location, it broadcasts its position to other sensor nodes. This enables the sensor nodes which have overheard the localization process to estimate their positions without initializing another localization task [53].

The time it takes for an underwater node to gather at least  $K$  different packets from  $K$  different anchors is called the localization time. In the next section, we formally define the localization time, and show how it can be minimized for the collision-free and collision-tolerant packet transmission schemes.

## 6.3 Packet scheduling

### 6.3.1 Collision-free packet scheduling

Collision-free localization packet transmission is analyzed in [134], where it is shown that in a fully-connected (single-hop) network, based on a given sequence of the anchors' indices, each anchor has to transmit immediately after receiving the previous anchor's packet. Furthermore, it is shown that there exists an optimal ordering sequence which minimizes the localization time. However, to obtain that

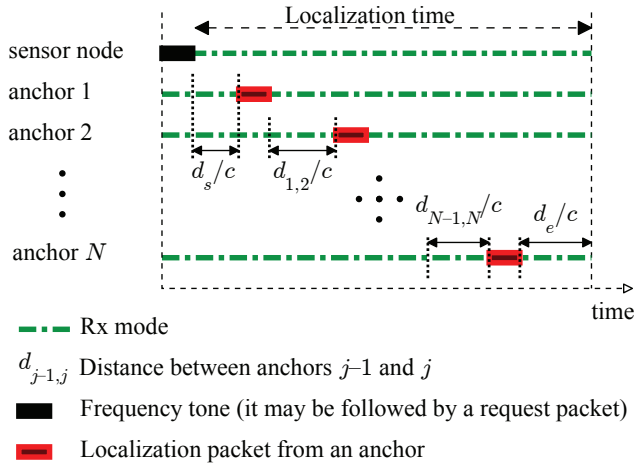


Figure 6.1: Packet transmission from anchors in the collision-free scheme. Here, each anchor transmits its packets according to its index value (ID number). All links between anchors are assumed to function properly in this figure (there are no missing links).

sequence, a fusion center is required to know the positions of all the anchors. In a situation where this information is not available, we may assume that anchors simply transmit in order of their ID numbers as illustrated in Fig. 6.1.

In the event of a packet loss, a subsequent anchor will not know when to transmit. If an anchor does not receive a packet from a previous anchor, it waits for a predefined time (counting from the starting time of the localization process), and then transmits its packet, similarly as introduced in [65]. With a slight modification of the result from [65], the waiting time for the  $j$ -th anchor who has not received a packet from its previous anchor, could be as short as  $t_k + (j - k) (T_p + \frac{D_{aa}}{c})$ , where  $k$  is the index of the anchor whose packet is the last one which has been received by the  $j$ -th anchor,  $t_k$  is the time at which this packet was transmitted from the  $k$ -th anchor (counting from the starting time of the localization process),  $c$  is the sound speed,  $\frac{D_{aa}}{c}$  is the maximum propagation delay between two anchors, and  $T_p$  is the packet length. The packet length is related to the system bandwidth  $B$  (or symbol time  $T_s \approx \frac{1}{B}$ ), number of bits in each symbol  $b_s$ , number of bits in each packet  $b_p$ , and guard time  $T_g$  as formulated in

$$T_p = T_g + \frac{b_p}{b_s} T_s. \quad (6.4)$$

Under this condition, the transmission time of the  $j$ -th anchor  $t_j$  can be selected

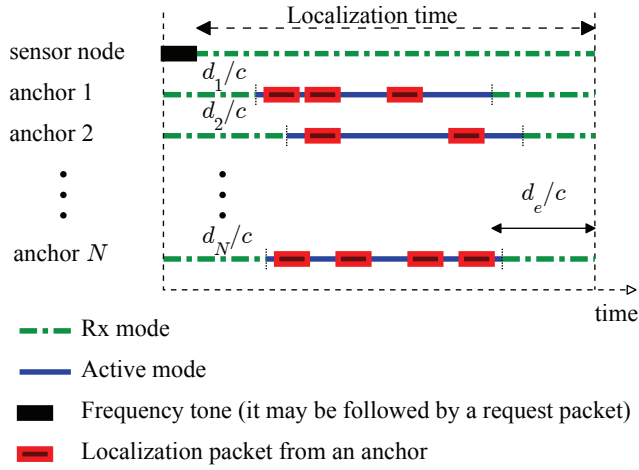


Figure 6.2: Packet transmission from anchors in the collision-tolerant scheme. Here, each anchor transmits its packets at random according to a Poisson distribution.

from one of the values listed in Table 6.1.

Table 6.1: Possible times that anchor  $j$  transmits its packet.

|          | $t_j$                                                           | with probability                                   |
|----------|-----------------------------------------------------------------|----------------------------------------------------|
| 1        | $t_{j-1} + \frac{d_{j,j-1}}{c} + T_p$                           | $1 - p_l(d_{j,j-1})$                               |
| 2        | $t_{j-2} + 2 \left( \frac{D_{aa}}{c} + T_p \right)$             | $[1 - p_l(d_{j,j-2})]p_l(d_{j,j-1})$               |
| $\vdots$ | $\vdots$                                                        | $\vdots$                                           |
| $j - k$  | $t_k + (j - k) \left( \frac{D_{aa}}{c} + T_p \right)$           | $[1 - p_l(d_{j,k})]\Pi_{m=k+1}^{j-1} p_l(d_{j,m})$ |
| $\vdots$ | $\vdots$                                                        | $\vdots$                                           |
| $j - 1$  | $\frac{D_r}{c} + (j - 1) \left( \frac{D_{aa}}{c} + T_p \right)$ | $\Pi_{m=1}^{j-1} p_l(d_{j,m})$                     |

where  $D_r = D_{sa}$  in on-demand localization which is the distance corresponding to the maximally separated sensor-anchor pair, and  $D_r = 0$  in periodic localization,  $t_1 = 0$  for periodic localization, and  $t_1 = \frac{d_s}{c}$  for on-demand localization, with  $d_s$  the distance between the first anchor and the sensor who sent the request packet, and  $p_l(d_{i,j})$  is the probability of packet loss between two anchors located  $d_{i,j}$  meters away from each other. The packet loss can be defined as

$$p_l(d) = 1 - \int_{\gamma_0 N_0 B}^{\infty} f_{X_0|d}(x) dx \quad (6.5)$$

where  $N_0B$  is the noise power,  $\gamma_0$  is the minimum SNR at which a received packet can be detected at the receiver, and given the distance between two nodes,  $d$ ,  $f_{X_0|d}(x)$  is the conditional pdf of the received signal power which will be derived in the next subsection. The first row of Table 6.1 indicates that no packet loss (with probability  $1 - p_l(d_{j,j-1})$ ) occurs between the  $j$ -th and  $(j-1)$ -th anchor, and the  $j$ -th anchor transmits after it receives the packet from the  $(j-1)$ -th anchor. The second row denotes that there is a packet loss between the  $j$ -th and  $(j-1)$ -th anchor (with probability  $p_l(d_{j,j-1})$ ), but there is no packet-loss between the  $j$ -th and  $(j-2)$ -th anchor (with probability  $1 - p_l(d_{j,j-2})$ ). Therefore, according to the protocol, the  $j$ -th anchor waits until  $t_{j-2} + 2\left(\frac{D_{aa}}{c} + T_p\right)$  before it transmits its packet. The last row of Table 6.1 specifies that the  $j$ -th anchor has lost all the packets from all anchors, and as a result transmits at a worst possible time to avoid any collision.

Since  $d_{i,j}$  for  $j = 1, \dots, N-1$ , and  $d_s$  are independent of each other, the average time at which the  $j$ -th anchor transmits its packet can be obtained as

$$\begin{aligned} \bar{t}_j = & (1 - \bar{p}_l) \sum_{k=1}^{j-1} \bar{t}_k \bar{p}_l^{j-k-1} + T_p(1 - \bar{p}_l) + \frac{\bar{d}}{c} - \frac{\bar{d}p_l}{c} + \\ & (1 - \bar{p}_l) \left( \frac{D_{aa}}{c} + T_p \right) \sum_{k=2}^{j-1} k \bar{p}_l^{k-1} + \\ & (j-1) \left( \frac{D_{aa}}{c} + T_p \right) \bar{p}_l^{j-1} + \frac{D_r}{c} \bar{p}_l^{j-1} \end{aligned} \quad (6.6)$$

where  $\bar{p}_l$ ,  $\bar{d}$ , and  $\bar{d}p_l$  are the expected values of  $p_l(d_{i,j})$ ,  $d_{i,j}$ , and  $d_{i,j}p_l(d_{i,j})$ , respectively.

The average localization time of a collision-free scheme can be obtained as

$$T_{CF}^{avg} = \bar{t}_N + T_p + \frac{D_{sa}}{c}, \quad (6.7)$$

where  $\frac{D_{sa}}{c}$  is added to ensure that the last transmitted packet from the  $N$ -th anchor reaches the furthest point in the operating area.

In the best case there is no packet loss between the anchors and the average localization time reaches its minimum value at

$$T_{CF}^{low} = (N-1) \frac{\bar{d}}{c} + \frac{\bar{d}_s}{c} + NT_p + \frac{D_{sa}}{c}, \quad (6.8)$$

where  $\bar{d}_s$  is the average distance between a sensor node and an anchor. In the worst case, all the packets between the anchors are lost, and the requesting sensor node is the farthest from the initiating anchor. This case yields the longest localization time given by

$$T_{CF}^{upp} = NT_p + (N-1) \frac{D_{aa}}{c} + \frac{D_{sa}}{c} + \frac{D_{sa}}{c}, \quad (6.9)$$

which is equivalent to a packet transmission based on time division multiple access (TDMA) with time-slot duration  $T_p + \frac{D}{c}$  (assuming  $D = D_{sa} = D_{aa}$ ).

Another figure of merit is the probability with which a node can localize itself. If this probability is required to be above a design value  $P_{ss}$ , the necessary number of anchors which also minimizes  $T_{CF}^{avg}$  ( $T_{CF}^{avg}$  is an increasing function of  $N$ ) is determined as the smallest  $N$  for which

$$P_{CF}^{loc} = \sum_{k=K}^N \binom{N}{k} p_{CF}^k (1 - p_{CF})^{N-k} \geq P_{ss} \quad (6.10)$$

where  $p_{CF}$  is the probability that a transmitted packet reaches a sensor node correctly, and it can be calculated as

$$p_{CF} = \int_{\gamma_0 N_0 B}^{\infty} f_{X_0}(x) dx, \quad (6.11)$$

where  $f_{X_0}(x)$  is the pdf of the received signal power.

### 6.3.2 Collision-tolerant packet scheduling

To avoid the need for coordination among anchor nodes, in a collision-tolerant packet scheduling, anchors work independently of each other. During a localization period or upon receiving a request from a sensor node, they transmit randomly, e.g. according to a Poisson distribution with an average transmission rate of  $\lambda$  packets per second as shown in Fig. 6.2. Packets transmitted from different anchors may now collide at a sensor node, and the question arises as to what is the probability of successful reception. This problem is a mirror image of the one investigated in [135] where sensor nodes transmit their packets to a common fusion center. Unlike [135] however, where the sensors know their location, and power control fully compensates for the known path-loss, path-loss is not known in the present scenario, and there is no power control. The average received signal strength is thus different for different links (this signal strength, along with a given fading model, determines the probability of packet loss). In this regard, the signal received at the  $m$ -th sensor node from the  $j$ -th anchor is

$$v_{m,j}(t) = c_{m,j} v_j(t) + i_m(t) + w_m(t), \quad (6.12)$$

where  $v_j(t)$  is the signal transmitted from the  $j$ -th anchor,  $c_{m,j}$  is the channel gain,  $w_m(t)$  is the additive white Gaussian noise with power  $N_0 B$ , and  $i_m(t)$  is the interference caused by other anchors whose packets overlap with the desired packet,

$$i_m(t) = \sum_{k \neq j} c_{m,k} v_k(t - \tau_k), \quad (6.13)$$

with  $\tau_k$  being the difference in the arrival times of the interfering signals w.r.t. the desired signal which is modeled as an exponentially distributed random variable. The signal-to interference-plus-noise-ratio (SINR) at the receiver depends on the interference level, and is given by

$$\gamma = \frac{X_0}{I_0 + N_0 B}, \quad (6.14)$$

where  $X_0 = |c_{m,j}|^2 P_0$  is the power of the signal of interest with  $P_0$  the anchor's transmit power, and where  $I_0$  is the total interference power which can be expressed as

$$I_0 = \sum_{i=1}^q |c_{m,k_i}|^2 P_0 \quad (6.15)$$

with  $q$  the number of interferers, and  $k_i$  the index of the  $i$ -th interferer. We can express the signal power as

$$|c_{m,j}|^2 = a_{\text{PL}}^{-1} (d_{m,j}) e^{g_{m,j}} |h_{m,j}|^2, \quad (6.16)$$

where  $g_{m,j} \sim \mathcal{N}(0, \sigma_g^2)$  models the large scale log-normal shadowing,  $h_{m,j} \sim \mathcal{CN}(\bar{h}, \sigma_h^2)$  models the small scale fading, and  $a_{\text{PL}}$  models the path-loss attenuation which can be formulated as [136]

$$a_{\text{PL}}(d_{i,j}) = \alpha_0 \left( \frac{d_{i,j}}{d_0} \right)^{n_0} a(f)^{d_{i,j}} \quad (6.17)$$

where  $\alpha_0$  is a constant,  $d_0$  is the reference distance,  $n_0$  is the path-loss exponent, and  $a(f)$  is the frequency-dependent absorption coefficient. For localization, where the bandwidth is not large,  $\alpha(f)$  can be approximated by a constant.

The pdf of the received signal power,  $f_{X_0}(x)$  can be obtained numerically. Since  $a_{\text{PL}}$ ,  $g_{m,j}$  and  $h_{m,j}$  are independent random variables, we calculate the pdfs of  $10 \log |h_{m,j}|^2$ ,  $10 \log e^{g_{m,j}}$ , and  $-10 \log a_{\text{PL}}$  separately. Then we convolve them which results in  $f_{X_0, \text{dB}}(x_{\text{dB}})$ . With a simple change of variable  $x = 10^{0.1 x_{\text{dB}}}$  we can find  $f_{X_0}(x)$ , and the pdf of the interference can be obtained as

$$f_{I_0}(x) = \underbrace{f_{X_0}(x) * f_{X_0}(x) * \dots * f_{X_0}(x)}_{q \text{ times}}. \quad (6.18)$$

The probability that a packet is received correctly by a sensor node is then [135]

$$p_s = \sum_{q=0}^{N-1} P(q) p_{s|q}, \quad (6.19)$$

where  $P(q) = \frac{(2N\lambda T_p)^q}{q!} e^{-2N\lambda T_p}$  is the probability that  $q$  packets interfere with the desired packet, and  $p_{s|q}$  is the probability that the desired packet “survives” under

this condition,

$$p_{s|q} = \begin{cases} \int_{\gamma_0 N_0 B}^{\infty} f_{X_0}(x) dx & q = 0 \\ \int_{\gamma_0}^{\infty} \int_{N_0 B}^{\infty} f_{X_0}(\gamma w) f_{I_0}(w - N_0 B) w dw d\gamma & q \geq 1 \end{cases} \quad (6.20)$$

where  $w = I_0 + N_0 B$ .

In addition, it should be noted that multiple receptions of a packet from an anchor does not affect the probability of self-localization (localization coverage), but in case where a sensor node is able to localize itself, multiple receptions of a packet from an anchor affects the accuracy of the localization (see Section 6.4).

If we assume that the packets transmitted from the same anchor fade independently, the probability of receiving a useful packet from an anchor during the transmission time  $T_T$  can now be approximated by [135]

$$p_{CT} = 1 - e^{-p_s \lambda T_T}, \quad (6.21)$$

and the probability that a sensor node accomplishes self-localization using  $N$  anchors can be obtained as

$$P_{CT}^{\text{loc}} = \sum_{k=K}^N \binom{N}{k} p_{CT}^k (1 - p_{CT})^{N-k}, \quad (6.22)$$

which is equivalent to the probability that a node receives at least  $K$  different localization packets.

It can be shown that  $P_{CT}^{\text{loc}}$  is an increasing function of  $T_T$  (see Appendix A1), and as a result for any value of  $p_s \lambda \neq 0$ , there is a  $T_T$  that leads to a probability of self-localization equal to or greater than  $P_{ss}$ . The minimum value for the required  $T_T$  can be obtained at a point where  $p_s \lambda$  is maximum ( $\lambda_{\text{opt}}$ ). It can be proven that the lower bound of  $\lambda_{\text{opt}}$  is  $\lambda_{\text{opt}}^{\text{low}} = \frac{1}{2NT_p}$ , and its upper bound is  $\frac{N+1}{2NT_p}$  (see Appendix A2). These points will be illustrated via numerical examples in Section 6.6 (cf. Fig. 6.3).

Given the number of anchors  $N$ , and a desired probability of successful self-localization  $P_{ss}$ , one can determine  $p_{CT}$  from (6.22), while  $\lambda$  and the minimum localization time can be determined jointly from (6.19) and (6.21). Similarly as in the collision-free scheme, we then add the time of request  $\frac{d_s}{c}$ , and the maximum propagation delay between a sensor-anchor pair  $\frac{D_{sa}}{c}$  to the (minimum)  $T_T$  that is obtained from (6.19) and (6.21). The so-obtained value represents the (minimum) localization time ( $T_{CT}^{\text{min}}$ )  $T_{CT}$ , for the collision-tolerant scheme.

## 6.4 Self-localization process

We have seen that a sensor node requires at least  $K$  distinct packets (or time-of-flight measurements) in order to determine its location. However, it may receive more than  $K$  different packets, as well as some replicas, i.e,  $q_j$  packets from anchor  $j$ , where  $j = 1, \dots, N$ . In this case, a sensor uses all of this information for self-localization. Note that in the collision-free scheme,  $q_j$  is either zero or one; however, in the collision-tolerant scheme  $q_j$  can be more than 1. Packets received from the  $j$ -th anchor can be used to estimate the sensor node's distance to that anchor, and the redundant packets add diversity (or reduce measurement noise) for this estimate. In the next two subsections, we show how all of the correctly received packets can be used in a localization algorithm, and how the CRB of the location estimate can be obtained for the proposed scheduling schemes.

### 6.4.1 Localization algorithm

After the anchors transmit their localization packets, each sensor node has  $Q$  measurements. Each measurement is contaminated by noise whose power is related to the distance between the sensor and the anchor from which the measurement has been obtained. The  $l$ -th measurement obtained from the  $j$ -th anchor is related to the sensor's position  $\mathbf{x}$  (sensor index is omitted for simplicity) as

$$\hat{t}_l = f(\mathbf{x}) + n_l, \quad (6.23)$$

where  $n_l$  is the measurement noise (see (6.1)) and  $f(\mathbf{x})$  is

$$f(\mathbf{x}) = \frac{1}{c} \|\mathbf{x} - \mathbf{x}_j\|_2 \quad (6.24)$$

where  $\mathbf{x}_j$  is the  $j$ -th anchor's position. Stacking all the measurements gives us a  $Q \times 1$  vector  $\hat{\mathbf{t}}$ . The number of measurements is given by

$$Q = \sum_{j=1}^N q_j, \quad (6.25)$$

where  $q_j$  is the number of measurements which are obtained correctly from the  $j$ -th anchor. In CFS,  $q_j$  is a Bernoulli random variable with success probability  $P_j^1 = P(q_j = 1) = 1 - p_l(d_j)$  where  $d_j$  is the distance between the sensor node and the  $j$ -th anchor. In CTS  $q_j$  is a Poisson random variable with distribution

$$P_j^n = P(q_j = n) = \frac{(p_s \lambda T_T)^n}{n!} e^{-\lambda T_T p_{s|\mathbf{d}}^j}, \quad (6.26)$$

where  $p_{s|\mathbf{d}}^j$  is the conditional probability that a sensor node correctly receives a packet from the  $j$ -th anchor, knowing its distance from all anchors (elements of

d). This pdf can be found from the conditional pdf of the received signal and the interference power (see (6.19) and (6.20)).

Since the measurement errors are independent of each other, the maximum likelihood solution for  $\mathbf{x}$  is given by

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \|\hat{\mathbf{t}} - \mathbf{f}(\mathbf{x})\|_2, \quad (6.27)$$

which can be calculated using a method such as the Gauss-Newton algorithm specified in Algorithm 5. In this algorithm,  $\eta$  controls the convergence speed,

---

**Algorithm 5** Gauss-Newton Algorithm
 

---

Start with an initial location guess.

Set  $i = 1$  and  $E = \infty$ .

**while**  $i \leq I$  and  $E \geq \epsilon$  **do**

  Next state:

$$\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} -$$

$$\eta \left( \nabla \mathbf{f}(\mathbf{x}^{(i)})^T \nabla \mathbf{f}(\mathbf{x}^{(i)}) \right)^{-1} \nabla \mathbf{f}(\mathbf{x}^{(i)})^T (\mathbf{f}(\mathbf{x}^{(i)}) - \hat{\mathbf{t}})$$

$$E = \|\mathbf{x}^{(i+1)} - \mathbf{x}^{(i)}\|$$

$$i = i + 1$$

**end while**

$$\hat{\mathbf{x}} = \mathbf{x}^{(i)}$$


---

$\nabla \mathbf{f}(\mathbf{x}^{(i)}) = \left[ \frac{\partial f_1}{\partial \mathbf{x}}, \frac{\partial f_2}{\partial \mathbf{x}}, \dots, \frac{\partial f_Q}{\partial \mathbf{x}} \right]_{\mathbf{x}=\mathbf{x}^{(i)}}^T$  represents the gradient of the vector  $\mathbf{f}$  w.r.t. the variable  $\mathbf{x}$  at  $\mathbf{x}^{(i)}$ ,  $\mathbf{x}^{(i)}$  is the estimate in the  $i$ -th iteration, and  $\frac{\partial f_l}{\partial \mathbf{x}} = \left[ \frac{\partial f_l}{\partial x}, \frac{\partial f_l}{\partial y}, \frac{\partial f_l}{\partial z} \right]^T$  where  $l = 1, \dots, Q$ . Here,  $I$  and  $\epsilon$  are the user-defined limits on the stopping criterion. The initial guess is also an important factor and can be determined through triangulation, similarly as explained in [137].

### 6.4.2 Cramér-Rao bound

The Cramér-Rao bound is a lower bound on the variance of any unbiased estimator of a deterministic parameter. In this subsection, we derive the CRB for the location estimate of a sensor node.

In order to find the CRB, the Fisher information matrix (FIM) has to be calculated. The Fisher information is a measure of information that an observable random variable  $\hat{\mathbf{t}}$  carries about an unknown parameter  $\mathbf{x}$  upon which the pdf of  $\hat{\mathbf{t}}$  depends. The elements of the FIM are defined as

$$\mathbf{I}(\mathbf{x})_{i,j} = -\mathbb{E} \left[ \frac{\partial^2 \log h(\hat{\mathbf{t}}; \mathbf{x})}{\partial x_i \partial x_j} \right] \quad (6.28)$$

where  $\mathbf{x}$  is the location of the sensor node,  $h(\hat{\mathbf{t}}; \mathbf{x})$  is the pdf of the measurements

parametrized by the value of  $\mathbf{x}$ , and the expected value is over the cases where the sensor is localizable.

In a situation where the measurements (ToF or RTTs between a sensor node and the anchors) are contaminated with Gaussian noise (whose power is related to the mutual distance between a sensor-anchor pair), the elements of the FIM can be formulated as

$$\mathbf{I}(\mathbf{x})_{i,j} = \frac{1}{P^{\text{loc}}} \sum_{q_N=0}^{Q_N} \cdots \sum_{q_2=0}^{Q_2} \sum_{q_1=0}^{Q_1} \text{s.t. } \{q_1, \dots, q_N\} \text{ enable self-localization} \quad (6.29)$$

$$\left\{ \frac{\partial \mathbf{f}}{\partial x_i}^T \mathbf{R}_w^{-1} \frac{\partial \mathbf{f}}{\partial x_j} + \frac{1}{2} \text{tr} \left[ \mathbf{R}_w^{-1} \frac{\partial \mathbf{R}_w}{\partial x_i} \mathbf{R}_w^{-1} \frac{\partial \mathbf{R}_w}{\partial x_j} \right] \right\} \prod_{j=1}^N P_j^{q_j}$$

where  $P^{\text{loc}}$  is the localization probability (see (6.10) and (6.22)),  $Q_i = 1$  for CFS, and  $\infty$  for CTS,  $\mathbf{R}_w$  is the  $\mathcal{Q} \times \mathcal{Q}$  noise covariance matrix

$$\frac{\partial \mathbf{R}_w}{\partial x_i} = \text{diag} \left( \frac{\partial [\mathbf{R}_w]_{11}}{\partial x_i}, \frac{\partial [\mathbf{R}_w]_{22}}{\partial x_i}, \dots, \frac{\partial [\mathbf{R}_w]_{\mathcal{Q}\mathcal{Q}}}{\partial x_i} \right), \quad (6.30)$$

and

$$\frac{\partial \mathbf{f}}{\partial x_i} = \left[ \frac{\partial f_1}{\partial x_i}, \frac{\partial f_2}{\partial x_i}, \dots, \frac{\partial f_{\mathcal{Q}}}{\partial x_i} \right]^T, \quad (6.31)$$

with  $f_i$  a ToF or RTT measurement.

Once the FIM has been computed, the lower bound on the variance of the estimation error can be expressed as  $\text{CRB} = \sum_{i=1}^3 \text{CRB}_{x_i}$  where  $\text{CRB}_{x_i}$  is the variance of the estimation error in the  $i$ -th variable, defined as

$$\text{CRB}_{x_i} = [\mathbf{I}^{-1}(\mathbf{x})]_{ii}. \quad (6.32)$$

Note that the CRB is meaningful if a node is localizable ( $\frac{1}{P^{\text{loc}}}$  in (6.29)), meaning that a sensor node has at least  $K$  different measurements. Hence, only  $\sum_{k=K}^N \binom{N}{k}$  possible states have to be considered in order to calculate (6.29) for collision-free scheduling, while the number of states is countless for collision-tolerant scheduling. Nonetheless, it can be shown that the number of possible states in CTS can be dropped to that of CFS (see Appendix A3).

## 6.5 Energy consumption

In this section, we investigate the average energy consumed by all the anchors during the localization. In CFS, the receiver of anchor  $j$  is on for  $t_j$  seconds, and

its transmitter is on only for  $T_p$  seconds. With power consumption  $P_L$  in listening mode and  $P_T$  in transmitting mode, the average energy consumption in CFS is

$$E_{CF}^{avg} = NT_p P_T + \sum_{j=1}^N \bar{t}_j P_L, \quad (6.33)$$

where the energy consumed for processing is ignored. As is clear from (6.6), an anchor with a higher index value has to listen to the channel longer, and consequently consumes more energy in comparison with the one that has a lower index. To overcome this problem, anchors can swap indices between localization procedures.

In CTS, the anchors do not need to listen to the channel and they only transmit at an average rate of  $\lambda$  packets per second. The average energy consumption is thus

$$E_{CT}^{avg} = \lambda T_T NT_p P_T. \quad (6.34)$$

For  $(\frac{P_L}{P_T} < \frac{NT_p(\lambda T_T - 1)}{\sum_{j=1}^N \bar{t}_j})$ , the average energy consumption of CTS is always greater than that of CFS. However, as  $\lambda$  gets smaller (or equivalently  $T_{CT}$  gets larger), the energy consumption of CTS reduces.

## 6.6 Numerical results

To illustrate the results, a two-dimensional rectangular-shape operating area with length  $D_x$  and width  $D_y$  is considered with uniformly distributed anchors and sensors. There is no difference in how the anchors and sensor nodes are distributed, and therefore we have  $f_D(d) = g_D(d)$  which can be obtained as [138]

$$f_D(d) = \frac{2d}{D_x^2 D_y^2} [d^2 (\sin^2 \theta_e - \sin^2 \theta_s) + 2D_x D_y (\theta_e - \theta_s) + 2D_x d (\cos \theta_e - \cos \theta_s) - 2D_y d (\sin \theta_e - \sin \theta_s)] \quad (6.35)$$

where  $\theta_s$  and  $\theta_e$  are related to  $d$  as given in Table 6.2.

Table 6.2: Values of  $\theta_s$  and  $\theta_e$  based on distance  $d$ .

| distance                               | $\theta_s$                | $\theta_e$                |
|----------------------------------------|---------------------------|---------------------------|
| $0 \leq d \leq D_y$                    | 0                         | $\frac{\pi}{2}$           |
| $D_y \leq d \leq D_x$                  | 0                         | $\sin^{-1} \frac{D_y}{d}$ |
| $D_y \leq d \leq \sqrt{D_x^2 + D_y^2}$ | $\cos^{-1} \frac{D_x}{d}$ | $\sin^{-1} \frac{D_y}{d}$ |

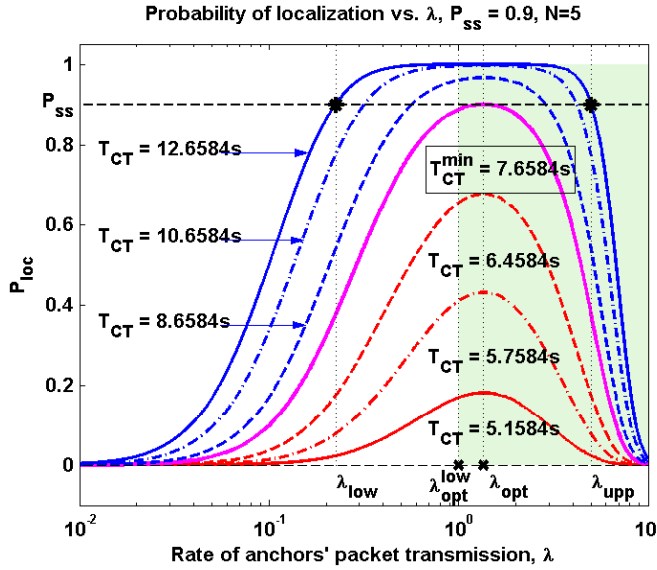


Figure 6.3: Probability of successful localization for different values of  $\lambda$  and  $T_{CT}$ .

The parameter values for the numerical results are listed in Table 6.3, and used for all examples.

The number of bits in each packet is set to  $b_p = 200$  which is sufficient for the position information of each anchor, time of transmission, (arrival time of the request packet), and the training sequence. Assuming QPSK modulation ( $b_s = 2$ ), guard time  $T_g = 50$  ms, and a bandwidth of  $B = 2$  kHz the localization packet length is  $T_p = 100$  ms (see (6.4)). In addition,  $k_E$  is set to  $10^{-10}$  which is approximately equivalent to 1.9 m range accuracy at 1 km away from an anchor. Moreover, to keep the transmitted packets from an anchor in CTS independent of each other, we set  $\sigma_g = 0$  (no shadowing effect) for the simulations. Fig. 6.3 shows the probability of successful self-localization in the collision-tolerant scheme as a function of  $\lambda$  and the indicated value for  $T_{CT}$ . It can be observed that there is an optimal value of  $\lambda$  (denoted by  $\lambda_{opt}$ ) which corresponds to the minimal value of  $T_{CT}$  ( $T_{CT}^{min}$ ) which satisfies  $P_{CT}^{loc} \geq P_{ss}$ . The highlighted area in Fig. 6.3 shows the predicted region of  $\lambda_{opt}$  (obtained in Appendix A2). As it can be seen,  $\lambda_{opt}$  is close to  $\lambda_{opt}^{low}$ , and it gets closer to this value as  $P_{s|q>0}$  gets smaller. In addition, for the values of  $T_{CT}$  greater than  $T_{CT}^{min}$ , a range of values for  $\lambda \in [\lambda_{low}, \lambda_{upp}]$  can attain the desired probability of self-localization. In this case, the lowest value for  $\lambda$  should be selected to minimize the energy consumption.

Table 6.3: Simulation parameters. Note that, in this table some parameters such as  $N$ ,  $D_{aa}$ ,  $T_g$ , etc. are related to other parameters, e.g.,  $N$  depends on the values of the  $\bar{p}_l$ , and  $P_{ss}$ .

| Description                                     | Para.      | Value        | Unit       |
|-------------------------------------------------|------------|--------------|------------|
| Number of anchor nodes                          | $N$        | 5            | -          |
| Number of sensor nodes                          | $M$        | 100          | -          |
| Sound speed                                     | $c$        | 1500         | m/s        |
| Number of required different packets            | $K$        | 3            | -          |
| Area size in $x$ -axis                          | $D_x$      | $3c$         | m          |
| Area size in $y$ -axis                          | $D_y$      | $3c$         | m          |
| Maximum anchor-anchor distance                  | $D_{aa}$   | $3c\sqrt{2}$ | m          |
| Maximum anchor-sensor distance                  | $D_{sa}$   | $3c\sqrt{2}$ | m          |
| Guard time for localization packet              | $T_g$      | 50           | ms         |
| Number of bits per sample                       | $b_s$      | 2            | -          |
| Number of bits per packet                       | $b_p$      | 200          | -          |
| System bandwidth                                | $B$        | 2            | kHz        |
| Localization packet length                      | $T_p$      | 100          | ms         |
| Noise power                                     | $N_0B$     | 100          | dB re 1uPa |
| ToF noise power coefficient                     | $k_E$      | $10^{-10}$   |            |
| Transmit power                                  | $P_0$      | 166          | dB re 1uPa |
| Reference distance                              | $d_0$      | 1            | m          |
| Power coefficient                               | $\alpha_0$ | 1            | m          |
| Path-loss exponent                              | $n_0$      | 1.4          | -          |
| Fading variance                                 | $\bar{h}$  | $\sqrt{1/4}$ | -          |
| Shadowing variance                              | $\sigma_h$ | $\sqrt{3/4}$ | -          |
| Fading mean                                     | $\sigma_g$ | 0            | dB         |
| Absorption coefficient                          | $a(f)$     | 1            | dB/km      |
| Required SNR for packet detection               | $\gamma_0$ | 6            | dB         |
| Request packet arrival delay                    | $d_s/c$    | 0            | s          |
| Required probability of successful localization | $P_{ss}$   | 0.99         | -          |

Fig. 6.4 shows the probability of correct packet reception versus the number of interferers (the desired  $P_{ss}$  is set to 0.90 in this example) for different values of the path-loss exponent  $n_0$ . When there is no interference, the probability of packet reception is high. Yet, when there is an interferer, the chance of correct reception of a packet becomes small (0.126 for  $n_0 = 1.4$ ), and as the number of interferers grows, it gets smaller.

The probability that two or more packets overlap is also depicted in part (b) of this figure for the three values of  $\lambda$  shown in Fig. 6.3. It can be seen that as the value of  $\lambda$  is reduced from  $\lambda_{\text{opt}}$  (which is equivalent to a larger  $T_{\text{CT}}$ ), the probability of collision becomes smaller. The chance of correct packet reception thus increases, and the energy consumption reduces as explained in Section 6.5. In addition, it can be observed that although using  $\lambda_{\text{upp}}$  results in the same performance as  $\lambda_{\text{low}}$ , it relies on the packets that have survived collisions, which is not energy-efficient in practical situations neither for anchors (required energy for multiple packet transmissions) nor for sensor nodes (processing energy needed for packet detection).

Part (a) of Fig. 6.5 shows the time required for localization versus the transmit power. As  $P_0$  increases,  $\bar{p}_l$  gets smaller, and consequently fewer anchors are required for collision-free localization. In Fig. 6.5, for a given  $P_0$ , the number of anchors  $N$  is calculated using (6.10), which is then used to calculate the minimum required time for the collision-free and collision-tolerant localization. Each fall in  $T_{\text{CF}}^{\text{upp}}$  in CFS indicates that the number of anchors has been decreased by one. We also note that for a given number of anchors, the upper and lower bounds of  $T_{\text{CF}}$  are constant over a range of  $P_0$  values; however, the actual performance of both schemes becomes better as  $P_0$  grows. The collision-tolerant approach performs better for a wide range of  $P_0$  values, and as the number of anchors decreases, its performance slightly degrades. In part (b) of Fig. 6.5, we calculate the ratio  $\frac{P_L}{P_T}$  below which the average energy of CTS is greater than that of CFS. The ratio of  $E_{\text{CF}}^{\text{avg}}/E_{\text{CT}}^{\text{avg}}$  is a linear function of  $\frac{P_L}{P_T}$ , and as  $P_0$  increases for larger values of  $\frac{P_L}{P_T}$ , the average energy consumption of CTS becomes greater than that of CFS. In practice, for a range of 6km the  $\frac{P_L}{P_T}$  is less than  $\frac{1}{100}$  [139], and this means that CTS consumes more energy.

Many factors such as noise power or packet length are directly dependent on the operating frequency and the system bandwidth. Assuming single-hop communication among the sensor nodes, an optimum frequency band exists for a given operating area. As the size of the operating area increases, a lower operating frequency (with less bandwidth) is used to compensate for the increased

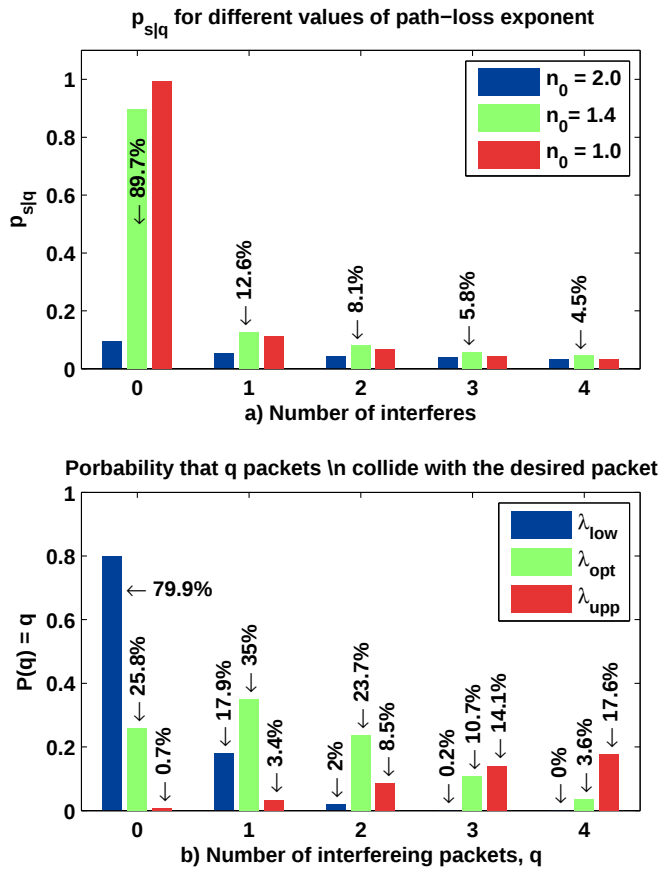


Figure 6.4: a) Probability of successful packet reception versus different number of interferers. b) Probability that  $q$  interferers collide with the desired packet. For this figure,  $\lambda_{low}$ ,  $\lambda_{opt}$  and  $\lambda_{upp}$  are chosen from Fig. 6.3.

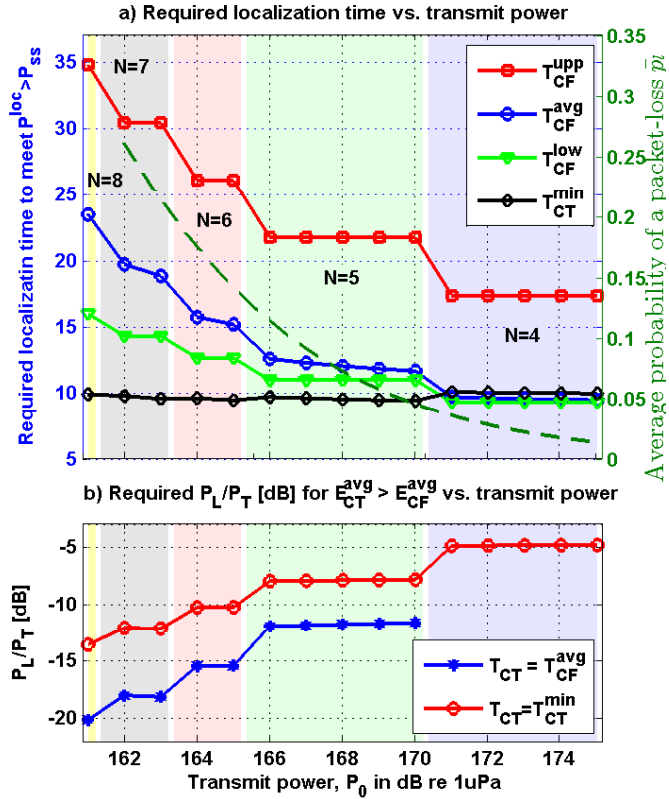


Figure 6.5: a) Effect of transmit power on the minimum time required for localization, and the average probability of a packet-loss  $\bar{p}_l$  (dashed-line); b) The minimum value of  $\frac{P_L}{P_T}$  in dB below which the average energy consumption of CTS is greater than that of CFS.

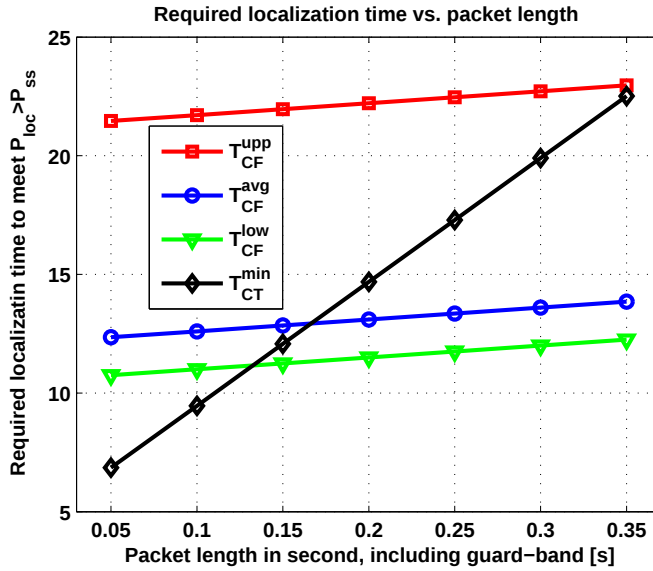


Figure 6.6: Effect of packet length on the minimum time required for localization.

attenuation. Furthermore, as the distance increases, the amount of available bandwidth for the optimum operating frequency also gets smaller [136]. As it was mentioned before, the localization packet is usually short in terms of the number of bits, but its duration (in seconds) still depends on the system bandwidth. Below, we investigate the effect of packet length (or equivalently system bandwidth) on the localization time.

As it is shown in Fig. 6.6, the length of the localization packet plays a significant role in the collision-tolerant algorithm. The minimum localization time grows almost linearly with  $T_p$  in all cases; however, the rate of growth is much higher for the collision-tolerant system than for the collision-free one. At the same time, as shown in Fig. 6.7, the size of the operating area has a major influence on the performance of the CFS, while that of the CTS does not change very much. It can be deduced that in a network where the ratio of packet length to the maximum propagation delay is low, collision-tolerant algorithm outperforms the collision-free one in terms of localization time.

The localization accuracy is related to the noise level at which a ToF measurement is taken, and to the anchors' constellation. If a sensor node in a 2D operating system receives packets from the anchors which are (approximately) located on a line, the sensor node is unable to localize itself (or it experiences a

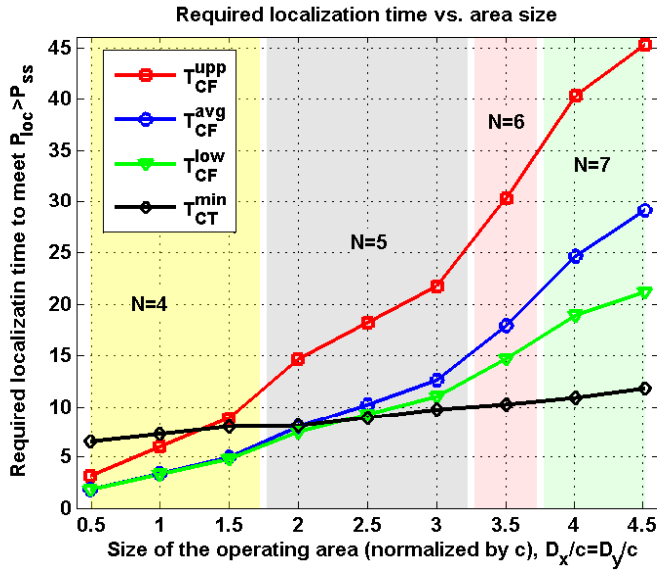


Figure 6.7: Effect of the operating area size on the time required localization.

large error). To evaluate the localization accuracy of each algorithm, we considered  $M = 100$  sensor nodes, and run a Monte Carlo simulation ( $10^3$  runs) to extract the results. The number of iterations in Algorithm 1 is set to  $I = 50$ , and the convergence rate is  $\eta = \frac{1}{5}$ . The  $T_{CF}$  was set equal to the average localization time of CFS. In this special case where  $T_{CT}^{min}$  is lower than  $T_{CF}^{avg}$ , the successful localization probability ( $P^{loc}$ ) of CTS is greater than that of CFS. The probability distribution of the localization error  $\|\hat{\mathbf{x}} - \mathbf{x}\|$  is illustrated in Fig. 6.8 for both schemes. In this figure, the root mean square error (RMSE), and root CRB (R-CRB) are also shown with the dashed and dash-dotted lines, respectively. It can be observed that in CTS the pdf is concentrated at lower values of the localization error compared to CFS, because each sensor in CTS has a chance of receiving multiple copies of the same packet, and thus reducing the range estimation error.

## 6.7 Conclusion

We have considered two classes of packet scheduling for self-localization in an underwater acoustic sensor network, one based on a collision-free design and another based on a collision-tolerant design. In collision-free packet scheduling, the time of the packet transmission from each anchor is set in such a way that

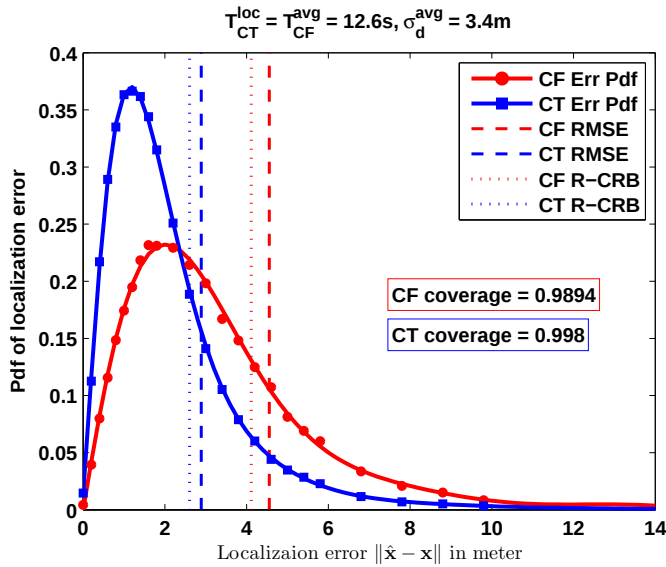


Figure 6.8: Probability distribution of the localization error, and its corresponding CRB for CTS and CFS.

none of the sensor nodes experiences a collision. In contrast, collision-tolerant algorithms are designed so as to control the probability of collision to ensure successful localization with a pre-specified reliability. We have also proposed a simple Gauss-Newton based localization algorithm for these schemes, and derived their Cramér-Rao lower bounds. The performance of the two classes of algorithms in terms of the time required for localization was shown to be dependent on the circumstances. When the ratio of the packet length to the maximum propagation delay is low, as it is the case with localization, and the average probability of packet-loss is not close to zero, the collision-tolerant protocol requires less time for localization in comparison with the collision-free one for the same probability of successful localization. Except for the average energy consumed by the anchors, the collision-tolerant scheme has multiple advantages. The major one is its simplicity of implementation due to the fact that anchors work independently of each other, and as a result the scheme is spatially scalable, with no need for a fusion center. Furthermore, its localization accuracy is always better than that of the collision-free scheme due to multiple receptions of desired packets from anchors. These features make the collision-tolerant localization scheme appealing from a practical implementation view point. In the future, we will extend our work to a multi-hop network where the communication range of the acoustic modems is

much shorter than the size of the operating area.

## Appendices

### A1. $P_{\text{CT}}^{\text{loc}}$ is an increasing function of $T_{\text{CT}}$

In this appendix, we show that the probability of successful localization is an increasing function of the localization time. According to (6.21), and the fact that  $p_s \lambda$  is independent of  $T_{\text{T}}$ , it is clear that  $p_{\text{CT}}$  is an increasing function of  $T_{\text{T}}$ . Therefore,  $P_{\text{CT}}^{\text{loc}}$  is an increasing function of  $T_{\text{T}}$  if  $P_{\text{CT}}^{\text{loc}}$  is an increasing function of  $p_{\text{CT}}$ . The derivative of  $P_{\text{CT}}^{\text{loc}}$  w.r.t.  $p_{\text{CT}}$  is

$$\frac{\partial P_{\text{CT}}^{\text{loc}}}{\partial p_{\text{CT}}} = \sum_{k=K}^N \binom{N}{k} (k - Np_{\text{CT}}) p_{\text{CT}}^{k-1} (1 - p_{\text{CT}})^{N-k-1}. \quad (6.36)$$

With a simple modification we have

$$\begin{aligned} \frac{\partial P_{\text{CT}}^{\text{loc}}}{\partial p_{\text{CT}}} &= \frac{1}{p_{\text{CT}}(1 - p_{\text{CT}})} \left\{ \right. \\ &\quad \left[ \sum_{k=0}^N \binom{N}{k} k p_{\text{CT}}^k (1 - p_{\text{CT}})^{N-k} - \sum_{k=0}^{K-1} \binom{N}{k} k p_{\text{CT}}^k (1 - p_{\text{CT}})^{N-k} \right] - \\ &\quad \left. Np_{\text{CT}} \left[ \sum_{k=0}^N \binom{N}{k} p_{\text{CT}}^k (1 - p_{\text{CT}})^{N-k} - \sum_{k=0}^{K-1} \binom{N}{k} p_{\text{CT}}^k (1 - p_{\text{CT}})^{N-k} \right] \right\}. \end{aligned} \quad (6.37)$$

Using the properties of binomial random variables we have that

$$\sum_{k=0}^N \binom{N}{k} k p_{\text{CT}}^k (1 - p_{\text{CT}})^{N-k} = Np_{\text{CT}}, \quad (6.38)$$

and

$$\sum_{k=0}^N \binom{N}{k} p_{\text{CT}}^k (1 - p_{\text{CT}})^{N-k} = 1. \quad (6.39)$$

Now, equation (6.37) (or equivalently (6.36)) is equal to

$$\frac{\partial P_{\text{CT}}^{\text{loc}}}{\partial p_{\text{CT}}} = \sum_{k=0}^{K-1} \binom{N}{k} (Np_{\text{CT}} - k) p_{\text{CT}}^{k-1} (1 - p_{\text{CT}})^{N-k-1}. \quad (6.40)$$

It can be observed that (6.36) is always positive for  $p_{\text{CT}} < \frac{K}{N}$ , and (6.40) is always positive for  $p_{\text{CT}} > \frac{K}{N}$ . As a result  $\frac{\partial P_{\text{CT}}^{\text{loc}}}{\partial p_{\text{CT}}}$  is positive for any value of  $p_{\text{CT}}$ ; therefore,  $P_{\text{CT}}^{\text{loc}}$  is an increasing function of  $p_{\text{CT}}$ , and consequently of  $T_{\text{T}}$ .

## A2. Maximum value of $p_s \lambda$

The first and second derivatives of  $p_s \lambda$  w.r.t.  $\lambda$  can be obtained as

$$\frac{\partial p_s \lambda}{\partial \lambda} = \sum_{q=0}^N p_{s|q} \frac{x^q e^{-x}}{q!} (q - x + 1), \quad (6.41)$$

$$\frac{(\partial p_s \lambda)^2}{\partial^2 \lambda} = \sum_{q=0}^N p_{s|q} \frac{x^{q-1} e^{-x}}{q!} [(q - x)(q - x + 1) - x], \quad (6.42)$$

where  $x = 2N\lambda T_p$ . For  $x < 1$  the derivative in (6.41) is positive, and for  $x > N + 1$  it is negative. Therefore,  $p_s \lambda$  has at least one maximum within  $x \in [1, N + 1]$ . In practical scenarios the value of  $p_{s|q}$  for  $k > 0$  is usually small, so that it can be approximated by zero. For a special case where  $p_{s|q>0} = 0$ , (6.41) is zero if  $x = 1$ , and (6.42) is negative, and as a result  $\lambda_{\text{opt}}^{\text{low}} = \frac{1}{2NT_p}$  maximizes  $P_{\text{CT}}^{\text{loc}}$ . This corresponds to a lower bound on the optimal point in a general problem (i.e.,  $p_{s|q>0} \neq 0$ ).

## A3. Cramér Rao lower bound for CTS

The upper bound on the sum operation in (6.29) for CTS is  $\infty$  (note that in practice at most  $\frac{T_T}{T_p}$  packets can be transmitted from an anchor), and this makes the CRB calculation very difficult even if it is implemented numerically. In order to reduce the complexity of the problem, the observation of a sensor node from the  $j$ -th anchor is divided into two parts: Either a sensor node does not receive any packet from this anchor (no information is obtained), or it receives one or more packets. Since the anchor and the sensor node do not move very much during the localization procedure, their distance can be assumed almost constant, and therefore the noise power is the same for all measurements obtained from an anchor. When a sensor node gathers multiple measurements contaminated with independent noise with the same power (diagonal covariance matrix), CRB can be computed with less complexity. We will explain complexity reduction for the first anchor, and then generalize for the other anchors.

Considering the first anchor, each element of the FIM can be calculated in two parts: no correct packet reception, and one or more correct packet receptions from this anchor, which can be formulated as

$$\mathbf{I}(\mathbf{x})_{i,j} = P_1^0 \mathbf{I}(\mathbf{x}|q_1 = 0)_{i,j} + P_1^{>0} \mathbf{I}(\mathbf{x}|q_1 > 0)_{i,j}, \quad (6.43)$$

where  $P_1^0$  is the probability that no packet is received from the first anchor, and  $P_1^{>0} = \sum_{q_1=1}^{\infty} P_1^k$  is the probability that one or more than one packets are received

from the first anchor which depends on the distance between the sensor node and the anchor. The second term in (6.43) can be expanded as

$$\begin{aligned}
 \mathbf{I}(\mathbf{x}|q_1 > 0)_{i,j} &= \frac{1}{P_{\text{loc}}} \sum_{q_N=0}^{Q_N} \cdots \sum_{q_2=0}^{Q_2} \\
 &\quad \text{s.t. } \{q_1, \dots, q_N\} \text{ enable self-localization} \\
 &\quad \left\{ 1\sigma_1^{-2} \frac{\partial f_1}{\partial x_i} \frac{\partial f_1}{\partial x_j} + c_1 + 1\sigma_1^{-4} \frac{\partial \sigma_1^2}{\partial x_i} \frac{\partial \sigma_1^2}{\partial x_j} + c_2 \right\} P_1^1 / P_1^{>0} \Pi_{j=2}^N P_j^{q_j} + \\
 &\quad \left\{ 2\sigma_1^{-2} \frac{\partial f_1}{\partial x_i} \frac{\partial f_1}{\partial x_j} + c_1 + 2\sigma_1^{-4} \frac{\partial \sigma_1^2}{\partial x_i} \frac{\partial \sigma_1^2}{\partial x_j} + c_2 \right\} P_1^2 / P_1^{>0} \Pi_{j=2}^N P_j^{q_j} + \\
 &\quad \vdots \\
 &\quad \left\{ k\sigma_1^{-2} \frac{\partial f_1}{\partial x_i} \frac{\partial f_1}{\partial x_j} + c_1 + k\sigma_1^{-4} \frac{\partial \sigma_1^2}{\partial x_i} \frac{\partial \sigma_1^2}{\partial x_j} + c_2 \right\} P_1^k / P_1^{>0} \Pi_{j=2}^N P_j^{q_j} + \\
 &\quad \vdots
 \end{aligned} \tag{6.44}$$

where  $c_1$  and  $c_2$  are affected only by measurements from the other anchors. Using a simple factorization we have

$$\begin{aligned}
 \mathbf{I}(\mathbf{x}|q_1 > 0)_{i,j} &= \frac{1}{P_{\text{loc}}} \sum_{q_N=0}^{Q_N} \cdots \sum_{q_2=0}^{Q_2} \\
 &\quad \text{s.t. } \{q_1, \dots, q_N\} \text{ enable self-localization} \\
 &\quad \left\{ g_j \left[ \sigma_1^{-2} \frac{\partial f_1}{\partial x_i} \frac{\partial f_1}{\partial x_j} + \sigma_1^{-4} \frac{\partial \sigma_1^2}{\partial x_i} \frac{\partial \sigma_1^2}{\partial x_j} \right] + c_1 + c_2 \right\} \Pi_{j=2}^N P_j^{q_j}
 \end{aligned} \tag{6.45}$$

where

$$g_j = \frac{\sum_{q_j=1}^{\infty} k P_j^k}{\sum_{q_j=1}^{\infty} P_j^k} = \frac{\lambda T_{\text{T}} p_{s|\mathbf{d}}^j}{1 - P_j^0}. \tag{6.46}$$

Now, we define  $\mathbf{a}_{N \times 1}$  with its  $k$ -th element  $a_k$  either zero (if  $q_k = 0$ ) or  $g_j$  (if  $q_k > 0$ ). We also define  $\mathbf{b}_{N \times 1}$  with its  $k$ -th element  $b_k = \left[ \sigma_k^{-2} \frac{\partial f_k}{\partial x_i} \frac{\partial f_k}{\partial x_j} + \sigma_k^{-4} \frac{\partial \sigma_k^2}{\partial x_i} \frac{\partial \sigma_k^2}{\partial x_j} \right]$ . Then, we have

$$\begin{aligned}
 \mathbf{I}(\mathbf{x}|\mathbf{a})_{i,j} &= \frac{1}{P_{\text{loc}}} \\
 &\quad \left( \mathbf{a}^T \mathbf{b} \right) \left( \Pi_{n=1}^{N-n_{\mathbf{a}}} P_{k, a_k=0}^0 \right) \left( \Pi_{n=1}^{n_{\mathbf{a}}} (1 - P_{k, a_k > 0}^0) \right),
 \end{aligned} \tag{6.47}$$

where  $n_{\mathbf{a}}$  is the number of non-zero elements in  $\mathbf{a}$ . Hence, to evaluate  $\mathbf{I}(\mathbf{x})_{i,j}$  for the localizable scenarios only  $\binom{N}{K}$  possible states (different realizations of  $\mathbf{a}$  which lead to localizable scenarios) have to be considered. This number is the same as that of CFS.

---

---

## **PART IV**

---

## **POSTFACE**



# Conclusion and Future Work

In this chapter, we provide the conclusive findings of the thesis, and list some suggestions for future work.

## 7.1 Conclusions

The problem we set out to solve in this thesis was related to underwater acoustic localization in terms of accuracy and localization time. As we mentioned earlier in Chapter 1, underwater acoustic communication is quite challenging, and unique. Low data rate, large propagation delays, and variable sound speed are the unique characteristics of this medium which have to be considered in the designed localization algorithms.

In this thesis, we made it possible to accomplish more accurate underwater acoustic localization by utilizing the sound speed profile of the operating area. We decomposed the complex problem of localization in a medium with varying sound speed into simpler problems explained in Chapters 2 to 4.

In Chapter 2, we considered the problem of target node localization and tracking in an underwater environment with an isogradient SSP. In this chapter, we showed that the traditional terrestrial approaches for localization which assume a constant sound speed for the whole underwater environment are not so accurate for large networks. As the distance between two underwater nodes increases, the inhomogeneous underwater medium upsets the linear dependency of the pair-wise distances to the time of flight. To include this behavior into our model, we directly related the ToF between two underwater nodes to their locations for an isogradient

SSP, and formulated the localization problem as a time-based problem instead of a range-based one. Afterwards, we utilized the Gauss-Newton algorithm and the extended Kalman filter with a proper formulation to solve this problem. It was shown that our proposed algorithms perform better than the algorithms based on a straight-line wave propagation model, especially for large distances. Since an isogradient SSP is not valid for all practical situations, we further extended this for an underwater medium with multiple iso-gradient layers in Chapter 3.

In Chapter 3, we showed that, if the depth information of the unlocalized node is available, then the problem of underwater localization can be converted to the traditional range-based one which is studied extensively in the literature. For an underwater medium with several isogradient sound speed profile layers, we introduced different Lemmas to predict how sound propagates among different layers. It was analytically shown that even without reflection from the surface or the seabed, multiple copies of the transmitted signal can be received by a node. Under such conditions, we proposed an iterative algorithm for the range estimation between two nodes, and we demonstrated that the proposed algorithm attains the Cram r-Rao bound (CRB) and performs superbly in comparison with other existing algorithms which do not consider SSP in estimating node locations. For instance, for a typical deep underwater SSP, we showed that our proposed algorithm can estimate the node positions (located 5 km away from the anchors' center of gravity) with approximately 1 meter accuracy. While the error we get from the algorithms which only consider the average sound speed is around 10 meters.

In Chapters 2 and 3, we assumed that the SSP is known a priori for the localization algorithms. However, in practice it has to be measured in different ways such as using CTD sensors. In Chapter 4, we considered the problem of deriving the CRB on the localization accuracy where noisy sound speed measurements are available instead of the exact SSP. In this chapter, the effect of each measurement noise (depth, ToF and sound speed measurements), on the CRB of range estimation is evaluated analytically. It was shown that, for long distances, the noise power of the depth measurements does not play a significant role in the CRB, while those of the ToF and the sound speed samples are dominant. Over long distances, even with perfect ToF measurements, the range estimation cannot be perfect. For a typical summer shallow underwater SSP, we showed that the error of range estimation due to imperfect SSP estimation can be up to a few meters (2 to 3 meters) at long distances (8 to 10 km). Such error is also location based, and at lower depth is even more (up to 8 m). We showed that for a few sound speed samples at different depths, several factors play a vital role in the CRB such as

the basis functions that the sound speed profile (SSP) is composed of, the actual SSP, the number of sound speed samples, and the positions of the source and the destination.

Other than localization accuracy, it is also important to study how a network accomplishes the localization tasks, how the packets are transmitted from the anchors to the sensor nodes, and how the sensor nodes estimate their location. A localization task might benefit from optimizing network level parameters in terms of localization speed, amount of communication messages, coverage, and even accuracy. In this thesis we showed how the underwater acoustic localization time can be optimized by designing a dedicated medium access layer. This was addressed in Chapters 5 and 6.

In Chapter 5, we formulated the problem of scheduling the localization packets of the anchors in single-channel and multi-channel partially-connected underwater sensor networks. The concept of dynamic multi-channel packet scheduling was introduced in this chapter. This concept proposes to split the existing channel into several subchannels adaptively in order to reduce the scheduling time. We did not only show how the optimal scheduling can be obtained in a UASN, but we also proposed two low-complexity algorithms which perform near optimal, and much better than other alternative solutions such as TDMA-based or position-unaware approaches. The proposed low complexity algorithms can reduce the localization time up to 30% of what is required by the (relatively dense) TDMA based system. Furthermore, through comprehensive simulations, it was revealed that the mean time of the localization task depends on the number of subchannels, localization packet length, the anchors' maximum transmission range, the number of collision-risk neighbors and their modified average distances. We found that, multi-channel scheduling approaches perform better than their single-channel ones especially when the ratio of the packet length to the average pair-wise distance is low. Moreover, we observed that a system that adjusts the number of subchannels dynamically has the highest performance among other position-aware algorithms.

All proposed algorithms in Chapter 5 satisfy the constraint of being collision free. Although the collision-free property of these algorithms makes them energy efficient, they all rely on the decisions made at a central unit. Furthermore, the central unit has to know the relative positions of the anchors to determine their scheduling time for packet transmission. In Chapter 6, we came up with two categories of scheduling algorithms (namely, collision-free and collision tolerant) which do not require a fusion center or relative position information of the anchors. The collision tolerant scheduling algorithm is designed so as to control

the probability of collision to ensure successful localization with a pre-specified reliability. In Chapter 6, this criterion was formulated, solved analytically, and evaluated through different network setups. Furthermore, the collision-free scheduling scheme of Chapter 5, was adapted (formulated and analytically solved) here to work in a network without any fusion center which has no information about the anchors' positions.

In addition, we also proposed a simple Gauss-Newton based localization algorithm for these schemes, and derived their Cramér-Rao lower bounds. Moreover, the performance of the two classes of algorithms (collision-free and collision-tolerant) in terms of the time required for localization was shown to be dependent on the circumstances as explained below. When the ratio of the packet length to the maximum propagation delay is low, as is the case for localization, and the average probability of packet-loss is not close to zero, the collision-tolerant protocol requires less time for localization in comparison with the collision-free one for the same probability of successful localization. In some normal situations ( $5 \times 5$  km operating area, with 6 anchors, and probability of packet loss of 0.1), the collision-tolerant scheme can reduce the location time to 30% of what is required by the collision-free approach. Except for the average energy consumed by the anchors, the collision-tolerant scheme has multiple advantages. The major one is its simplicity of implementation due to the fact that anchors work independently of each other, and as a result the scheme is spatially scalable, with no need for a fusion center. Furthermore, its localization accuracy is always better than that of the collision-free scheme due to multiple receptions of desired packets from anchors. These features make the collision-tolerant localization scheme appealing from a practical implementation view point.

Although in this thesis we addressed several problems related to the underwater acoustic localization, and introduced different solutions and algorithms to overcome them, still several interesting directions for future work exist. In the next section, we highlight a few research topics which can be pointed to by the results of this dissertation.

## 7.2 Suggestions for future works

- **Hybrid underwater networks:** The most difficult problems for Autonomous Underwater Vehicles (AUVs) are often navigation and communications. As mentioned earlier in Chapter 1, wide-band acoustic signals (which cover higher acoustic frequencies, i.e., above 20 kHz) cannot

propagate over long distances due to the high propagation attenuations. Furthermore, due to the propagation delay of acoustic signals, they are not a good candidate for fast communication (data exchanging). However, emerging underwater applications demand more bandwidth, and real-time data which is not realizable physically. For instance, it is still not possible to transmit live video between distant underwater nodes through acoustic signals, and it is very unlikely to happen in the future. Therefore, we have to devise new ways of data communications for underwater networks. One way is to build appropriate infrastructures similar to what we have for terrestrial wireless networks, such as cellular systems with a wired backbone. In this way, each sensor node can communicate with its closest base station (within short distances) at a much higher data rate, and a much lower delay (smaller propagation time). Furthermore, sensor nodes can be equipped with different types of wave generators such as optical, magnetic, and radio transmitters. If nodes are close enough to each other, they can communicate at higher data-rates through optical or even radio signals; otherwise, they can switch to acoustic reception/transmission to establish long-distance communication if they are located far from each other.

- **Joint sound speed profile and position estimation:** In Chapters 2 and 3, we assumed that the localization algorithm is aware of the actual sound speed profile. However, in practical scenarios where the environment is not known a priori, this information is not available. With anchors located at the bottom and the surface of the water, it might be possible to estimate the sound speed profile of the environment by using ToF measurements between the anchors. This can be done by solving a set of integral equations. Moreover, this SSP can be jointly estimated with the node location, leading to joint SSP and position estimation algorithms.
- **General sound speed profile:** In Chapter 3, we approximate the sound speed profile by a piece-wise linear function, and derive appropriate patterns in which a wave can travel between two points in an underwater environment. The concept of wave pattern construction can be extended to a more general form where the profile can be divided into regions where the sound speed monotonically increases or decreases. Instead of working with a set of polynomial equations, we will then deal with a set of equations in different integral forms.

- **Multipath and robust localization:** In Chapter 3, we have shown that the communication channel between two underwater nodes is a multi-path channel. However, we only selected the fastest path in order to perform localization. This can be extended by using all the received paths to increase the localization accuracy (more measurements), or to remove the localization ambiguities (when a depth measurement is not available).
- **Cooperative localization:** The proposed scheduling algorithms in Chapters 5 and 6 cannot directly be used for cooperative localization, unless a localized sensor node participates in the localization process as an anchor before the next scheduling phase. Such a problem where most of the underwater nodes are not under the coverage of the anchors, and hierarchical packet scheduling is required can be addressed as one of the future works of this thesis.
- **Anchors selection:** In a randomly distributed underwater sensor network, we might have more anchors available than the minimum required. Under this condition, we can activate only a portion of the anchors to decrease the energy consumption, and as a result to increase the network localization life time. The problem of anchor selection for such a scenario has not been well-studied in the literature and can be considered as a continuation of our work in Chapters 5 and 6.
- **Simultaneous localization and synchronization:** In an underwater acoustic sensor network, not only the location but the time at which the sensor node measures a phenomenon is important. Due to the large propagation delays and harsh underwater medium, synchronization is not an easy task, and researchers have a tendency to propose algorithms suitable for asynchronous networks. Since localization is a necessary task for most underwater applications, developing algorithms that can jointly perform localization and synchronizations are highly beneficial. Many algorithms such as routing, resource scheduling and so on, can perform better in a synchronized network.

# Bibliography

- [1] C.-F. Huang, P. Gerstoft, and W. S. Hodgkiss, "Effect of ocean sound speed uncertainty on matched-field geoacoustic inversion," *The Journal of the Acoustical Society of America*, vol. 123, no. 6, pp. EL162–EL168, 2008.
- [2] M. B. Porter, "Acoustic models and sonar systems," *IEEE Journal of Oceanic Engineering*, vol. 18, no. 4, pp. 425–437, 1993.
- [3] Sonardyne positioning products. [Online]. Available: <http://www.sonardyne.com/products/positioning/lbl.html>
- [4] planetos positioning products. [Online]. Available: <https://planetos.com/blog/precise-underwater-acoustic-positioning-ocean-data/>
- [5] G. Casalino, A. Turetta, E. Simetti, and A. Caiti, "RT<sup>2</sup>: A real-time ray-tracing method for acoustic distance evaluations among cooperating AUVs," in *Proc. OCEANS 2010 IEEE - Sydney*, May 2010, pp. 1–8.
- [6] C. R. Berger, S. Zhou, P. Willett, and L. Liu, "Stratification effect compensation for improved underwater acoustic ranging," *Signal Processing, IEEE Transactions on*, vol. 56, no. 8, pp. 3779–3783, 2008.
- [7] J. Lloret, S. Sendra, M. Ardid, and J. J. Rodrigues, "Underwater wireless sensor communications in the 2.4 GHz ism frequency band," *Sensors*, vol. 12, no. 4, pp. 4237–4264, 2012.
- [8] M. Chitre, S. Shahabudeen, L. Freitag, and M. Stojanovic, "Recent advances in underwater acoustic communications & networking," in *OCEANS 2008*, vol. 2008. IEEE, 2008, pp. 1–10.
- [9] M. Stojanovic, J. A. Catipovic, and J. G. Proakis, "Phase-coherent digital communications for underwater acoustic channels," *Oceanic Engineering, IEEE Journal of*, vol. 19, no. 1, pp. 100–111, 1994.

- [10] M. Stojanovic and J. Preisig, "Underwater acoustic communication channels: Propagation models and statistical characterization," *Communications Magazine, IEEE*, vol. 47, no. 1, pp. 84–89, 2009.
- [11] F. Frassati, C. Lafon, P.-A. Laurent, and J.-M. Passerieux, "Experimental assessment of OFDM and DSSS modulations for use in littoral waters underwater acoustic communications," in *Oceans 2005-Europe*, vol. 2. IEEE, 2005, pp. 826–831.
- [12] J. J. Zhang, A. Papandreou-Suppappola, B. Gottin, and C. Ioana, "Time-frequency characterization and receiver waveform design for shallow water environments," *Signal Processing, IEEE Transactions on*, vol. 57, no. 8, pp. 2973–2985, 2009.
- [13] L. Wan, H. Zhou, X. Xu, Y. Huang, S. Zhou, Z. Shi, and J.-H. Cui, "Adaptive modulation and coding for underwater acoustic ofdm," *Oceanic Engineering, IEEE Journal of*, vol. 40, no. 2, pp. 327–336, 2015.
- [14] J. Huang, S. Zhou, J. Huang, C. R. Berger, and P. Willett, "Progressive inter-carrier interference equalization for OFDM transmission over time-varying underwater acoustic channels," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 5, no. 8, pp. 1524–1536, 2011.
- [15] Y. Huang, L. Wan, S. Zhou, Z. Wang, and J. Huang, "Comparison of sparse recovery algorithms for channel estimation in underwater acoustic OFDM with data-driven sparsity learning," *Physical Communication*, 2014.
- [16] G. Baiden and Y. Bissiri, "High bandwidth optical networking for underwater untethered telerobotic operation," in *OCEANS 2007*. IEEE, 2007, pp. 1–9.
- [17] B. Gulbahar and O. B. Akan, "A communication theoretical modeling and analysis of underwater magneto-inductive wireless channels," *Wireless Communications, IEEE Transactions on*, vol. 11, no. 9, pp. 3326–3334, 2012.
- [18] M. Emoto, *The hidden messages in water*. Simon and Schuster, 2011.
- [19] M. C. Domingo, "Magnetic induction for underwater wireless communication networks," *Antennas and Propagation, IEEE Transactions on*, vol. 60, no. 6, pp. 2929–2939, 2012.
- [20] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad hoc networks*, vol. 3, no. 3, pp. 257–279, 2005.
- [21] W. A. Kuperman and J. F. Lynch, "Shallow-water acoustics," *Physics Today*, vol. 57, no. 10, pp. 55–61, 2004.
- [22] K. Fang, L. Rugini, and G. Leus, "Block transmissions over doubly selective channels: iterative channel estimation and turbo equalization," *EURASIP Journal on Advances in signal Processing*, vol. 2010, p. 13, 2010.
- [23] P. C. Etter, *Underwater acoustic modeling and simulation*. CRC Press, 2013.

- [24] Kastner research group university of california san diego. [Online]. Available: <http://kastner.ucsd.edu/research/underwater-systems-research/>
- [25] B. G. A., H. G., and R. S. P., "The provision of standards for underwater acoustics at simulated ocean conditions by use of the NPL acoustic pressure vessel." NPL Report CMAM 76, Tech. Rep., Oct. 2001.
- [26] C. Fletcher, J. Rice, R. Creber, and D. Codiga, "Undersea acoustic network operations through a database-oriented server/client interface," in *OCEANS, 2001. MTS/IEEE Conference and Exhibition*, vol. 4. IEEE, 2001, pp. 2071–2075.
- [27] J. Kalwa, "The RACUN-project: Robust acoustic communications in underwater networks an overview," in *OCEANS, 2011 IEEE-Spain*. IEEE, 2011, pp. 1–6.
- [28] B. Tomasi, "JANUS: The genesis, propagation and use of an underwater standard, in," in *European Conference on Underwater Acoustics*, 2010.
- [29] S. Blouin, G. J. Heard, and S. Pecknold, "Autonomy and networking challenges of future underwater systems," in *Electrical and Computer Engineering (CCECE), 2015 IEEE 28th Canadian Conference on*. IEEE, 2015, pp. 1514–1519.
- [30] D. B. Kilfoyle and A. B. Baggeroer, "The state of the art in underwater acoustic telemetry," *Oceanic Engineering, IEEE Journal of*, vol. 25, no. 1, pp. 4–27, 2000.
- [31] J. Almeida, M. Bayat, B. Cardeira, R. Cunha, P. Maurya, A. Pascoal, A. Pereira, M. Rufino, L. Sebastião, C. Silvestre *et al.*, "Cooperative autonomous marine vehicle motion control in the scope of the EU GREX project: theory and practice," in *Oceans 2009-Europe*. IEEE, 2009, pp. 1–10.
- [32] P. J. Sanz, P. Ridao, G. Oliver, G. Casalino, Y. Petillot, C. Silvestre, C. Melchiorri, and A. Turetta, "TRIDENT an european project targeted to increase the autonomy levels for underwater intervention missions," in *Oceans-San Diego, 2013*. IEEE, 2013, pp. 1–10.
- [33] TRIDENT project info. [Online]. Available: <http://www.irs.uji.es/trident/index.html>
- [34] A shoal of underwater robots to protect the environment. [Online]. Available: <http://www.irs.uji.es/trident/index.html>
- [35] Control for coordination of distributed systems (C4C). [Online]. Available: [http://data.jic.quonia.cz/VAN\\_SCHUPPEN.pdf](http://data.jic.quonia.cz/VAN_SCHUPPEN.pdf)
- [36] S. Chatzicristofis, A. Kapoutsis, E. Kosmatopoulos, L. Doitsidis, D. Rovas, and J. Sousa, "The NOPTILUS project: Autonomous multi-AUV navigation for exploration of unknown environments," in *IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles (NGCUV2012)*, vol. 3, 2012, pp. 373–380.

- [37] J. Heidemann, M. Stojanovic, and M. Zorzi, "Underwater sensor networks: applications, advances and challenges," *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 370, no. 1958, pp. 158–175, 2012.
- [38] Y. Shang, W. Ruml, Y. Zhang, and M. P. Fromherz, "Localization from mere connectivity," in *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*. ACM, 2003, pp. 201–212.
- [39] C. Wang and L. Xiao, "Sensor localization under limited measurement capabilities," *Network, IEEE*, vol. 21, no. 3, pp. 16–23, 2007.
- [40] X. Sheng and Y.-H. Hu, "Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks," *Signal Processing, IEEE Transactions on*, vol. 53, no. 1, pp. 44–53, 2005.
- [41] M. M. Noel, P. P. Joshi, and T. C. Jannett, "Improved maximum likelihood estimation of target position in wireless sensor networks using particle swarm optimization," in *Information Technology: New Generations, 2006. ITNG 2006. Third International Conference on*. IEEE, 2006, pp. 274–279.
- [42] N. Patwari, A. O. Hero III, M. Perkins, N. S. Correal, and R. J. O’dea, "Relative location estimation in wireless sensor networks," *Signal Processing, IEEE Transactions on*, vol. 51, no. 8, pp. 2137–2148, 2003.
- [43] R. Zemek, S. Hara, K. Yanagihara, and K.-i. Kitayama, "A joint estimation of target location and channel model parameters in an IEEE 802.15. 4-based wireless sensor network," in *Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on*. IEEE, 2007, pp. 1–5.
- [44] M. Meurer, S. Heilmann, D. Reddy, T. Weber, and P. Baier, "A signature based localization technique relying on covariance matrices of channel impulse responses," in *Proceedings of Workshop on Positioning, Navigation and Communication (WPNC)*, 2005.
- [45] R. Mandelbaum and M. Mintz, "Feature-based localization using fixed ultrasonic transducers," in *Intelligent Robots and Systems 95. Human Robot Interaction and Cooperative Robots, Proceedings. 1995 IEEE/RSJ International Conference on*, vol. 3. IEEE, 1995, pp. 210–215.
- [46] R. Peng and M. L. Sichitiu, "Angle of arrival localization for wireless sensor networks," in *Sensor and Ad Hoc Communications and Networks, 2006. SECON’06. 2006 3rd Annual IEEE Communications Society on*, vol. 1. IEEE, 2006, pp. 374–382.
- [47] V. Chandrasekhar and W. Seah, "An area localization scheme for underwater sensor networks," in *OCEANS 2006-Asia Pacific*. IEEE, 2007, pp. 1–8.

- [48] H. Jamali-Rad, H. Ramezani, and G. Leus, "Sparsity-aware multi-source RSS localization," *Signal Processing*, vol. 101, pp. 174–191, 2014.
- [49] M. Erol-Kantarci, H. T. Mouftah, and S. Oktug, "A survey of architectures and localization techniques for underwater acoustic sensor networks," *Communications Surveys & Tutorials, IEEE*, vol. 13, no. 3, pp. 487–502, 2011.
- [50] K. Langendoen and N. Reijers, "Distributed localization algorithm," *Embedded Systems Handbook*, R. Zurawski (Editor), CRC Press, Boca Raton, FL, 2004.
- [51] R. Diamant, H.-P. Tan, and L. Lampe, "LOS and NLOS classification for underwater acoustic localization," *Mobile Computing, IEEE Transactions on*, vol. 13, no. 2, pp. 311–323, 2014.
- [52] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *Proceedings of the IEEE*, vol. 97, no. 2, pp. 427–450, 2009.
- [53] P. Carroll, S. Zhou, K. Mahmood, H. Zhou, X. Xu, and J.-H. Cui, "On-demand asynchronous localization for underwater sensor networks," in *Oceans, 2012*. IEEE, 2012, pp. 1–4.
- [54] L. Paull, S. Saeedi, M. Seto, and H. Li, "AUV navigation and localization: A review," *Oceanic Engineering, IEEE Journal of*, vol. 39, no. 1, pp. 131–149, 2014.
- [55] V. Chandrasekhar, W. K. Seah, Y. S. Choo, and H. V. Ee, "Localization in underwater sensor networks: survey and challenges," in *Proceedings of the 1st ACM international workshop on Underwater networks*. ACM, 2006, pp. 33–40.
- [56] G. Han, J. Jiang, L. Shu, Y. Xu, and F. Wang, "Localization algorithms of underwater wireless sensor networks: A survey," *Sensors*, vol. 12, no. 2, pp. 2026–2061, 2012.
- [57] O. Hegrenas, K. Gade, O. K. Hagen, and P. E. Hagen, "Underwater transponder positioning and navigation of autonomous underwater vehicles," in *OCEANS 2009, MTS/IEEE Biloxi-Marine Technology for Our Future: Global and Local Challenges*. IEEE, 2009, pp. 1–7.
- [58] M. Erol, L. F. Vieira, and M. Gerla, "Localization with dive'n'rise (DNR) beacons for underwater acoustic sensor networks," in *Proceedings of the second workshop on Underwater networks*. ACM, 2007, pp. 97–100.
- [59] M. Erol, L. F. Vieira, A. Caruso, F. Paparella, M. Gerla, and S. Oktug, "Multi stage underwater sensor localization using mobile beacons," in *Sensor Technologies and Applications, 2008. SENSORCOMM'08. Second International Conference on*. IEEE, 2008, pp. 710–714.
- [60] L. Bin, C. Hong-Yang, and Z. Zi-Guo, "Asymmetrical round trip based synchronization-free localization in," *IEEE Transactions on Wireless Communications*, vol. 9, no. 11, pp. 3532–3542, 2010.

- [61] A.-K. Othman, "GPS-less localization protocol for underwater acoustic networks," in *Wireless and Optical Communications Networks, 2008. WOCN'08. 5th IFIP International Conference on*. IEEE, 2008, pp. 1–6.
- [62] M. Erol, L. F. Vieira, and M. Gerla, "AUV-aided localization for underwater sensor networks," in *Wireless Algorithms, Systems and Applications, 2007. WASA 2007. International Conference on*. IEEE, 2007, pp. 44–54.
- [63] M. Waldmeyer, H.-P. Tan, and W. K. Seah, "Multi-stage AUV-aided localization for underwater wireless sensor networks," in *Advanced Information Networking and Applications (WAINA), 2011 IEEE Workshops of International Conference on*. IEEE, 2011, pp. 908–913.
- [64] A. Y. Teymorian, W. Cheng, L. Ma, and X. Cheng, "An underwater positioning scheme for 3D acoustic sensor networks," in *Proc. of Second Workshop on Underwater Networks, Montreal, Quebec, Canada, 2007*.
- [65] H.-P. Tan, Z. A. Eu, and W. K. Seah, "An enhanced underwater positioning system to support deepwater installations," in *OCEANS 2009, MTS/IEEE Biloxi-Marine Technology for Our Future: Global and Local Challenges*. IEEE, 2009, pp. 1–8.
- [66] H.-P. Tan, A. F. Gabor, Z. A. Eu, and W. K. Seah, "A wide coverage positioning system (WPS) for underwater localization," in *Communications (ICC), 2010 IEEE International Conference on*. IEEE, 2010, pp. 1–5.
- [67] D. Mirza and C. Schurgers, "Collaborative localization for fleets of underwater drifters," in *OCEANS 2007*. IEEE, 2007, pp. 1–6.
- [68] M. Erol-Kantarci, H. T. Mouftah, and S. Oktug, "Localization techniques for underwater acoustic sensor networks," *Communications Magazine, IEEE*, vol. 48, no. 12, pp. 152–158, 2010.
- [69] H.-P. Tan, R. Diamant, W. K. Seah, and M. Waldmeyer, "A survey of techniques and challenges in underwater localization," *Ocean Engineering*, vol. 38, no. 14, pp. 1663–1676, 2011.
- [70] X. Tan and J. Li, "Cooperative positioning in underwater sensor networks," *Signal Processing, IEEE Transactions on*, vol. 58, no. 11, pp. 5860–5871, 2010.
- [71] X. Cheng, H. Shu, Q. Liang, and D. H.-C. Du, "Silent positioning in underwater acoustic sensor networks," *Vehicular Technology, IEEE Transactions on*, vol. 57, no. 3, pp. 1756–1766, 2008.
- [72] M. T. Isik and O. B. Akan, "A three dimensional localization algorithm for underwater acoustic sensor networks," *Wireless Communications, IEEE Transactions on*, vol. 8, no. 9, pp. 4457–4463, 2009.
- [73] K. V. Mackenzie, "Nine-term equation for sound speed in the oceans," *The Journal of the Acoustical Society of America*, vol. 70, no. 3, pp. 807–812, 1981.

- [74] P. Ameer and L. Jacob, "Localization using ray tracing for underwater acoustic sensor networks," *Communications Letters, IEEE*, vol. 14, no. 10, pp. 930–932, 2010.
- [75] K. Y. Foo and P. R. Atkins, "A relative-localization algorithm using incomplete pairwise distance measurements for underwater applications," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, p. 11, 2010.
- [76] P.-M. Lee, B.-H. Jun, J.-H. Li, H. T. Choi, K. Kim, S.-M. Kim, C.-M. Lee, S.-C. Han, B.-M. Gu, S.-R. Lee *et al.*, "Navigation and control system of a deep-sea unmanned underwater vehicle'HEMIRE'," in *OCEANS 2006-Asia Pacific*. IEEE, 2007, pp. 1–8.
- [77] N. H. Kussat, C. D. Chadwell, and R. Zimmerman, "Absolute positioning of an autonomous underwater vehicle using GPS and acoustic measurements," *Oceanic Engineering, IEEE Journal of*, vol. 30, no. 1, pp. 153–164, 2005.
- [78] P.-M. Lee, S.-M. Kim, B.-H. Jeon, H. T. Choi, and C.-M. Lee, "Improvement on an inertial-doppler navigation system of underwater vehicles using a complementary range sonar," in *Underwater Technology, 2004. UT'04. 2004 International Symposium on*. IEEE, 2004, pp. 133–138.
- [79] W. Cheng, A. Thaeler, X. Cheng, F. Liu, X. Lu, and Z. Lu, "Time-synchronization free localization in large scale underwater acoustic sensor networks," in *Distributed Computing Systems Workshops, 2009. ICDCS Workshops' 09. 29th IEEE International Conference on*. IEEE, 2009, pp. 80–87.
- [80] L. Liu, Y. Xiao, and J. Zhang, "A linear time synchronization algorithm for underwater wireless sensor networks," in *Communications, 2009. ICC'09. IEEE International Conference on*. IEEE, 2009, pp. 1–5.
- [81] M. J. Hahn, "Undersea navigation via a distributed acoustic communications network," DTIC Document, Tech. Rep., 2005.
- [82] S. K. Sengijpta, "Fundamentals of statistical signal processing: Estimation theory," *Technometrics*, vol. 37, no. 4, pp. 465–466, 1995.
- [83] P. Tichavsky, C. H. Muravchik, and A. Nehorai, "Posterior Cramér-Rao bounds for discrete-time nonlinear filtering," *Signal Processing, IEEE Transactions on*, vol. 46, no. 5, pp. 1386–1396, May 1998.
- [84] L. Arienzo and M. Longo, "Posterior Cramér-Rao bound for range-based target tracking in sensor networks," in *Proc. IEEE/SP 15th Workshop Statistical Signal Processing SSP '09*, Aug. 2009, pp. 541–544.
- [85] Ocean acoustics library. [Online]. Available: <http://oalib.hlsresearch.com/Rays/index.html>

- [86] M. B. Porter, "The bellhop manual and users guide: Preliminary draft," *Heat, Light, and Sound Research, Inc., La Jolla, CA, USA, Tech. Rep.*, 2011.
- [87] G. Mao, B. Fidan, and B. D. Anderson, "Wireless sensor network localization techniques," *Computer networks*, vol. 51, no. 10, pp. 2529–2553, 2007.
- [88] B. Roberts, "Waves in inhomogeneous media," in *ESA Special Publication*, vol. 220, 1984, pp. 137–145.
- [89] H. Ramezani, H. Jamali-Rad, and G. Leus, "Localization and tracking of a mobile target for an isogradient sound speed profile," in *Communications (ICC), 2012 IEEE International Conference on*. IEEE, 2012, pp. 3654–3658.
- [90] A. B. Coppins, "Simple equations for the speed of sound in neptunian waters," *The Journal of the Acoustical Society of America*, vol. 69, no. 3, pp. 862–863, 1981.
- [91] A. Y. Teymorian, W. Cheng, L. Ma, X. Cheng, X. Lu, and Z. Lu, "3D underwater sensor network localization," *Mobile Computing, IEEE Transactions on*, vol. 8, no. 12, pp. 1610–1621, 2009.
- [92] R. Cardinali, L. De Nardis, M. Di Benedetto, and P. Lombardo, "UWB ranging accuracy in high-and low-data-rate applications," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 54, no. 4, pp. 1865–1875, 2006.
- [93] Y. Wang, G. Leus, and A.-J. van der Veen, "Cramér-rao bound for range estimation," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*. IEEE, 2009, pp. 3301–3304.
- [94] H. Ramezani and G. Leus, "Ranging in an underwater medium with multiple isogradient sound speed profile layers," *Sensors*, vol. 12, no. 3, pp. 2996–3017, 2012.
- [95] O. Carriere, J.-P. Hermand, M. Meyer, and J. Candy, "Dynamic estimation of the sound-speed profile from broadband acoustic measurements," in *OCEANS 2007-Europe*. IEEE, 2007, pp. 1–6.
- [96] S. M. Kay, "Fundamentals of statistical signal processing: Detection theory, vol. 2," 1998.
- [97] H. Ramezani, H. Jamali-Rad, and G. Leus, "Target localization and tracking for an isogradient sound speed profile," *Signal Processing, IEEE Transactions on*, vol. 61, no. 6, pp. 1434–1446, 2013.
- [98] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU Press, 2012, vol. 3.
- [99] E. Gallimore, J. Partan, I. Vaughn, S. Singh, J. Shusta, and L. Freitag, "The WHOI micromodem-2: A scalable system for acoustic communications and networking," in *OCEANS 2010*. IEEE, 2010, pp. 1–7.

- [100] M. Molins and M. Stojanovic, "Slotted FAMA: a MAC protocol for underwater acoustic networks," in *OCEANS 2006-Asia Pacific*. IEEE, 2007, pp. 1–7.
- [101] B. Peleato and M. Stojanovic, "Distance aware collision avoidance protocol for ad-hoc underwater acoustic sensor networks," *Communications Letters, IEEE*, vol. 11, no. 12, pp. 1025–1027, 2007.
- [102] L. Hong, F. Hong, Z. Guo, and X. Yang, "A TDMA-based MAC protocol in underwater sensor networks," in *Wireless Communications, Networking and Mobile Computing, 2008. WiCOM'08. 4th International Conference on*. IEEE, 2008, pp. 1–4.
- [103] A. Syed, W. Ye, J. Heidemann *et al.*, "T-Lohi: A new class of MAC protocols for underwater acoustic sensor networks," in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*. IEEE, 2008.
- [104] —, "Comparison and evaluation of the T-Lohi MAC for underwater acoustic sensor networks," *Selected Areas in Communications, IEEE Journal on*, vol. 26, no. 9, pp. 1731–1743, 2008.
- [105] Z. Peng, Y. Zhu, Z. Zhou, Z. Guo, and J.-H. Cui, "COPE-MAC: a contention-based medium access control protocol with parallel reservation for underwater acoustic networks," in *OCEANS 2010 IEEE-Sydney*. IEEE, 2010, pp. 1–10.
- [106] C.-C. Hsu, K.-F. Lai, C.-F. Chou, and K. C.-J. Lin, "ST-MAC: Spatial-temporal MAC scheduling for underwater sensor networks," in *INFOCOM 2009, IEEE*. IEEE, 2009, pp. 1827–1835.
- [107] K. Kredo, P. Djukic, P. Mohapatra *et al.*, "STUMP: Exploiting position diversity in the staggered TDMA underwater MAC protocol," in *INFOCOM 2009, IEEE*. IEEE, 2009, pp. 2961–2965.
- [108] W.-H. Liao and C.-C. Huang, "SF-MAC: A spatially fair MAC protocol for underwater acoustic sensor networks," *Sensors Journal, IEEE*, vol. 12, no. 6, pp. 1686–1694, 2012.
- [109] L. S. Committee *et al.*, "Part 15.4: wireless medium access control (MAC) and physical layer (phy) specifications for low-rate wireless personal area networks (lr-wpans)," *IEEE Computer Society*, 2003.
- [110] J.-P. Kim, H.-P. Tan, and H.-S. Cho, "Impact of MAC on localization in large-scale seabed sensor networks," in *Advanced Information Networking and Applications (AINA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 391–396.
- [111] Y.-J. Chen and H.-L. Wang, "Ordered CSMA: a collision-free MAC protocol for underwater acoustic networks," in *OCEANS 2007*. IEEE, 2007, pp. 1–6.

- [112] P. Soonchul and L. Jaesung, "A parallel transmission scheme for all-to-all broadcast in underwater sensor networks," *IEICE transactions on communications*, vol. 93, no. 9, pp. 2309–2315, 2010.
- [113] P. Nicopolitidis, G. I. Papadimitriou, and A. S. Pomportsis, "Adaptive data broadcasting in underwater wireless networks," *Oceanic Engineering, IEEE Journal of*, vol. 35, no. 3, pp. 623–634, 2010.
- [114] P. Casari, M. Rossi, and M. Zorzi, "Fountain codes and their application to broadcasting in underwater networks: Performance modeling and relevant tradeoffs," in *Proceedings of the third ACM international workshop on Underwater Networks*. ACM, 2008, pp. 11–18.
- [115] W. van Kleunen, N. Meratnia, and P. J. Havinga, "Scheduled MAC in beacon overlay networks for underwater localization and time-synchronization," in *Proceedings of the Sixth ACM International Workshop on Underwater Networks*. ACM, 2011, p. 6.
- [116] W. Van Kleunen, N. Meratnia, and P. J. Havinga, "A set of simplified scheduling constraints for underwater acoustic MAC scheduling," in *Advanced Information Networking and Applications (WAINA), 2011 IEEE Workshops of International Conference on*. IEEE, 2011, pp. 902–907.
- [117] W. van Kleunen, N. Meratnia, and P. J. Havinga, "MAC scheduling in large-scale underwater acoustic networks," in *Wireless Information Networks and Systems (WINSYS), 2011 Proceedings of the International Conference on*. IEEE, 2011, pp. 27–34.
- [118] D. Mirza, F. Lu, and C. Schurgers, "Efficient broadcast MAC for underwater networks," *Proceedings of ACM WUWNet, Berkeley, CA, USA*, 2009.
- [119] S. Shahabudeen, M. Chitre, and M. Motani, "A multi-channel MAC protocol for AUV networks," in *IEEE Oceans*, vol. 7, 2007.
- [120] D. Pompili, T. Melodia, and I. F. Akyildiz, "A distributed CDMA medium access control for underwater acoustic sensor networks," in *Proc. of Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, 2007, pp. 63–70.
- [121] Z. Zhou, Z. Peng, J.-H. Cui, and Z. Shi, "Analyzing multi-channel MAC protocols for underwater acoustic sensor networks," UCONN CSE Technical Report: UbiNet-TR08-02, August 2008., Tech. Rep., 2009.
- [122] H. Ramezani and G. Leus, "DMC-MAC: Dynamic multi-channel MAC in underwater acoustic networks," in *Signal Processing Conference (EUSIPCO), 2013 Proceedings of the 21st European*. IEEE, 2013, pp. 1–5.
- [123] C. Gentile, N. Alsindi, R. Raulefs, and C. Teolis, "Ranging and localization in harsh multipath environments," in *Geolocation Techniques*. Springer, 2013, pp. 17–57.

- [124] R. Ramaswami and K. K. Parhi, "Distributed scheduling of broadcasts in a radio network," in *INFOCOM'89. Proceedings of the Eighth Annual Joint Conference of the IEEE Computer and Communications Societies. Technology: Emerging or Converging, IEEE*. IEEE, 1989, pp. 497–504.
- [125] C. R. Berger, S. Zhou, J. C. Preisig, and P. Willett, "Sparse channel estimation for multicarrier underwater acoustic communication: From subspace methods to compressed sensing," *Signal Processing, IEEE transactions on*, vol. 58, no. 3, pp. 1708–1721, 2010.
- [126] Z. Zhou, Z. Peng, J.-H. Cui, Z. Shi, and A. C. Bagtzoglou, "Scalable localization with mobility prediction for underwater sensor networks," *Mobile Computing, IEEE Transactions on*, vol. 10, no. 3, pp. 335–348, 2011.
- [127] H. Kenneth, O. Anders, and M. Marcus, "User's guide for TOMLAB/CPLEX v12.1."
- [128] P. Kuakowski, J. Vales-Alonso, E. Egea-Lpez, W. Ludwin, and J. Garca-Haro, "Angle-of-arrival localization based on antenna arrays for wireless sensor networks," *Computers and Electrical Engineering*, vol. 36, no. 6, pp. 1181 – 1186, 2010.
- [129] S. P. Chepuri, G. Leus, and A.-J. Van der Veen, "Sparsity-exploiting anchor placement for localization in sensor networks," in *Signal Processing Conference (EUSIPCO), 2013 Proceedings of the 21st European*. IEEE, 2013, pp. 1–5.
- [130] R. Stuart, "Acoustic digital spread spectrum: An enabling technology," *Sea Technology*, vol. 46, no. 10, pp. 15–20, 2005.
- [131] X. Cheng, H. Shu, and Q. Liang, "A range-difference based self-positioning scheme for underwater acoustic sensor networks," in *Wireless Algorithms, Systems and Applications, 2007. WASA 2007. International Conference on*. IEEE, 2007, pp. 38–43.
- [132] M. K. Watfa, T. Nsouli, M. Al-Ayache, and O. Ayyash, "Reactive localization in underwater wireless sensor networks," in *International Conference on Computer and Network Technology (ICCNT), 2010 Second*. IEEE, 2010, pp. 244–248.
- [133] S. Shahabudeen, M. Motani, and M. Chitre, "Analysis of a high-performance mac protocol for underwater acoustic networks," *Oceanic Engineering, IEEE Journal of*, vol. 39, no. 1, pp. 74–89, 2014.
- [134] H. Ramezani and G. Leus, "L-MAC: Localization packet scheduling for an underwater acoustic sensor network," *published in the Proc. of the International Conference on Communications, ICC2013*, pp. 1459–1463, 2013.
- [135] F. Fazel, M. Fazel, and M. Stojanovic, "Random access compressed sensing over fading and noisy communication channels," *IEEE Transactions on Wireless Communications*, vol. 12, no. 5, pp. 2114–2125, 2013.

- [136] M. Stojanovic, "On the relationship between capacity and distance in an underwater acoustic communication channel," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 11, no. 4, pp. 34–43, Oct. 2007.
- [137] H. Jamali-Rad, H. Ramezani, and G. Leus, "Cooperative localization in partially connected mobile wireless sensor networks using geometric link reconstruction," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE, 2012, pp. 2633–2636.
- [138] H. Ramezani, F. Fazel, M. Stojanovic, and G. Leus, "Packet scheduling for underwater acoustic sensor network localization," in *Communications Workshops (ICC), 2014 IEEE International Conference on*. IEEE, 2014, pp. 108–113.
- [139] Evologics, underwater acoustic modems, S2CR series. [Online]. Available: [http://www.evologics.de/en/products/acoustics/s2cr\\_12\\_24.html](http://www.evologics.de/en/products/acoustics/s2cr_12_24.html)

# Samenvatting

In het begin van de 21ste eeuw had telecommunicatie als doel iedereen te verbinden, altijd en overal. Nu, met de grote vooruitgang in de elektronica en de communicatietechnologie denkt men zelfs een stap verder en probeert men om niet enkel personen maar alle dingen te verbinden, altijd en overal (internet der dingen). Deze droom zou niet mogelijk zijn zonder de recente ontwikkelingen in sensornetwerken en het massale gebruik van deze technologie in ons alledaags leven. Maar om dit doel te bereiken worden we toch geconfronteerd met fysieke en technologische uitdagingen, vooral wanneer men denkt aan ruimte- of onderwatertoepassingen. In dit proefschrift bestuderen we het fundamentele probleem van sensorlokalisatie in onderwaternetwerken, een onmiskenbare taak voor elk sensornetwerk.

In het eerste deel van dit proefschrift worden de belangrijkste uitdagingen van akoestische onderwatercommunicatie doorlopen, alsook hun effect op lokalisatietechnieken. Verder wordt besproken hoe deze lokalisatiealgoritmes kunnen worden onderverdeeld in verschillende groepen en hoe ze kunnen worden beoordeeld op basis van verschillende criteria.

Het tweede deel van dit proefschrift richt zich op de ontwikkeling van nauwkeurige lokalisatiealgoritmes voor een medium met een variabele geluidssnelheid. De geluidssnelheid wordt gemodelleerd via een lineaire of stuksgewijs lineaire functie van de diepte en op basis hiervan worden optimale lokalisatiealgoritmes ontwikkeld. Voor grote netwerken presteren deze algoritmes veel beter dan traditionele technieken. De grote propagatievertraging van akoestische golven speelt een belangrijke rol in de tijd die nodig is om sensoren in een onderwaternetwerk te lokaliseren. Om deze tijd te minimaliseren worden in het derde deel van dit proefschrift twee communicatieprotocollen voorgesteld, een

protocol waarbij datapakketten niet op elkaar botsen en een protocol dat robuust is tegen zulke botsingen. Via een gedetailleerde analyse en numerieke resultaten tonen we aan dat een lokalisatiealgoritme veel voordeel kan halen uit een optimaal communicatieprotocol.

# Acknowledgments

It was an honor for me to conduct my Ph.D. research at Circuit and System (CAS) group of the Technical University of Delft (TU Delft), one of the best research centres which has already proved its positive impacts on academia and industry developments.

I would like to express my first and warmest gratitude to professor Geert Leus for supervising me through my Ph.D. As a researcher you like to be supervised by a patient, smart, keen observer, and a visionary person. Geert fulfilled all of them during my research as a Ph.D. student. Having discussions with him are considered as some of the proudest moments of my carrier. Not only I learned about the beauty and elegance of signal processing from him, but also I was inspired by his attitude and social life especially during the business trips we had together. Geert, thanks for giving me the opportunity to work alongside you, and observe the intricacies that make you a likable and successful supervisor.

Another important factor that makes a research group fruitful and constructive is the management. Thanks to Professor Alle-Jan van der Veen for his smooth management of the group. He is a big figure in signal processing with many years of academic experiences, and that is why he exactly knows what are the requirements of a researcher in this filed. As a member of CAS, I could work within a well-organized and motivative center without any deficiencies.

I would like to thank all the people of the NOPTILUS projects who supported us financially, and gave us this opportunity to express out ideas worldwide in bi-annual project meetings, and various top conferences.

I would also thank professor Milica Stojanovic for her help and guidance during my visit at Northeastern University. With her brilliant background in acoustic communications, we could publish outstanding results together through a joint

collaboration of our research group and hers. It was also a great pleasure for me to work with the people of Communication and Digital Signal Processing (CDSP) laboratory of Northeastern University. These friendly people who helped me a lot to settle in a new city, and get along with the new environment.

I would like to thank all of my colleagues and friends who accompanied me during this journey, people who have always been ready to discuss any thing with me ranging from academia to life matters. Thanks for their advices and recommendations, I could handle living as an international inside this beautiful country, the Netherlands.

Last but not least, thanks to my family who always supports me in bitter and sweet events of my life.

Hamid Ramezani,  
31-Dec-2015

# Curriculum Vitae

Hamid Ramezani was born in Tehran, Iran. He received the bachelors degree (B.Sc.) in Electrical Engineering from University of Tehran, Tehran, Iran, in Oct. 2004, and his master of science degree (M.Sc.) in Communication systems from Iran University of Science and Technology (IUST), Tehran, Iran, in March 2007. After working for four years in industry for several companies, he decided to pursue his study, and in May 2011 he joined the Circuit and System (CAS) group at Technical University of Delft, Delft the Netherlands. In Sep. 2013, he visited the Communication and Digital Signal Processing (CDSP) laboratory of Northeastern University for five months. In Jan. 2015, he joined Catena Group, an international group of innovative companies, experts in design of Integrated Circuits and independent System IP, where he has been working since then as a system engineer. He has served as a reviewer for several journals and major conferences. His research interests include signal processing, communications protocols and algorithms in physical and network layers of wireless networks.



# Publications during Ph.D

## Journals

- $\mathcal{J}_5$  **Hamid Ramezani**, and Geert Leus. “Localization Packet Scheduling for Underwater Acoustic Sensor networks.” IEEE Journal on Selected Areas in Communications, vol. 33 , no. 7, May 2015.
- $\mathcal{J}_4$  **Hamid Ramezani**, Fatemeh Fazel, Milica Stojanovic, and Geert Leus. “Collision Tolerant and Collision Free Packet Scheduling for Underwater Acoustic Localization.” IEEE Transactions on Wireless Communications, vol. 14, no. 5, pp. 2584-2595, Jan 2015.
- $\mathcal{J}_3$  Hadi Jamali-Rad, **Hamid Ramezani**, and Geert Leus, Sparsity-Aware Multi-Source RSS Localization, Elsevier Signal Processing, Vol. 101, pp. 174-191, Aug. 2014
- $\mathcal{J}_2$  **Hamid Ramezani**, and Geert Leus. “Ranging in an underwater medium with multiple isogradient sound speed profile layers.” A special issue of Sensors; Underwater Sensor Nodes and Underwater Sensor Networks, Feb. 2012.
- $\mathcal{J}_1$  **Hamid Ramezani**, Hadi Jamali-Rad, and Geert Leus. “Target localization and tracking for an isogradient sound speed profile.” Signal Processing, IEEE Transactions on 61.6 (2013): 1434-1446.

## Conferences

- $\mathcal{C}_{10}$  **Hamid Ramezani**, Raj Thilak Rajan, and Geert Leus. “Cramer Rao Lower Bound for Underwater Range Estimation with Noisy Sound Speed Profile.”

- 49th Annual Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, California November 8-11, 2014.
- $\mathcal{C}_9$  **Hamid Ramezani** Fatemeh Fazel, Milica Stojanovic, and Geert Leus. Packet Scheduling for Underwater Acoustic Sensor Network Localization., accepted in the Proceeding of IEEE ICC 2014 Workshop on Advances in Network Localization and Navigation (ANLN), 10-14 June 2014, Sydney, Australia.
- $\mathcal{C}_8$  **Hamid Ramezani**, and Geert Leus. Dynamic Multi-Channel Packet Scheduling in an Underwater Acoustic Sensor Network, accepted in Proc. of the Asilomar Conference on Signals, systems, and Computers (Asilomar 2013), Pacific Grove, California, USA, November 2013.
- $\mathcal{C}_7$  **Hamid Ramezani**, and Geert Leus. DMC-MAC: Dynamic Multi-channel MAC in Underwater Acoustic Networks, accepted in the European Signal Processing Conference (EUSIPCO'13), Marrakech, Morocco, Sep. 2013.
- $\mathcal{C}_6$  Hadi Jamali-Rad, **Hamid Ramezani**, and G. Leus, Blind Sparsity-Aware Multi-Source Localization, accepted in the European Signal Processing Conference (EUSIPCO'13), Marrakech, Morocco.
- $\mathcal{C}_5$  **Hamid Ramezani**, and Geert Leus. L-MAC: Localization packet scheduling for an underwater acoustic sensor network, accepted in Proc. of the International Conference on Communications (ICC2013), Budapest, Hungary, June 2013.
- $\mathcal{C}_4$  Hadi Jamali-Rad, **Hamid Ramezani**, and Geert Leus, Sparse Multi-Target Localization using Cooperative Access Points, in Proc. of IEEE SAM, pp. 353-356, Hoboken, NJ, June 2012.
- $\mathcal{C}_3$  **Hamid Ramezani**, Hadi Jamali-Rad, and Geert Leus. "Localization and tracking of a mobile target for an isogradient sound speed profile." Communications (ICC), 2012 IEEE International Conference on. IEEE, June 2012.
- $\mathcal{C}_2$  **Hamid Ramezani**, and Geert Leus. "Accurate Ranging in a Stratified Underwater Medium with Multiple Iso-gradient Sound Speed Profile Layers." Navigation, Guidance and Control of Underwater Vehicles. Vol. 3. No. 1. April 2012.

- $\mathcal{C}_1$  Hadi Jamali-Rad, **Hamid Ramezani**, and Geert Leus, Cooperative Localization in Partially Connected Mobile Wireless Sensor Networks Using Geometric Link Reconstruction, in Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2633-2636, Kyoto, Japan, March 2012.

## Technical Reports

- $\mathcal{T}_2$  **Hamid Ramezani**, and Geert Leus. “Addendum for Deliverable 3.1: Synergetic/Cognitive Sonar AUV Communications ”, Technical Report Deliverable 3.1, NOPTILUS Project, Nov. 2013.
- $\mathcal{T}_1$  **Hamid Ramezani**, and Geert Leus. “Synergetic/Cognitive Sonar AUV Communications”, Technical Report Deliverable 3.1, NOPTILUS Project, Oct. 2012.

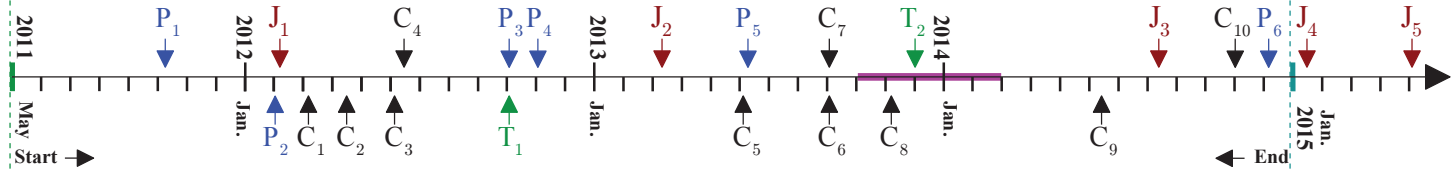


Figure 7.1: The chronological order of our contributions during the Ph.D. based on the date of the conferences, published papers or technical reports. **C** stands for conference paper, **J** for journal paper, **P** for project meeting, and **T** for technical report. The purple line shows the duration I visited Professor *Milica Stojanovic* at Northeastern University.

# Glossary

## Acronyms

|              |                                             |
|--------------|---------------------------------------------|
| <b>AAB</b>   | All-to-All Broadcasting                     |
| <b>ACK</b>   | Acknowledgment                              |
| <b>ALS</b>   | Area Localization Scheme                    |
| <b>AoA</b>   | Angle of Arrival                            |
| <b>APS</b>   | Acoustic Pressure Vessel                    |
| <b>ARTL</b>  | Asymmetrical Round Trip based Localization  |
| <b>ASK</b>   | Amplitude-shift keying                      |
| <b>AUV</b>   | Autonomous Underwater Vehicle               |
| <b>B-MAC</b> | Broadcast MAC                               |
| <b>BPSK</b>  | Binary Phase Shift Keying                   |
| <b>CCK</b>   | Complementary code keying                   |
| <b>CDMA</b>  | Code Division Multiple Access               |
| <b>CFS</b>   | Collision-Free Scheme                       |
| <b>CL</b>    | Collaborative Localization                  |
| <b>CRB</b>   | Cramér-Rao bound                            |
| <b>CRC</b>   | Cyclic Redundancy Check                     |
| <b>CSMA</b>  | Carrier Sense Multiple Access               |
| <b>CTD</b>   | Conductivity-Temperature-Depth              |
| <b>CTS</b>   | Collision-Tolerant Scheme                   |
| <b>DACAP</b> | Distance Aware Collision Avoidance Protocol |
| <b>DDN</b>   | Distance-Dependent Noise                    |
| <b>DIN</b>   | Distance-Independent Noise                  |

|                |                                            |
|----------------|--------------------------------------------|
| <b>DMC</b>     | Dynamic Multi-Channel                      |
| <b>DNR</b>     | Dive and Rise                              |
| <b>DVL</b>     | Doppler Velocity Log                       |
| <b>ECC</b>     | Error-Correcting Code                      |
| <b>EKF</b>     | Extended Kalman Filter                     |
| <b>EU</b>      | European                                   |
| <b>FAMA</b>    | Floor Acquisition Multiple Access          |
| <b>FH</b>      | Frequency Hopping                          |
| <b>FIM</b>     | Fisher Information Matrix                  |
| <b>FSK</b>     | Frequency-Shift Keying                     |
| <b>GNA</b>     | Gauss-Newton algorithm                     |
| <b>GPS</b>     | Global Positioning System                  |
| <b>IMU</b>     | Inertial Navigation System                 |
| <b>KF</b>      | Kalman Filter                              |
| <b>LBL</b>     | Long Baseline                              |
| <b>LDB</b>     | Localization with Directional Beacons      |
| <b>L-MAC</b>   | Localization MAC                           |
| <b>LOS</b>     | Line-of-Sight                              |
| <b>LS</b>      | Least Square                               |
| <b>LUT</b>     | Lookup Table                               |
| <b>MAC</b>     | Medium Access Control                      |
| <b>MDS</b>     | Multi-Dimensional Scaling                  |
| <b>MILP</b>    | Mixed Integer Linear Program               |
| <b>MIMO</b>    | Multiple-Input and Multiple-Output         |
| <b>MS-AUV</b>  | Multi-Stage AUV-Aided Localization         |
| <b>MSE</b>     | Mean Squared Error                         |
| <b>NACK</b>    | Negative Acknowledgment                    |
| <b>NCTS</b>    | Not Clear to Send                          |
| <b>NDLP</b>    | Node Discovery and Localization Protocol   |
| <b>NP-hard</b> | Non-Deterministic Polynomial-Time hard     |
| <b>OCSMA</b>   | Ordered Carrier Sense Multiple Access      |
| <b>OFDM</b>    | Orthogonal Frequency division multiplexing |
| <b>PCRB</b>    | Posterior CRB                              |
| <b>PPM</b>     | Pulse-position modulation                  |
| <b>QPSK</b>    | Quadrature Phase Shift Keying              |
| <b>RMSE</b>    | Root Mean Squared Error                    |
| <b>ROV</b>     | Remote Operated Vehicle                    |

|               |                                    |
|---------------|------------------------------------|
| <b>RRT</b>    | Round-Trip-Time                    |
| <b>RSS</b>    | Received Signal Strength           |
| <b>RT</b>     | Ray Tracing                        |
| <b>RTS</b>    | Request to Send                    |
| <b>SBL</b>    | Short Baseline                     |
| <b>SSP</b>    | Sound Speed Profile                |
| <b>TB-MAC</b> | Tone Broadcast MAC protocol        |
| <b>TDMA</b>   | Time Division Multiple Access      |
| <b>TDoA</b>   | Time-Difference-of-Arrival         |
| <b>ToA</b>    | Time-of-Arrival                    |
| <b>ToF</b>    | Time-of-Flight                     |
| <b>UAN</b>    | Underwater Acoustic Network        |
| <b>UASN</b>   | Underwater Acoustic Sensor Network |
| <b>UPS</b>    | Underwater Positioning Scheme      |
| <b>USBL</b>   | Ultra Short Baseline               |
| <b>WPS</b>    | Wide Coverage Positioning          |
| <b>WSN</b>    | Wireless Sensor Network            |
| <b>w.r.t.</b> | with respect to the                |

## Notations

|                                       |                                                                                      |
|---------------------------------------|--------------------------------------------------------------------------------------|
| $x$                                   | Scalar $x$                                                                           |
| $\mathbf{x}$                          | Vector $\mathbf{x}$                                                                  |
| $\hat{\mathbf{x}}$                    | Estimate of vector $\mathbf{x}$                                                      |
| $\mathbf{x}^T$                        | Transpose of vector $\mathbf{x}$                                                     |
| $[\mathbf{x}]_i$                      | $i$ -th entry of the vector $\mathbf{x}$                                             |
| $\mathbf{X}$                          | Matrix $\mathbf{X}$                                                                  |
| $\mathbf{X}^{-1}$                     | Inverse of matrix $\mathbf{X}$                                                       |
| $\mathbf{X}^\dagger$                  | pseudoinverse of matrix $\mathbf{X}$                                                 |
| $\text{tr}\{\mathbf{X}\}$             | trace of matrix $\mathbf{X}$                                                         |
| $[\mathbf{X}]_{i,j}$                  | $(i, j)$ -th element of the matrix $\mathbf{X}$                                      |
| $[\mathbf{X}]_{k:l,:}$                | rows $k$ up to $l$ of the matrix $\mathbf{X}$                                        |
| $[\mathbf{X}]_{:,k:l}$                | columns $k$ up to $l$ of the matrix $\mathbf{X}$                                     |
| $\mathbf{I}_N$                        | Identity matrix of size $N \times N$                                                 |
| $\mathbf{1}_{M \times N}$             | $M \times N$ matrix with all components one                                          |
| $\mathbf{0}_{M \times N}$             | $M \times N$ matrix with all components zero                                         |
| $\star$                               | Convolution                                                                          |
| $\text{diag}\{\cdot\}$                | Block diagonal matrix                                                                |
| $\lfloor \mathbf{x} \rfloor$          | Largest integer smaller or equal to $\mathbf{x}$                                     |
| $\ \mathbf{x}\ _p$                    | $\ell_p$ -norm of $\mathbf{x}$ , i.e., $(\sum_{i=0}^{N-1}  [\mathbf{x}]_i ^p)^{1/p}$ |
| $\ \mathbf{x}\ $                      | it is the same as $\ \mathbf{x}\ _2$                                                 |
| $p(x)$                                | Probability density function of $x$                                                  |
| $\mathbb{E}[x]$                       | Statistical expectation of $x$                                                       |
| $\nabla \mathbf{f}(\mathbf{x}^{(k)})$ | gradient of vector $\mathbf{f}$ w.r.t. the variable $\mathbf{x}$                     |
| $f_x(x)$                              | Polarity of $x$                                                                      |
| $\frac{\partial f}{\partial x}$       | derivative of $f$ w.r.t. the variable $x$                                            |