Localization using Distance Geometry

Minimal Solvers and Robust Methods for Sensor Network Self-Calibration

Larsson, Martin

2022

Document Version:
Publisher's PDF, also known as Version of record

– CENTRUM SCIENTIARUM MATHEMATICARUM –

Localization using Distance Geometry

Minimal Solvers and Robust Methods for Sensor Network Self-Calibration

MARTIN LARSSON

Lund University
Faculty of Engineering
Centre for Mathematical Sciences
Mathematics

Localization using Distance Geometry

Localization using Distance Geometry

Minimal Solvers and Robust Methods for Sensor Network Self-Calibration

by Martin Larsson

LUND
UNIVERSITY

Thesis for the degree of Doctor of Philosophy in Engineering
Thesis advisors: Dr. Magnus Oskarsson, Prof. Kalle Åström, Lic. Björn Lindquist
Faculty opponent: Prof. Henk Wymeersch

To be presented, with the permission of the Faculty of Engineering of Lund University, for public criticism in the lecture hall MH:Gårding at the Centre for Mathematical Sciences on Friday, the 28th of October 2022 at 13:15.

Organization
LUND UNIVERSITY

Centre for Mathematical Sciences
Box 118
SE–221 00 LUND
Sweden

Author(s)
Martin Larsson

Document name
DOCTORAL DISSERTATION

Date of disputation
2022-10-28

Sponsoring organization
WASP

Title and subtitle
Localization using Distance Geometry: Minimal Solvers and Robust Methods for Sensor Network Self-Calibration

Abstract
In this thesis, we focus on the problem of estimating receiver and sender node positions given some form of distance measurements between them. This kind of localization problem has several applications, e.g., global and indoor positioning, sensor network calibration, molecular conformations, data visualization, graph embedding, and robot kinematics. More concretely, this thesis makes contributions in three different areas.

First, we present a method for simultaneously registering and merging maps. The merging problem occurs when multiple maps of an area have been constructed and need to be combined into a single representation. If there are no absolute references and the maps are in different coordinate systems, they also need to be registered.

In the second part, we construct robust methods for sensor network self-calibration using both Time of Arrival (TOA) and Time Difference of Arrival (TDOA) measurements. One of the difficulties is that corrupt measurements, so-called outliers, are present and should be excluded from the model fitting. To achieve this, we use hypothesis-and-test frameworks together with minimal solvers, resulting in methods that are robust to noise, outliers, and missing data. Several new minimal solvers are introduced to accommodate a range of receiver and sender configurations in 2D and 3D space. These solvers are formulated as polynomial equation systems which are solved using methods from algebraic geometry.

In the third part, we focus specifically on the problems of trilateration and multilateration, and we present a method that approximates the Maximum Likelihood (ML) estimator for different noise distributions. The proposed approach reduces to an eigendecomposition problem for which there are good solvers. This results in a method that is faster and more numerically stable than the state-of-the-art, while still being easy to implement. Furthermore, we present a robust trilateration method that incorporates a motion model. This enables the removal of outliers in the distance measurements at the same time as drift in the motion model is canceled.

Key words
localization, TDOA, TOA, trilateration, multilateration, distance geometry, registration, sensor network self-calibration

Classification system and/or index terms (if any)

Supplementary bibliographical information

Language
English

ISSN and key title
1404-0034

ISBN
978-91-8039-377-5 (print)
978-91-8039-378-2 (pdf)

Recipient's notes

Number of pages
200

Price

Security classification

Localization using Distance Geometry

Minimal Solvers and Robust Methods for Sensor Network Self-Calibration

by Martin Larsson

LUND
UNIVERSITY

A doctoral thesis at a university in Sweden takes either the form of a single, cohesive research study (monograph) or a summary of research papers (compilation thesis), which the doctoral student has written alone or together with one or several other author(s).

In the latter case the thesis consists of two parts. An introductory text puts the research work into context and summarizes the main points of the papers. Then, the research publications themselves are reproduced, together with a description of the individual contributions of the authors. The research papers may either have been already published or are manuscripts at various stages (in press, submitted, or in draft).

Funding information: The thesis work was partially supported by Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation.

Faculty of Engineering, Centre for Mathematical Sciences

ISBN: 978-91-8039-377-5 (print)
ISBN: 978-91-8039-378-2 (pdf)
ISSN: 1404-0034
LUTFMA-1079-2022

Printed in Sweden by Media-Tryck, Lund University, Lund 2022

Media-Tryck is a Nordic Swan Ecolabel certified provider of printed material. Read more about our environmental work at www.mediatryck.lu.se

Printed matter 3041 0903

MADE IN SWEDEN

Contents

List of Publications

This thesis is based on the following publications, referred to by their Roman numerals. All papers are reproduced with permission of their respective publishers. The author's contribution to each paper is listed below.

Main papers:

I **Registration and Merging Maps with Uncertainties**

M. Larsson, K. Åström, M. Oskarsson
2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN), pp. 206–212

Author's contribution: MO and KÅ came up with the original ideas, but all authors took part in discussing them. ML wrote the code and conducted the experiments. Kenneth Batstone aided with the collection of real data. ML, KÅ, and MO wrote the paper jointly.

II **Upgrade Methods for Stratified Sensor Network Self-Calibration**

M. Larsson, G. Flood, M. Oskarsson, K. Åström
ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4851–4855

Author's contribution: All authors contributed to the theory development. ML and MO performed experiments. All authors jointly wrote the code and the paper.

III **Fast and Robust Stratified Self-Calibration Using Time-Difference-Of-Arrival Measurements**

M. Larsson, G. Flood, M. Oskarsson, K. Åström
ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4640–4644

Author's contribution: ML, MO, and KÅ developed the theory. ML created the minimal solvers. ML, GF, and MO performed experiments. ML wrote most of the paper with contributions from all authors.

IV **Extension of Time-Difference-of-Arrival Self Calibration Solutions Using Robust Multilateration**

K. Åström, M. Larsson, G. Flood, M. Oskarsson
2021 29th European Signal Processing Conference (EUSIPCO), pp. 870–874

Author's contribution: KÅ developed much of the theory together with ML and MO. ML and KÅ created the minimal solvers. ML and GF performed experiments. All authors jointly wrote the paper.

V **Sensor node calibration in presence of a dominant reflective plane**

E. Tegler, M. Larsson, M. Oskarsson, K. Åström
Accepted for publication in *2022 30th European Signal Processing Conference (EUSIPCO)*

Author's contribution: KÅ came up with the original idea. All authors contributed to the theory development. ML created the minimal solvers. KÅ performed experiments. All authors jointly wrote the paper.

VI **Optimal Trilateration Is an Eigenvalue Problem**

M. Larsson, V. Larsson, K. Åström, M. Oskarsson
ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5586–5590

Author's contribution: All authors contributed to the theory development. ML created the solvers for offsets and planar position. ET, ML, and MO performed experiments. All authors jointly wrote the paper.

VII **Trilateration Using Motion Models**

M. Larsson, E. Tegler, K. Åström, M. Oskarsson
2022 25th International Conference on Information Fusion (FUSION)

Author's contribution: MO came up with the idea, performed the experiments and wrote most of the paper. All authors took part in discussing the ideas. ML contributed to the writing process and theory development.

VIII **Single Source Localization As an Eigenvalue Problem**

M. Larsson, V. Larsson, K. Åström, M. Oskarsson
Submitted for publication in *IEEE Transactions on Signal Processing*

Author's contribution: This paper extends Paper VI. ML developed the theory, performed experiments, and wrote the manuscript.

Subsidiary papers:

Improved Functional MRI Activation Mapping in White Matter Through Diffusion-Adapted Spatial Filtering

D. Abramian, M. Larsson, A. Eklund, H. Behjat
2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI), pp. 539–543

Spectral Characterization of Functional MRI Data on Voxel-Resolution Cortical Graphs

H. Behjat, M. Larsson
2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI), pp. 558–562

Diffusion-informed spatial smoothing of fMRI data in white matter using spectral graph filters

D. Abramian, M. Larsson, A. Eklund, I. Aganj, C.-F. Westin, H. Behjat
NeuroImage, Volume 237, 2021, Article 118095

Accurate Indoor Positioning Based on Learned Absolute and Relative Models

C. Kjellson, M. Larsson, K. Åström, M. Oskarsson
2021 International Conference on Indoor Positioning and Indoor Navigation (IPIN), pp. 1–8

Acknowledgments

I would like to thank my supervisors Magnus Oskarsson, Kalle Åström, and Björn Lindquist for guiding me through the last five years. I would also like to thank my colleagues at the math department and Combain, in particular, my other coauthors Gabrielle Flood, Viktor Larsson, and Erik Tegler. Finally, I wish to give a special thanks to my friends and family, in particular, Bibiana Prinoth for keeping me motivated during the writing of this thesis.

Funding

This work was partially supported by the Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation.

Abstract

In this thesis, we focus on the problem of estimating receiver and sender node positions given some form of distance measurements between them. This kind of localization problem has several applications, e.g., global and indoor positioning, sensor network calibration, molecular conformations, data visualization, graph embedding, and robot kinematics. More concretely, this thesis makes contributions in three different areas.

First, we present a method for simultaneously registering and merging maps. The merging problem occurs when multiple maps of an area have been constructed and need to be combined into a single representation. If there are no absolute references and the maps are in different coordinate systems, they also need to be registered.

In the second part, we construct robust methods for sensor network self-calibration using both Time of Arrival (TOA) and Time Difference of Arrival (TDOA) measurements. One of the difficulties is that corrupt measurements, so-called outliers, are present and should be excluded from the model fitting. To achieve this, we use hypothesis-and-test frameworks together with minimal solvers, resulting in methods that are robust to noise, outliers, and missing data. Several new minimal solvers are introduced to accommodate a range of receiver and sender configurations in 2D and 3D space. These solvers are formulated as polynomial equation systems which are solved using methods from algebraic geometry.

In the third part, we focus specifically on the problems of trilateration and multilateration, and we present a method that approximates the Maximum Likelihood (ML) estimator for different noise distributions. The proposed approach reduces to an eigendecomposition problem for which there are good solvers. This results in a method that is faster and more numerically stable than the state-of-the-art, while still being easy to implement. Furthermore, we present a robust trilateration method that incorporates a motion model. This enables the removal of outliers in the distance measurements at the same time as drift in the motion model is canceled.

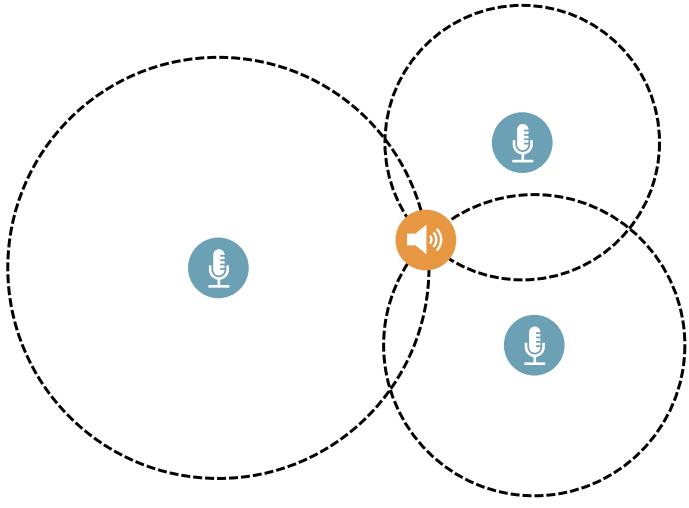
Populärvetenskaplig Sammanfattning

Positionering har blivit mer och mer viktigt i dagens samhälle. Numera går de flesta människor runt med en smartphone i fickan och nyttjar GPS för positionering och navigering. Utomhus är detta ofta en fullt fungerande lösning, men inomhus fungerar det sämre. Där dämpas och studsar GPS-signalerna mot väggar och golv vilket försvårar positioneringen, och i vissa fall går det inte alls. Således krävs det en utveckling av andra metoder för när GPS inte är ett alternativ. En möjlighet är att använda andra radiosignaler (t.ex. Wi-Fi och Bluetooth), men det går också att använda ljud eller ljus. För många positioneringsproblem är geometrin den samma oberoende av vilken teknik som används, så som exempel kan vi för tillfället fokusera på ljudmätningar.

Anta att vi har ett rum med ett antal mikrofoner godtyckligt utplacerade. Anta vidare att vi går runt i det rum med en högtalare, spelandes musik. Kan vi med hjälp av ljudinspelningar från mikrofonerna lista ut var högtalaren befann sig vid en viss tidpunkt? Kan vi lista ut var mikrofonerna var placerade relativt högtalarens positioner, eller kanske relativt golvet och väggarna i rummet? Svaret på dessa frågor är ja, och i den här avhandlingen utforskar vi metoder för att lösa dessa och liknande positioneringsproblem.

Ofta jobbar vi med någon form av avståndsmätningar som har uppkommit med hjälp av ljud- eller radiosignaler. Som exempel, säg att en högljudd trumpetare spelar en signal och vi, som befinner oss på ett säkert avstånd, hör signalen efter en sekund. Då är avståndet mellan oss ungefär 340 m, eftersom ljudets hastighet är ca 340 m/s. Endast en sådan här avståndsmätning är dock inte tillräcklig för att exakt bestämma trumpetarens position. Vi kan enbart säga att den entusiastiske musikern finns någonstans på en cirkel med radien 340 m och med oss i centrum. Fler mätningar gör dock problemet lösbart (se figuren på nästa sida).

Det finns flera varianter av den här typen av problem som är svårare och mer intressanta att lösa. Många varianter uppstår på grund av att vi inte har perfekta avståndsmätningar. I praktiken är inga mätningar exakta utan de innehåller brus. Då kan vi inte få exakta lösningar för positionerna, och vi måste hitta den lösning som passar data bäst. Det kan dock förekomma mätningar som är så dåliga att det är bättre om vi kunde identifiera dem och ta bort dem. En anledning till att så stora fel uppstår är att signaler kan studsa på väggar och tak och skapa ekon. Då går signalen inte raka vägen mellan källan och mottagaren, och avstånden blir således fel. Med mer avancerade metoder är det dock möjligt att modellera hur signalerna studsar, och på så vis behöver vi inte ta bort de här mätningarna. Som en bonus kan vi då samtidigt hitta var väggar och tak befinner sig. De metoder som presenteras i denna avhandlingen är konstruerade för att just kunna hantera brusiga mätningar och stora fel, vilket i slutändan resulterar i bättre positionering.



Från varje mikrofon (blå) till högtalaren (orange) har vi kända avstånd som illustreras med de streckade cirklarna. Högtalaren befinner sig där cirklarna skär varandra. Minst tre mikrofoner behövs för att exakt bestämma högtalarens position.

Part I

Research Context

Chapter 1

Introduction

Localization problems are ubiquitous in the modern world, as it becomes increasingly common for people to use their smartphones and the Global Navigation Satellite System (GNSS) for positioning and navigation. However, GNSS does not typically perform well indoors, so other technologies and methods need to be used in these environments. There is also a growing need for localization indoors with applications such as guiding visitors through museum exhibitions, navigation in large shopping centers, and tracking personnel and equipment [1]. Though, indoor positioning is just one of the applications for the localization methods presented in this thesis. Other applications include sensor network self-calibration, molecular conformations [2, 3], data visualization, graph embedding, and robot kinematics [4].

A more abstract description of the problems we solve in this thesis is the following. Assume we have a set of unknown points in Euclidean space, and we are given pairwise distances between some of them. Can we then find the location of the points? This is the fundamental problem in a branch of mathematics called *distance geometry* [5]. More accurately, it is a whole family of problems with a plethora of variations. For example, if all possible distances are given, the problem is known as classical *multidimensional scaling* [6]. If all but one point are known, it reduces to *trilateration* [7]. The problem can also be made more difficult by introducing unknown offsets in the distances that also must be estimated along with the points.

The methods presented in this thesis are mostly agnostic when it comes to how the distance measurements are acquired. A wide range of technologies can be used, e.g., Received Signal Strength (RSS) measurements from Wi-Fi access points or Bluetooth beacons, Time of Arrival (TOA) measurements using Ultra-Wideband (UWB), or Time

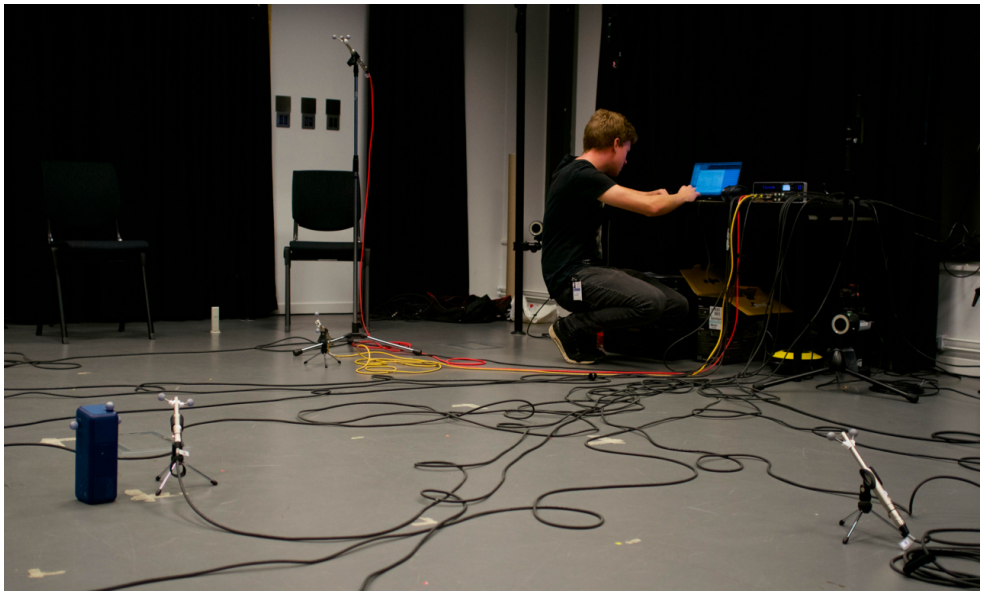


Figure 1.1: Part of the setup used for one of our experiments. A number of microphones (receivers) are placed in a room and a single speaker (sender, blue cuboid to the left) is moved through the setup while playing music or chirp sounds. Using audio recordings from the microphones, the receiver and sender positions can be recovered. For the purpose of collecting ground truth, both the microphones and the speaker are equipped with reflective markers tracked by an optical motion capture system.

Difference of Arrival (TDOA) measurements extracted from audio recordings.

In this thesis, we mainly focus on two different variations of the localization problem described above. The first one involves partitioning the points into two groups which we refer to as *receivers* and *senders*, and distance measurements can only exist between the two groups - not within them. This structure can be represented using a bipartite graph, where in the case of audio measurements, the nodes correspond to microphones and speakers (see Figure 1.1). We refer to the problem of positioning the receivers and senders as *sensor network self-calibration*. The second variation of the problem is *trilateration* and *multilateration*, where a single point is to be located.

The thesis is divided into two parts. In this first part, we provide background and context for the scientific publications which are included in the second part. In the remainder of this first part, Chapter 2 deals with the distance measurements and how they are acquired. Chapter 3 then provides more details regarding the methods and tools used to solve the localization problems. Chapter 4 gives an overview of some problems from distance geometry that are relevant for the publications. Finally, in Chapter 5, we conclude with a summary of the scientific publications and provide possible directions for future work.

Chapter 2

Distance Measurements

To perform localization using the methods proposed in this thesis, we require some form of distance measurements between our receiver and sender positions. There are a large number techniques that can be used for this with various benefits and drawbacks. In the scientific publications, we primarily focus on using sound, but it is important to note that the proposed methods are largely agnostic to which technique is used. As long as some form of distance can be acquired, the proposed methods can be used.

In the following, we will let $\boldsymbol{r}_i \in \mathbb{R}^N$ for $i = 1, \ldots, m$ and $\boldsymbol{s}_j \in \mathbb{R}^N$ for $j = 1, \ldots, n$ denote m receivers and n senders in Euclidean space. Distances between them we notate $d_{ij} = \|\boldsymbol{r}_i - \boldsymbol{s}_j\|$, where $\|\cdot\|$ is the Euclidean norm. For practical applications, we usually have $N = 2$ or $N = 3$, but several of the localization methods discussed generalize to higher dimensions.

2.1 Time of Arrival

Signals, be it radio, light, or sound, have some finite velocity through a medium. This introduces a delay between the Time of Transmission (TOT) at the sender and the Time of Arrival (TOA) at the receiver. If the sender and receiver have synchronized clocks, this delay, the Time of Flight (TOF), can be measured. Assuming the velocity v of the signal is known, the TOF directly translates to a distance measurement. We can model this as

$$(t_{ij} - \tau_j)v = d_{ij} = \|\boldsymbol{r}_i - \boldsymbol{s}_j\|, \tag{2.1}$$

where t_{ij} is the TOA at receiver i and τ_j is the TOT at sender j. Note that t_{ij} is indexed with both i and j as the signal transmitted from a sender arrives at all receivers.

When working with radio, or light the velocity v is simply the speed of light and can be considered constant. When working with audio, the velocity through air is dependent on several factors, such as temperature, pressure, and humidity, although, the temperature dominates. As an example, at 15 °C the speed of sound is approximately $340\,\mathrm{m\,s^{-1}}$, while at 30 °C it is $350\,\mathrm{m\,s^{-1}}$ – an increase of 3 %. Since the velocity gets multiplied by all TOFs, the resulting node positions can be scaled with any relative error in the velocity.

2.2 Round-Trip Time

In the above scenario, the receivers and senders were synchronized, but this is not always the case. When the nodes are unsynchronized, we can instead use Round-Trip Time (RTT). This technique works by the sender first transmitting a signal to the receiver, and after a known delay, the receiver responds. The time between the TOT and TOA on the sender's side then equals two TOF plus the delay on the receiver's side. After deducting this delay it is trivial to calculate the distance.

With the release of the Wi-Fi IEEE 802.11-2016 standard, this technique has become available in a growing number of smartphones and Wi-Fi access points. This has the potential of greatly improving indoor positioning as methods reach submeter accuracy [8, 9].

2.3 Received Signal Strength

Signals naturally get weaker further away from their source. If the transmission strength is known, it is possible to estimate the distance to the source based on the Received Signal Strength (RSS). In the case of Wi-Fi and Bluetooth radio signals, the log-distance path loss model [10, Chapter 8] (also referred to as the one-slope model [11, Chapter 4.7]) is commonly used, and is given by

$$C_{ij} = C_0 + 10\eta \log_{10}(d_{ij}) + \epsilon, \tag{2.2}$$

where the RSS C_{ij}, as measured in dBm, is a function of the distance d_{ij} in meters. C_0 is the signal strength 1 m from the source and η is an attenuation factor determining how quickly the RSS will fade. In indoor environments, RSS measurements from Wi-Fi

access points are commonly used for positioning. Reasonable values for C_0 and η are then -37 dBm and 2.8, respectively (see e.g. [12]). The noise ϵ can be modeled as a zero-mean Gaussian with a standard deviation in the order of 5 dBm. Alternatively, we can view d_{ij} as our measurements with log-normal noise distribution. Additional terms can also be added to (2.2) to account for the dampening of floors or walls [13].

Compared to RTT, RSS measurements are very noisy, and consequently do not offer the same resulting localization accuracy. However, a benefit of RSS is that the sender can be completely passive, as it does not need to respond to incoming packets or be synchronized with the receiver. Furthermore, in the case of indoor environments, Wi-Fi access points and Bluetooth beacons are ubiquitous, enabling localization without the introduction of additional infrastructure.

2.4 Time Difference of Arrival

Time Difference of Arrival (TDOA) is similar to TOA, except that the TOT τ is unknown due to receivers and senders not being synchronized. For example, in the case of audio, we do not know when a particular sound event occurred, only when the microphones detected it. In this scenario, the exact distance cannot be retrieved but rather a so-called pseudo-range or pseudo-distance. Letting $z_{ij} = t_{ij}v$ and $o_j = \tau_j v$, we can write

$$(t_{ij} - \tau_j)v = z_{ij} - o_j = d_{ij} = \|\boldsymbol{r}_i - \boldsymbol{s}_j\|, \tag{2.3}$$

where z_{ij} is the measurement and o_j is an unknown offset. Working with TDOA measurements is inherently more difficult than TOA due to the additional unknowns o_j that need to be solved for. However, it is also more powerful. For example, the lack of synchronization and two-way communication enables the positioning of sporadic sound events, e.g., noisy animals. Also, since the receivers only need to passively listen for incoming signals, introducing a large number of receivers does not clutter the medium.

Chapter 3

Minimal Solvers and Polynomial Equation Systems

Real data is noisy, and this needs to be taken into consideration when fitting a model. It is often the case that the distribution of the noise is known, and it then becomes possible to construct maximum likelihood estimators. However, data can also suffer from spurious measurements, so-called *outliers*, that fall outside of the expected noise distribution. These measurements would be detrimental to the model fitting if included, and we therefore wish to identify and remove them. One way of achieving this is to use minimal solvers in a hypothesis-and-test framework. In this chapter, we introduce minimal solvers and the robust estimation method RANSAC [14]. As the minimal solvers in this thesis take the form of polynomial equation systems, we conclude the chapter with an overview of the mathematical tools used for solving these.

3.1 Minimal Solvers

Minimal problems are instances of model fitting where the minimal amount of data is given, i.e., with any less data, the problem becomes underdetermined and has infinitely many solutions. *Minimal solvers* provide solutions to minimal problems. Example 3.1 shows a minimal solver for fitting a circle to points in the plane. These kinds of solvers are often used in hypothesis-and-test frameworks, e.g., RANSAC, for robust estimation, as they minimize the risk of including outliers in the fitting and hence give optimal time complexity. In this thesis, all minimal problems are formulated as polynomial equation systems, and as we will see, this enables us to use methods from algebraic geometry and

automated tools to construct solvers.

Example 3.1. (Circle fitting in the plane) Let $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n \in \mathbb{R}^2$ be a set of points in the plane, to which we want to fit a circle. The circle is defined by a center point $\boldsymbol{c} \in \mathbb{R}^2$ and a radius $r \in \mathbb{R}$. Every point $\boldsymbol{p}_i$ admits the equation $\|\boldsymbol{p}_i - \boldsymbol{c}\| = r$, and since we have three unknowns $(\boldsymbol{c}, r)$, the problem is minimal when $n = 3$.

Squaring the equations results in the polynomial equation system $\|\boldsymbol{p}_i - \boldsymbol{c}\|^2 = r^2$ for $i = 1, \ldots, 3$. Subtracting the first equation from the rest, we get a linear system in $\boldsymbol{c}$.

$$2 \begin{bmatrix} (\boldsymbol{p}_2 - \boldsymbol{p}_1)^T \\ (\boldsymbol{p}_3 - \boldsymbol{p}_1)^T \end{bmatrix} \boldsymbol{c} = \begin{bmatrix} \boldsymbol{p}_2^T \boldsymbol{p}_2 - \boldsymbol{p}_1^T \boldsymbol{p}_1 \\ \boldsymbol{p}_3^T \boldsymbol{p}_3 - \boldsymbol{p}_1^T \boldsymbol{p}_1 \end{bmatrix} \tag{3.1}$$

Once $\boldsymbol{c}$ is found, the radius is given by $r = \|\boldsymbol{p}_1 - \boldsymbol{c}\|$. □

3.2 RANSAC

Random Sample Consensus (RANSAC) [14] is an iterative estimation method, specifically designed to be robust against outliers, with several variants developed over the years (see e.g. [15] and references therein). In every iteration, it samples a minimal amount of data, fits a model to the sample, and then evaluates how well the remaining data agrees with the found model parameters. Data points that fit well can be classified as inliers, while the remaining points are classified as outliers. Across all iterations, the parameters with the most inliers are chosen. A common final step is then to fit the model to all inliers, e.g., using least squares. The method is summarized in Algorithm 1, and Figure 3.1 shows an example of how RANSAC can be applied to the circle fitting in Example 3.1.

Two parameters must be set for RANSAC: (i) the number of iterations, and (ii) the threshold to use when classifying data as inliers and outliers. With too few iterations, we might not find a good sample without outliers, and the resulting fit will be poor. With too many iterations, the method becomes unnecessarily slow. A poorly chosen threshold will result in the misclassification of inliers and outliers.

3.3 Algebraic Geometry

The minimal solvers in this thesis are formulated as polynomial equation systems. In Section 3.4, we will describe how to solve these, but in this section, we will first lay a

Algorithm 1 RANSAC

$\mathcal{I}_{\max} \leftarrow \emptyset$
while sufficient many iterations **do**
  Create a minimal sample $\mathcal{S}$ of the data.
  Fit model $\mathcal{M}$ to sample $\mathcal{S}$ using minimal solver.
  Evaluate how well remaining data fits $\mathcal{M}$.
  Partition data into inliers $\mathcal{I}$ and outliers $\mathcal{O}$ based on errors.
  if $|\mathcal{I}| > |\mathcal{I}_{\max}|$ **then**
    $\mathcal{I}_{\max} \leftarrow \mathcal{I}$
    $\mathcal{M}_{\max} \leftarrow \mathcal{M}$
  end if
end while

foundation by introducing a few concepts from *algebraic geometry*. For further reading, please see [16,17].

Definition 3.3.1. A *monomial* is a finite product of variables, and a *polynomial* is a finite linear combination of monomials.

Given a field $\mathbb{K}$, let $\mathbb{K}[\boldsymbol{x}]$ denote the set of polynomials in $\boldsymbol{x} = [x_1, \dots, x_N]^T$ with coefficients from $\mathbb{K}$. For our purposes, the field in question will either be the complex numbers $\mathbb{C}$ or the finite field $\mathbb{Z}_p$, where p is a prime number.

Some of the following theory require there to be a way of ordering the monomials in $\mathbb{K}[\boldsymbol{x}]$ – a so-called *monomial ordering*. For the univariate case it is intuitive that $1 < x < x^2 < x^3 \cdots$, but in the multivariate case it is less so, for example, is $xy < y^2$? There are several monomial orderings, such as *lex*, *grlex*, and *grevlex*, that disambiguate this. Given a monomial ordering, we can define the *leading term* of a polynomial f as the term with the largest monomial, and we denote this term as $\mathrm{LT}(f)$.

Definition 3.3.2. An *affine variety* V is the set of solutions to a polynomial equation system, i.e., for $f_1, \dots, f_n \in \mathbb{K}[\boldsymbol{x}]$ we have

$$V(f_1, \dots, f_n) = \{\boldsymbol{a} \in \mathbb{K}^N \mid f_i(\boldsymbol{a}) = 0, i = 1, \dots, n\}. \tag{3.2}$$

Definition 3.3.3. An *ideal* generated by the polynomials $f_1, \dots, f_n \in \mathbb{K}[\boldsymbol{x}]$ is the set

$$\langle f_1, \dots, f_n \rangle = \left\{ \sum_{i=1}^{n} h_i(\boldsymbol{x}) f_i(\boldsymbol{x}) \,\middle|\, h_i(\boldsymbol{x}) \in \mathbb{K}[\boldsymbol{x}] \right\}. \tag{3.3}$$

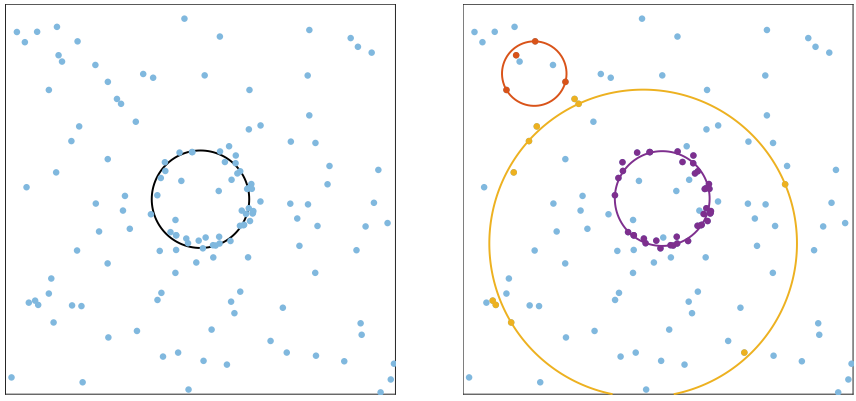


Figure 3.1: Example of RANSAC. (a) A set of points in the plane, some of which are from the circle but with added noise, and some of which are drawn from the uniform distribution (outliers). (b) Three iterations of RANSAC. By sampling three points, a circle can be fitted using the minimal solver in Example 3.1. Points sufficiently close to a circle are considered inliers. The purple circle has the most inliers and is therefore chosen as the best model.

Ideals and varieties correspond to each other in the sense that $V(I)$ is the set of points vanishing on I. When $V(I)$ is a finite set, i.e., the polynomial equation system generating I has a finite number of solutions, both $V(I)$ and I are referred to as *zero-dimensional*. An ideal can be generated by several different sets of polynomials, e.g., $\langle x, y\rangle = \langle x-y, x+y\rangle$ in $\mathbb{C}[x, y]$. For any ideal, a particular set of generating polynomials, called a Gröbner basis, can be calculated.

Definition 3.3.4. Given a monomial ordering on $\mathbb{K}[\boldsymbol{x}]$ and ideal I, the set $g_1, \ldots, g_t \in I$ is said to be a *Gröbner basis* if

$$\langle \mathrm{LT}(g_1), \ldots, \mathrm{LT}(g_t)\rangle = \langle \mathrm{LT}(I)\rangle, \tag{3.4}$$

where $\langle \mathrm{LT}(I)\rangle = \{\mathrm{LT}(f) \mid f \in I\}$.

Unfortunately, this definition does not give much insight in this context. Instead, the important part is that a Gröbner basis has several beneficial properties that enable us to study and solve polynomial equation systems.

Let us introduce the set $[a] = \{b \in \mathbb{K}[\boldsymbol{x}] \mid a - b \in I\}$ for some ideal I. We say that the elements in this set are *congruent modulo* I or in some sense equivalent.

Definition 3.3.5. Given an ideal I, the *quotient ring* $\mathbb{K}[\boldsymbol{x}]/I$ is the set of equivalence classes for congruence modulo I:

$$\mathbb{K}[\boldsymbol{x}]/I = \{[a] | a \in \mathbb{K}[\boldsymbol{x}]\}. \tag{3.5}$$

$\mathbb{K}[\boldsymbol{x}]/I$ is a $\mathbb{K}$-vector space, and provided $\mathbb{K}$ is algebraically closed and $V(I)$ is a finite set, it is finite-dimensional [16, Finiteness Theorem]. Furthermore, the dimension of $\mathbb{K}[\boldsymbol{x}]/I$ serves as an upper bound for the number of solutions in $V(I)$. Another consequence of $\mathbb{K}[\boldsymbol{x}]/I$ being a finite-dimensional vector space is that we can find a finite basis for it. One such basis is the *standard monomials*, defined as all monomials not in $\langle \mathrm{LT}(I) \rangle = \langle \mathrm{LT}(g_1), \ldots, \mathrm{LT}(g_t) \rangle$, where $g_1, \ldots, g_t$ is a Gröbner basis for I.

The last construction we define is saturation.

Definition 3.3.6. If I, J are ideals, then the *saturation* of I with respect to J is the ideal

$$I : J^{\infty} = \{f \in \mathbb{K}[\boldsymbol{x}] \mid \text{ for all } g \in J, \exists N \geq 0 \text{ such that } fg^N \in I\}. \tag{3.6}$$

Saturating an ideal roughly equates to removing solutions from the corresponding variety (see [16, Theorem 4.10]). This can be very useful if an equation system has an infinite set of spurious solutions that are of no interest. Removing these using saturation can then yield a system with a finite number of solutions that we can solve using the methods presented in the next section [18].

3.4 Solving Polynomial Equation Systems

There are several ways of solving polynomial equation systems, but in this section, we will limit ourselves to methods that reduce to eigendecomposition problems. The benefit of this is that eigendecomposition is a well-studied problem with fast and robust numerical eigensolvers available in practically any linear algebra software package. However, the required transformation of the polynomial equation system is in general far from trivial. To start, we will look at how to solve univariate polynomials.

3.4.1 Univariate Polynomials

Univariate polynomials are significantly easier to solve than multivariate ones. Indeed, it is well known that polynomials of degree less than five have closed-formed solutions [19]. For higher degrees, it is common to use the so-called *companion matrix*. Consider the monic polynomial

$$p(x) = x^n + c_{n-1}x^{n-1} + \cdots + c_1 x + c_0 \tag{3.7}$$

and the eigendecomposition

$$x\begin{bmatrix} x^{n-1} \\ x^{n-2} \\ \vdots \\ x \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} -c_{n-1} & \cdots & -c_1 & -c_0 \\ 1 & & & \\ & \ddots & & \\ & & 1 & \\ & & & 1 \end{bmatrix}}_{\triangleq \mathbf{C}} \begin{bmatrix} x^{n-1} \\ x^{n-2} \\ \vdots \\ x \\ 1 \end{bmatrix}. \tag{3.8}$$

The first row in (3.8) is equivalent to $p(x) = 0$, and the following rows are trivial equalities. The matrix $\mathbf{C}$ is called the *companion matrix*, and its eigenvalues are precisely the roots of $p(x)$. This is easy to see as $p(x)$ is the characteristic polynomial of $\mathbf{C}$, i.e., $\det(\lambda - \mathbf{C}) = p(\lambda)$. Another way of looking at it is that, if we evaluate (3.8) at a solution to $p(x) = 0$, we will get a valid eigenpair (eigenvalue and eigenvector) of $\mathbf{C}$. However, the converse is not necessarily true, in the sense that if x_0 is a multiple root, the corresponding eigenspace is multidimensional and there exists eigenvectors not in the form in (3.8). It is nevertheless the case that the number of solutions to $p(x) = 0$ determines the size of $\mathbf{C}$. Several of these observations will carry over to the multivariate case.

3.4.2 Variable Elimination

One way of solving multivariate polynomial equation systems is to reduce them to univariate ones using elimination. The elimination can be performed, e.g., linearly, using algebraic tools like Macaulay2 [20], or resultants [17]. However, elimination is not always an option, as the operation can be intractable even when using algebraic tools. Furthermore, elimination usually results in a higher degree polynomial which may become numerically unstable due to floating point errors.

3.4.3 Action Matrix Method

The multivariate analog of the companion matrix method in Section 3.4.1 is the action matrix method (also called the Gröbner basis method [21]). For a more detailed description of this method please see [17, 22] and references therein.

Consider the polynomial system $f_1, \ldots, f_n \in \mathbb{K}[\boldsymbol{x}]$ with the corresponding ideal $I = \langle f_1, \ldots, f_n \rangle$ and affine variety V. Furthermore, let $\boldsymbol{b} \in \mathbb{K}[\boldsymbol{x}]^K$ be a monomial basis for the quotient ring $\mathbb{K}[\boldsymbol{x}]/I$, and let $\alpha \in \mathbb{K}[\boldsymbol{x}]$ be a monomial. Typically, we choose $\boldsymbol{b}$ as the standard monomials, but other choices can result in faster solvers [23]. In the quotient ring, the operation of multiplying with α is linear and can consequently be expressed

with a matrix $\mathbf{M} \in \mathbb{K}^{K \times K}$, i.e.,

$$[\alpha \boldsymbol{b}] = [\mathbf{M}\boldsymbol{b}]\,. \tag{3.9}$$

$\mathbf{M}$ and α are referred to as the *action matrix* and *action monomial*, respectively. Note that, $[\alpha \boldsymbol{b}] = [\mathbf{M}\boldsymbol{b}] \Leftrightarrow \alpha \boldsymbol{b} - \mathbf{M}\boldsymbol{b} \in I$ which vanishes on V, i.e., for an $\boldsymbol{x} \in V$, we have $\alpha(\boldsymbol{x})\boldsymbol{b}(\boldsymbol{x}) - \mathbf{M}\boldsymbol{b}(\boldsymbol{x}) = 0$. Consequently, every solution in V yields an eigenpair of the action matrix $\mathbf{M}$. However, it is not necessarily true that every eigenpair of the action matrix corresponds to an element in V. Nevertheless, given the action matrix $\mathbf{M}$, it is often possible to perform an eigendecomposition and recover all solutions from the eigenvalues and eigenvectors.

$\mathbf{M}$ can be found using Gröbner basis calculations, but this approach is seldom used in practice due to floating point errors and runtime requirements [22]. A better approach is to use so-called *elimination templates*.

Elimination Templates

First, we need to find $\boldsymbol{b}$ without performing Gröbner basis calculations involving floating point numbers. The solution to this is to calculate $\boldsymbol{b}$ for an integer instance of the equation system, i.e., one where $f_1, \ldots, f_n \in \mathbb{Z}_p[\boldsymbol{x}]$ for some prime p. This calculation can be done exactly without any rounding errors. Typically, $\boldsymbol{b}$ does not depend on the exact value of the coefficients, so the basis found this way is likely correct also when $f_1, \ldots, f_n \in \mathbb{C}[\boldsymbol{x}]$.

To find the action matrix $\mathbf{M}$, we need to express the elements in $[\alpha \boldsymbol{b}]$ as a linear combination of the elements in $[\boldsymbol{b}]$. Since $\alpha \boldsymbol{b} - \mathbf{M}\boldsymbol{b} \in I$, we can write $\alpha \boldsymbol{b} - \mathbf{M}\boldsymbol{b} = \mathbf{H}\boldsymbol{f}$ for some $\mathbf{H} \in \mathbb{K}[\boldsymbol{x}]^{K \times n}$, where $\boldsymbol{f} = [f_1, \ldots, f_n]^T$. If we now construct a new set of polynomials $\boldsymbol{g} \in \mathbb{K}[\boldsymbol{x}]^m$ by multiplying every f_i by all monomials in $H_{1i}, \ldots, H_{ni}$, we can express $\alpha \boldsymbol{b} - \mathbf{M}\boldsymbol{b}$ linearly in $\boldsymbol{g}$. We can write the polynomials $\boldsymbol{g}$ in matrix form as

$$\boldsymbol{g} = \mathbf{C}\mathbf{X} = \begin{bmatrix} \mathbf{C}_e & \mathbf{C}_r & \mathbf{C}_b \end{bmatrix} \begin{bmatrix} \boldsymbol{e} \\ \alpha \boldsymbol{b} \\ \boldsymbol{b} \end{bmatrix}, \tag{3.10}$$

where $\mathbf{C}$ is referred to as the *elimination template* and contains all coefficients, and $\boldsymbol{X}$ contains the *excessive* $\boldsymbol{e}$, *reducible* $\alpha \boldsymbol{b}$, and *basis* $\boldsymbol{b}$ monomials. Some of the reducible monomials may already exist in $\boldsymbol{b}$ and will result in trivial rows in $\mathbf{M}$ similar to the companion matrix. Since we can express $\alpha \boldsymbol{b} - \mathbf{M}\boldsymbol{b}$ linearly in $\boldsymbol{g}$, there is some matrix $\mathbf{A}$ such that

$$\mathbf{A}\boldsymbol{g} = \mathbf{A}\mathbf{C}\boldsymbol{X} = \begin{bmatrix} \mathbf{O} & \mathbf{I} & -\mathbf{M} \end{bmatrix} \boldsymbol{X} = \alpha \boldsymbol{b} - \mathbf{M}\boldsymbol{b}. \tag{3.11}$$

Finding **M** is now a linear problem equivalent to performing Gaussian elimination on **C**. However, one thing that is not quite clear yet is how to find **H**, or more specifically, which monomials the elements of $\boldsymbol{f}$ should be multiplied with. A naive approach is to simply try progressively higher degree monomials until the above procedure works, but more systematic methods are discussed in [24].

In summary, given α, $\boldsymbol{b}$, and **H**, the procedure for solving a multivariate polynomial system is:

1. Calculate the elimination template **C**. This equates to evaluating several polynomials in the coefficients of $f_1, \ldots, f_n$.
2. Solve the linear system involving **C** to find the action matrix **M**.
3. Perform eigendecomposition of **M**, and extract the solutions from the eigenvalues and/or eigenvectors.

The third step assumes that the unknowns $\boldsymbol{x}$ can be extracted from α and $\boldsymbol{b}$, which in practice is almost always the case. For some problems though, it might be necessary to utilize the original equations $\boldsymbol{f}$ as well.

Automatic Solver Generators

While the above procedure for creating a solver can be done manually, it greatly benefits from automation as even for small problems it typically involves very large polynomials. Another benefit of automation is that the algebraic geometry can be abstracted away, resulting in tools that are accessible to a wider audience. A few automatic solver generators have been proposed [21, 24, 25]. We use the one presented in [24] which takes as input a family of polynomial equation systems and outputs a solver as MATLAB or C++ code. The solver in turn takes the family parameters as input and outputs all solutions to the corresponding system.

Practical Considerations

The approach outlined above is in practice quite limited, with the largest problem being the numerical stability of the resulting solvers. In general, as the number of unknowns, the degree, and the number of solutions to a polynomial system increase, the size of the elimination template increases as well, the numerical stability of the solver gets worse and the execution time gets slower. Consequently, when constructing a solver it is beneficial

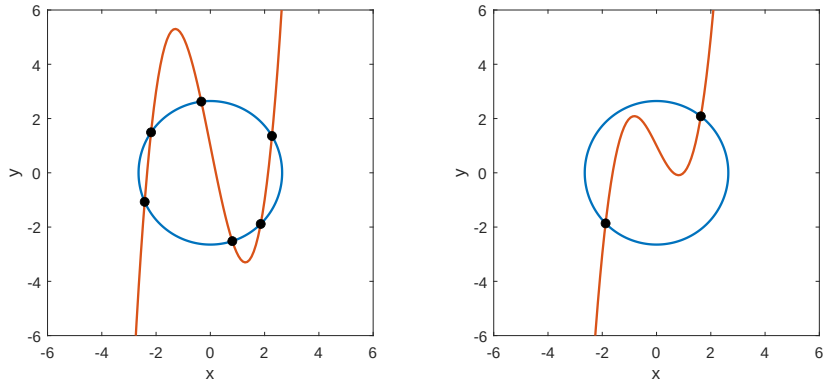


Figure 3.2: Two instances of the equation system in (3.12) for the parameters (a) $(\theta_1, \theta_2) = (7, -5)$ and (b) $(\theta_1, \theta_2) = (7, -2)$. The real solutions to $f_1 = 0$ and $f_2 = 0$ are shown in blue and red, respectively, and the common solutions are indicated by black dots.

to find a parametrization of the problem that has (i) few unknowns, (ii) a low degree, and (iii) few solutions. Finding such a parametrization is the most important part when constructing a solver and is not handled by the automatic generator. Below follows an example of the action matrix method for a toy equation system.

Example 3.2. (Circle and cubic intersection) Consider the polynomials $f_1, f_2 \in \mathbb{C}[x, y]$

$$\begin{cases} f_1 = x^2 + y^2 - \theta_1 \\ f_2 = x^3 + \theta_2 x + 1 - y \end{cases} \tag{3.12}$$

and the corresponding family of equation systems $f_1 = f_2 = 0$, parameterized by $\theta_1, \theta_2 \in \mathbb{C}$. Figure 3.2 shows two instances of this system.

We wish to construct a solver for this system using the action matrix method. First, we consider an integer instance of the system, e.g., let $\theta_1 = \theta_2 = 1$, and using Macaulay2 [20], we can calculate the standard monomials $\boldsymbol{b} = [y^3, y^2, y, xy, x, 1]^T$. Since $\boldsymbol{b}$ is of length six, the problem has at most six complex solutions. We choose the action monomial to be $\alpha = y$. Since $\alpha\boldsymbol{b} = [y^4, y^3, y^2, xy^2, xy, y]^T$ and $[\alpha\boldsymbol{b}] = [\mathbf{M}\boldsymbol{b}]$, we already know that the action matrix $\mathbf{M}$ will be on the form

$$\mathbf{M} = \begin{bmatrix} ? & ? & ? & ? & ? & ? \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ ? & ? & ? & ? & ? & ? \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}. \tag{3.13}$$

It remains to find the missing rows, i.e., to express $[\alpha b_1] = [y^4]$ and $[\alpha b_4] = [xy^2]$ as a linear combination of the basis elements $[\boldsymbol{b}]$. To do this, we construct an extended set of equations

$$\boldsymbol{g} = \begin{bmatrix} f_1 & xf_1 & yf_1 & x^2f_1 & y^2f_1 & f_2 & xf_2 \end{bmatrix}^T = \mathbf{C}\boldsymbol{X}, \tag{3.14}$$

where

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -\theta_1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -\theta_1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & -\theta_1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & -\theta_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & -\theta_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & \theta_2 & 1 \\ 1 & 0 & 0 & 0 & \theta_2 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \end{bmatrix}, \tag{3.15}$$

$$\boldsymbol{X} = \begin{bmatrix} \underbrace{x^4 & x^2y^2 & x^3 & x^2y & x^2}_{\boldsymbol{e}} & y^4 & xy^2 & \underbrace{y^3 & y^2 & y & xy & x & 1}_{\boldsymbol{b}} \end{bmatrix}^T. \tag{3.16}$$

The final solver consists of inserting the values of θ_1 and θ_2 in $\mathbf{C}$, solving the linear system and finding the two missing rows of $\mathbf{M}$, performing eigendecomposition of $\mathbf{M}$, normalizing the eigenvectors such that $b_6 = 1$, and extracting x and y from the eigenvalues and eigenvectors. □

Chapter 4

Localization

Equipped with the algebraic tools from Chapter 3 and the measurements from Chapter 2, we can now focus on the geometrical problems that are used and solved in this thesis. For some of these problems, novel contributions are provided in the scientific publications.

4.1 Trilateration and Multilateration

Trilateration and multilateration, sometimes referred to as *single source localization*, are two different problems occurring when localizing a single receiver using otherwise known data.

4.1.1 Trilateration

Trilateration is the problem of locating a single receiver $\boldsymbol{r} \in \mathbb{R}^N$ given distance measurements $d_j = \|\boldsymbol{r} - \boldsymbol{s}_j\| \in \mathbb{R}$ to known sender positions $\boldsymbol{s}_j \in \mathbb{R}^N$, for $j = 1, \ldots, n$. Every measurement restricts the receiver to a hypersphere centered at the corresponding sender (see Figure 4.1). In general, at least N measurements are required for the problem to have a finite number of solutions. In the minimal case, $n = N$ and there are closed-formed solutions to the problem [7, 26, 27], and in the overdetermined case several methods have been proposed (see Paper VIII and references therein).

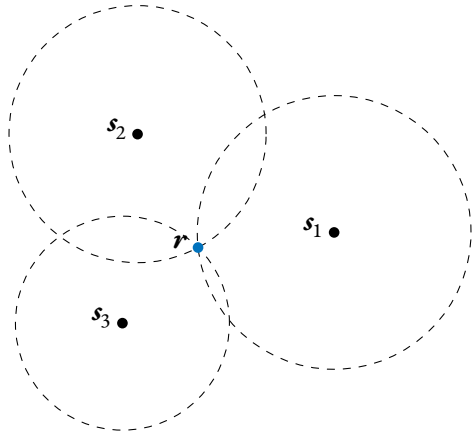


Figure 4.1: Trilateration in the plane. Three senders s_1, s_2, s_3 centered at circles of radii d_1, d_2, d_3, respectively. The sought receiver position r is at the intersection of the circles.

4.1.2 Multilateration

Multilateration is similar to trilateration except for an added offset $o \in \mathbb{R}$, associated with the receiver, that results from the senders and receiver being synchronized (see Section 2.4). The measurements $z_j \in \mathbb{R}$ are modeled as

$$z_j = \|\boldsymbol{r} - \boldsymbol{s}_j\| + o. \tag{4.1}$$

A single measurement does not restrict the receiver position, as was the case for trilateration. However, the difference of two measurements, $z_i - z_j = \|\boldsymbol{r} - \boldsymbol{s}_i\| - \|\boldsymbol{r} - \boldsymbol{s}_j\|$, restricts the receiver to a hyperboloid. Multilateration is then equivalent to finding the intersection of a number of hyperboloids (see Figure 4.2). As in the trilateration case, closed-formed solutions exist in the minimal case $n = N + 1$, as well as other methods for the overdetermined case.

4.2 Classical Multidimensional Scaling

The contributions of this thesis are focused primarily on robust methods that deal with missing data and outliers. With regards to missing data, it is also the case that we have a particular sparsity in our data resulting in the partitioning of the nodes into receivers and senders. Classical Multidimensional Scaling (MDS) on the contrary assumes no missing data and no outliers. In particular, the nodes $\boldsymbol{x}_i \in \mathbb{R}^N$, $i = 1, \ldots, n$ are not partitioned into receivers and senders, and there are distance measurement $d_{ij} = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|$ between every pair of nodes. The matrix $\mathbf{D}^{\circ 2}$ with elements d_{ij}^2 is called the Euclidean Distance

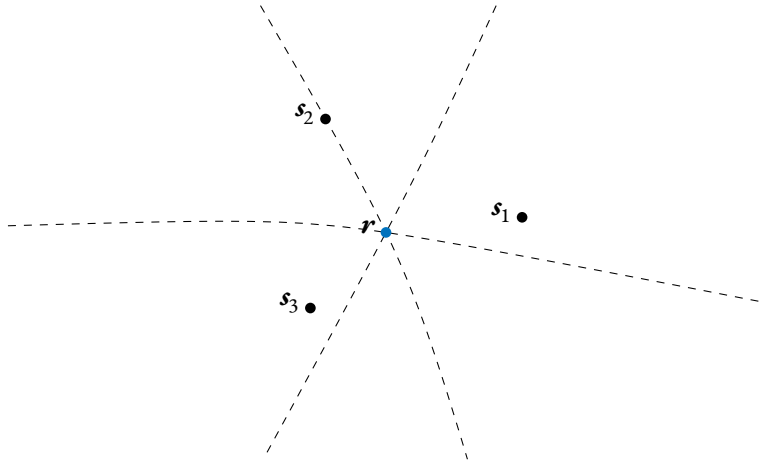


Figure 4.2: Multilateration in the plane. Each pair of senders in s_1, s_2, s_3 yields a hyperbola. The sought receiver position r is at the intersection of the hyperbolae.

Matrix (EDM) [28], and recovering the node positions (up to rigid transformations) from $\mathbf{D}^{\circ 2}$ is precisely MDS. This can be formulated as an eigendecomposition problem [6]. The squared distances are given by

$$d_{ij}^2 = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 = \boldsymbol{x}_i^T\boldsymbol{x}_i - 2\boldsymbol{x}_i^T\boldsymbol{x}_j + \boldsymbol{x}_j^T\boldsymbol{x}_j, \tag{4.2}$$

or expressed using matrices

$$\mathbf{D}^{\circ 2} = \operatorname{diag}(\mathbf{X}^T\mathbf{X})\mathbf{1}^T - 2\mathbf{X}^T\mathbf{X} + \mathbf{1}\operatorname{diag}(\mathbf{X}^T\mathbf{X})^T, \tag{4.3}$$

where $\mathbf{1}$ denotes a vector of ones. We can w.l.o.g. assume $\boldsymbol{x}_1 = \mathbf{0}$, and consequently, $d_{i1}^2 = \boldsymbol{x}_i^T\boldsymbol{x}_i$ and

$$\mathbf{G} \triangleq \mathbf{X}^T\mathbf{X} = -\frac{1}{2}(\mathbf{D}^{\circ 2} - \boldsymbol{d}_1\mathbf{1}^T - \mathbf{1}\boldsymbol{d}_1^T) \tag{4.4}$$

where $\boldsymbol{d}_1$ is the first column of $\mathbf{D}$. Note that $\mathbf{G}$ is defined completely by the distance measurements. Finding the node positions is now a matter of factorizing $\mathbf{G}$ and recovering $\mathbf{X}$. This is done using the eigendecomposition $\mathbf{G} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$,

$$\mathbf{X} = \mathbf{\Lambda}_N^{1/2}\mathbf{Q}_N^T, \tag{4.5}$$

where $\Lambda_N \in \mathbb{R}^{N\times N}$ and $\mathbf{Q}_N \in \mathbb{R}^{n\times N}$ are truncated to the N largest eigenvalues and corresponding eigenvectors. When there is no noise in the data, this reconstruction of $\mathbf{X}$ is exact, while in the presence of noise the cost $\|\mathbf{X}^T\mathbf{X} - \mathbf{G}\|_F^2$ is minimized, where $\mathbf{G}$ is defined using the distance measurements as in (4.4).

4.3 Sensor Network Self-Calibration

Sensor network self-calibration is the problem of localizing a number of sensors, i.e., calibrating their position, possibly without any known anchor positions. For example, we can place several microphones ad hoc in a room and then walk through the setup with a speaker playing some music. Using only the sound recordings from the microphones, we can then find the position of every microphone and simultaneously find the track of the speaker.

For our purposes the sensors, also referred to as nodes, are always partitioned into receivers $\boldsymbol{r}_i \in \mathbb{R}^N$ for $i = 1, \ldots, m$ and senders $\boldsymbol{s}_j \in \mathbb{R}^N$ for $j = 1, \ldots, n$. In the case of TOA, we have the distance measurements $d_{ij} = \|\boldsymbol{r}_i - \boldsymbol{s}_j\|$, which we use to find the node positions. If there is no missing data or outliers, as was the case for Classical Multidimensional Scaling (CMDS), then the problem is referred to as Multidimensional Unfolding (MDU) [29]. One approach for solving this problem is to use the so-called *double compaction matrix*.

4.3.1 Double Compaction Matrix

Consider the squared distances

$$d_{ij}^2 = \boldsymbol{r}_i^T \boldsymbol{r}_i - 2\boldsymbol{r}_i^T \boldsymbol{s}_j + \boldsymbol{s}_j^T \boldsymbol{s}_j. \tag{4.6}$$

Using linear combinations of these we can construct a new equivalent system of equations

$$\begin{bmatrix} d_{11}^2 & d_{12}^2 - d_{11}^2 & \cdots & d_{1n}^2 - d_{11}^2 \\ d_{21}^2 - d_{11}^2 & & & \\ \vdots & & \mathbf{M} & \\ d_{m1}^2 - d_{11}^2 & & & \end{bmatrix}, \tag{4.7}$$

where $\mathbf{M} \in \mathbb{R}^{m-1 \times n-1}$ is the so-called *double compaction matrix* [30, 31] with elements $d_{ij}^2 - d_{i1}^2 - d_{1j}^2 + d_{11}^2 = -2(\boldsymbol{r}_i - \boldsymbol{r}_1)^T(\boldsymbol{s}_j - \boldsymbol{s}_1)$ for $i = 2, \ldots, n$, $j = 2, \ldots, m$. If we collect the receivers and senders, except $\boldsymbol{r}_1$ and $\boldsymbol{s}_1$, as columns in the matrices $\mathbf{R} \in \mathbb{R}^{N \times n-1}$ and $\mathbf{S} \in \mathbb{R}^{N \times m-1}$, respectively, we can write $\mathbf{M} = -2(\mathbf{R} - \boldsymbol{r}_1 \mathbf{1}^T)^T(\mathbf{S} - \boldsymbol{s}_1 \mathbf{1}^T)$. From this, it is clear that the double compaction matrix has rank N, and consequently, we can factor it using Singular Value Decomposition (SVD) into $\mathbf{M} = -2\mathbf{U}^T\mathbf{V}$ where $\mathbf{U} \in \mathbb{R}^{N \times n-1}$ and $\mathbf{V} \in \mathbb{R}^{N \times m-1}$. In Paper II, we generalize this definition resulting in $\mathbf{M} \in \mathbb{R}^{m \times n}$, $\mathbf{R}, \mathbf{U} \in \mathbb{R}^{N \times n}$, and $\mathbf{S}, \mathbf{V} \in \mathbb{R}^{N \times m}$, but the rank constraint remains the same. The factorization of $\mathbf{M}$ is not unique, as for any invertible $\mathbf{L} \in \mathbb{R}^{N \times N}$, $\mathbf{M} = -2\mathbf{U}^T\mathbf{L}^{-1}\mathbf{L}\mathbf{V}$ is also a valid

solution. Nevertheless, we can solve for the receiver and sender positions up to some unknown affine transformations

$$\mathbf{R} = \mathbf{L}^{-T}\mathbf{U} + \boldsymbol{r}_1, \quad \mathbf{S} = \mathbf{L}\mathbf{V} + \boldsymbol{s}_1. \tag{4.8}$$

To find these transformations we will use the remaining equations in (4.7). Since we are only able to solve for the node positions up to a rigid transformation, we can w.l.o.g. assume $\boldsymbol{r}_1 = \mathbf{0}$. Furthermore, let $\boldsymbol{s}_1 = \mathbf{L}\boldsymbol{q}$ for some $\boldsymbol{q} \in \mathbb{R}^N$. We can now write the remaining equations in (4.7) as

$$d_{11}^2 = (\boldsymbol{r}_1 - \boldsymbol{s}_1)^T(\boldsymbol{r}_1 - \boldsymbol{s}_1) = \boldsymbol{s}_1^T\boldsymbol{s}_1 = \boldsymbol{q}^T\mathbf{L}^T\mathbf{L}\boldsymbol{q}, \tag{4.9a}$$

$$d_{1j}^2 - d_{11}^2 = \boldsymbol{s}_j^T\boldsymbol{s}_j - \boldsymbol{s}_1^T\boldsymbol{s}_1 = \boldsymbol{v}_j^T\mathbf{L}^T\mathbf{L}\boldsymbol{v}_j + 2\boldsymbol{v}_j\mathbf{L}^T\mathbf{L}\boldsymbol{q}, \tag{4.9b}$$

$$d_{i1}^2 - d_{11}^2 = \boldsymbol{r}_i^T\boldsymbol{r}_i - 2\boldsymbol{r}_i^T\boldsymbol{s}_1 = \boldsymbol{u}_i^T\mathbf{L}^{-1}\mathbf{L}^{-T}\boldsymbol{u}_i - 2\boldsymbol{u}_i^T\boldsymbol{q}, \tag{4.9c}$$

for $i = 2, \dots, n$, $j = 2, \dots, m$. These can be simplified further by letting $\mathbf{H} = (\mathbf{L}^T\mathbf{L})^{-1}$.

$$d_{11}^2 = \boldsymbol{q}^T\mathbf{H}^{-1}\boldsymbol{q}, \tag{4.10a}$$

$$d_{1j}^2 - d_{11}^2 = \boldsymbol{v}_j^T\mathbf{H}^{-1}\boldsymbol{v}_j + 2\boldsymbol{v}_j\mathbf{H}^{-1}\boldsymbol{q}, \tag{4.10b}$$

$$d_{i1}^2 - d_{11}^2 = \boldsymbol{u}_i^T\mathbf{H}\boldsymbol{u}_i - 2\boldsymbol{u}_i^T\boldsymbol{q}. \tag{4.10c}$$

Since $\mathbf{H}$ is symmetric, we have a total of $N(N+1)/2 + N$ unknowns in $\mathbf{H}$ and $\boldsymbol{q}$. Consequently, in 3D space, $N = 3$ and we have nine unknowns to solve for. If $m \geq 10$, we have nine equations of the type (4.10c), and we can solve for $\mathbf{H}$ and $\boldsymbol{q}$ linearly. However, the minimal cases occur when $(m, n) = (4, 6)$ or $(m, n) = (6, 4)$ and require a bit more work to solve.

If $(m, n) = (6, 4)$, we can use the five linear constraints from (4.10c) to express $\mathbf{H}$ and $\boldsymbol{q}$ in the four unknowns $\alpha_1, \dots, \alpha_4$ as

$$\mathbf{H} = \mathbf{H}_0 + \sum_{k=1}^{4} \alpha_k\mathbf{H}_k, \quad \boldsymbol{q} = \boldsymbol{q}_0 + \sum_{k_1}^{4} \alpha_k\boldsymbol{q}_k, \tag{4.11}$$

where $\mathbf{H}_k$ and $\boldsymbol{q}_k$ for $k = 0, \dots, 4$ are known. Inserting this in (4.10a) and (4.10b) and multiplying with $\det\mathbf{H}$, we get a the following polynomial equation system in $\alpha_1, \dots, \alpha_4$,

$$\det(\mathbf{H})d_{11}^2 = \boldsymbol{q}^T\operatorname{adj}(\mathbf{H})\boldsymbol{q}, \tag{4.12}$$

$$\det(\mathbf{H})(d_{1j}^2 - d_{11}^2) = \boldsymbol{v}_j^T\operatorname{adj}(\mathbf{H})\boldsymbol{v}_j + 2\boldsymbol{v}_j\operatorname{adj}(\mathbf{H})\boldsymbol{q}, \tag{4.13}$$

where $\operatorname{adj}(\mathbf{H}) = \det(\mathbf{H})\mathbf{H}^{-1}$. This system can then be solved using action matrix methods [18,30]. The problem of finding $\mathbf{L}$ and $\boldsymbol{q}$ for various numbers of receivers and senders is studied closer in Paper II.

4.3.2 TDOA and Offset Estimation

When we have TDOA measurements z_{ij} instead of TOA, there are additional offsets $o_j \in \mathbb{R}$ for $j = 1, \dots, n$ to be estimated along with the receiver and sender positions

$$d_{ij} = z_{ij} - o_j = \|\boldsymbol{r}_i - \boldsymbol{s}_j\|. \tag{4.14}$$

One approach for solving this problem is to use a stratified two-tier method [31, 32]. In the first step, the rank constraint on the double compaction matrix $\mathbf{M}$ is used to solve for the offsets. The second step then involves solving a TOA self-calibration problem as explained in the previous section.

Note that $\mathbf{M}$ can be written as $\mathbf{M} = \mathbf{C}_r^T \mathbf{D}^{\circ 2} \mathbf{C}_s$ where $\mathbf{D}^{\circ 2}$ is the EDM with elements $d_{ij}^2 = (z_{ij} - o_j)^2$ and

$$\mathbf{C}_r = \begin{bmatrix} -\mathbf{1}^T \\ \mathbf{I} \end{bmatrix} \in \mathbb{R}^{m-1 \times m}, \quad \mathbf{C}_s = \begin{bmatrix} -\mathbf{1}^T \\ \mathbf{I} \end{bmatrix} \in \mathbb{R}^{n-1 \times n}. \tag{4.15}$$

In contrast to the previous section, where $\mathbf{M}$ was completely defined by the measurements, $\mathbf{M}$ now depends on the offsets. However, it is still the case that $\operatorname{rank} \mathbf{M} = N$ which is equivalent to that every minor of order $N + 1$ vanishes. To solve for the offsets, we thus form the polynomial equation system consisting of all such minors of $\mathbf{M}$ and solve these using action matrix methods. For a sufficient number of receivers and senders, it is also possible to solve for the offsets linearly [31].

4.4 Point Cloud Registration

Localization problems involving only distance measurements, and no absolute references, can only be solved up to a rigid transformation. That is, the found node positions can be translated, rotated, and mirrored without violating any of the distance constraints. If two sets of node positions are to be compared, e.g., comparing estimated positions with ground truth when evaluating some method, the two sets must first be registered. Registering point sets with known correspondences is referred to as *Procrustes analysis* [6] (see also [33–35]). The transformation can be a rigid or similarity transformation, with or without reflection.

Let $\boldsymbol{x}_i, \boldsymbol{y}_i \in \mathbb{R}^N$ for $i = 1, \dots, n$ be two sets of points which we collect as columns in $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{N \times n}$, respectively. We then wish to find a transformation $(s, \mathbf{Q}, \boldsymbol{t}) \in \mathbb{R} \times \mathbb{R}^{N \times N} \times \mathbb{R}^N$, where $\mathbf{Q}^T \mathbf{Q} = \mathbf{I}$, that solves

$$\underset{s, \mathbf{Q}, \boldsymbol{t}}{\text{minimize}} \sum_{i=1}^{n} \|s\mathbf{Q}\boldsymbol{x}_i + \boldsymbol{t} - \boldsymbol{y}_i\|^2 \quad \Leftrightarrow \quad \underset{s, \mathbf{Q}, \boldsymbol{t}}{\text{minimize}} \ \|s\mathbf{Q}\mathbf{X} + \boldsymbol{t}\mathbf{1}^T - \mathbf{Y}\|_F^2. \tag{4.16}$$

Algorithm 2 Procrustes analysis

$\bar{\boldsymbol{x}} \leftarrow \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i$, $\bar{\boldsymbol{y}} \leftarrow \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{y}_i$ ▷ Calculate centroids.

$\bar{\mathbf{X}} \leftarrow \mathbf{X} - \bar{\boldsymbol{x}}\mathbf{1}^T$, $\bar{\mathbf{Y}} \leftarrow \mathbf{Y} - \bar{\boldsymbol{y}}\mathbf{1}^T$ ▷ Subtract centroids from point sets.

$\mathbf{USV}^T \leftarrow \bar{\mathbf{Y}}\bar{\mathbf{X}}^T$ ▷ Singular value decomposition.

$\mathbf{Q} \leftarrow \mathbf{UV}^T$ ▷ Rotation and reflection matrix.

$s \leftarrow \operatorname{tr}\mathbf{S} / \operatorname{tr}\bar{\mathbf{X}}\bar{\mathbf{X}}^T$ ▷ Scaling factor.

$\boldsymbol{t} \leftarrow \bar{\boldsymbol{y}} - s\mathbf{Q}\bar{\boldsymbol{x}}$ ▷ Translation vector.

The method for solving this problem is shown in Algorithm 2 [6, 36, 37]. If a rigid transformation is sought, i.e., without scaling, s is simply set to 1. If reflections are disallowed, we must ensure that $\det \mathbf{Q} = 1$ by replacing the fourth row in Algorithm 2 with $\mathbf{Q} \leftarrow \mathbf{UJV}^T$, where

$$\mathbf{J} = \begin{bmatrix} 1 & & & \\ & \ddots & & \\ & & 1 & \\ & & & \det \mathbf{U} \det \mathbf{V} \end{bmatrix}. \tag{4.17}$$

It is worth noting that the registration performed here assumes that the points are matched, i.e., $\boldsymbol{x}_i$ is associated with $\boldsymbol{y}_i$ for $i = 1, \ldots, n$. If no such matching exists, other methods must be employed, such as the Iterative Closest Point (ICP) algorithm [38].

Chapter 5

Conclusions

The scientific publications in this thesis consist of seven conference papers and one unpublished manuscript. These can be divided into three parts: Paper I which addresses the problem of registering and merging maps, Papers II-V which address the problem of sensor network self-calibration, and Papers VI-VIII which specifically treat trilateration and multilateration. Below follows a summary of the papers and possible directions for future research.

5.1 Paper I: Registration and Merging of Maps

In Paper I, we address the problem of simultaneously registering and merging maps in 2D. The individual problems already have solutions, but the joint problem is shown to be more difficult. Three solvers are proposed and evaluated against other methods using both synthetic and real data. Although the proposed method is shown to perform better than naive methods for certain types of synthetic data, it does not offer a clear benefit for the particular set of real data tested.

5.2 Papers II-V: Sensor Network Self-Calibration

In Papers II-V, we address the problem of sensor network self-calibration using the stratified approach in Section 4.3. The first step is to estimate any TDOA offsets. Minimal solvers for this are presented in Paper III. Once the offsets are found and the double compaction matrix is factorized, the next step is to upgrade the solution in $\mathbf{U}$ and $\mathbf{V}$

to the receiver and senders positions **R** and **S**. Several minimal solvers for performing this upgrade are presented in Paper II. In Paper III, we then put these components together into a full system for self-calibration, which we evaluate on real and synthetic data. Finally, Paper IV presents a post-processing step where robust multilateration is used, optionally with motion priors, to further improve the calibration result.

If a reflective plane, e.g., a floor or ceiling, is present, non-line-of-sight (NLOS) measurements can be incorporated into the calibration system, both to improve the node positioning and to estimate the location of the plane. In Paper V, we deal with this scenario and present methods for offset estimation and self-calibration.

5.2.1 Future Work

The sensor network calibration system presented in Paper III is a good start for a complete system, but more things can be added. First, a wider range of measurement types could be accommodated, e.g., constant offset TDOA [32] and unsynchronized TDOA [39]. Second, the current system only works with receivers and senders in 3D space, but one could also consider the planar case or when receivers are in 3D and senders are in the plane. There is also a need to compare the proposed system with state-of-the-art.

So far in this thesis, we have not mentioned the signal processing required to get TDOA measurements from sound recordings. In the papers, this has been a separate preprocessing step involving GCC-PHAT [40] or some other correlation method. This process can be sensitive to noise and result in erroneous TDOA measurements. A direction of future research could be to investigate if a joint method, where the signal processing and geometry estimation are combined, could be more robust.

5.3 Papers VI-VIII: Trilateration

In Paper VI, we formulate trilateration as an eigendecomposition problem. Specifically, we use a fourth-order approximation of the Maximum Likelihood (ML) cost and manually derive an action matrix for finding all stationary point. The ML estimate is then found by enumerating the stationary points and evaluating the cost. Furthermore, with the adjustment of a set of weights, the proposed method can accommodate various noise distributions, such as Gaussian and log-normal. The proposed method is compared against several other approaches with favorable results concerning execution speed and positioning accuracy.

In Paper VII, we explore how robust trilateration can be combined with a motion model to both tolerate outliers and missing data in the distance measurements and drift in the motion model.

Finally, Paper VIII, an intended journal extension of Paper VI, provides a number of improvements and new insights. In particular, (i) we generalize the cost function to account for dependent noise in the measurements, (ii) we show that it is not necessary to explicitly calculate the eigenvectors in the eigendecomposition and that the largest real eigenvalue corresponds to the global minimum, (iii) we give special treatment to the degenerate cases (e.g. when senders are collinear), and (iv) we describe our implementation and compare it against a range of state-of-the-art methods.

5.3.1 Future Work

Multilateration is briefly treated in Paper VI but did not receive the same attention trilateration did in Paper VIII. In particular, future work could include investigating whether or not there is a similar eigendecomposition for the multilateration case, and if we can say anything about which eigenvalue (e.g. largest real one) corresponds to the global minimizer.

Bibliography

[1] R. Mautz, "Indoor positioning technologies," p. 1 Band, 2012, artwork Size: 1 Band Medium: application/pdf Publisher: ETH Zurich. [Online]. Available: http://hdl.handle.net/20.500.11850/54888

[2] M. W. Trosset, "Applications of multidimensional scaling to molecular conformation," 1997.

[3] J. D. Dunitz, "Distance geometry and molecular conformation, by G. M. Crippen and T. F. Havel, Research Studies Press, Taunton, England, John Wiley and Sons, New York, 1988. pp. 541 + x pp. Price: $142.00," *Journal of Computational Chemistry*, vol. 11, no. 2, pp. 265–266, Mar. 1990. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/jcc.540110212

[4] B. Dasgupta and T. S. Mruthyunjaya, "The Stewart platform manipulator: a review," *Mechanism and Machine Theory*, vol. 35, no. 1, pp. 15–40, Jan. 2000. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0094114X99000063

[5] L. Liberti and C. Lavor, *Euclidean Distance Geometry*, ser. Springer Undergraduate Texts in Mathematics and Technology. Cham: Springer International Publishing, 2017. [Online]. Available: http://link.springer.com/10.1007/978-3-319-60792-4

[6] I. Borg and P. J. F. Groenen, *Modern multidimensional scaling: theory and applications*, 2nd ed., ser. Springer series in statistics. New York: Springer, 2005, oCLC: ocm61260823.

[7] F. Thomas and L. Ros, "Revisiting trilateration for robot localization," *IEEE Transactions on Robotics*, vol. 21, no. 1, pp. 93–101, Feb. 2005, conference Name: IEEE Transactions on Robotics.

[8] C. Gentner, M. Ulmschneider, I. Kuehner, and A. Dammann, "WiFi-RTT Indoor Positioning," in *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, Apr. 2020, pp. 1029–1035, iSSN: 2153-3598.

[9] O. Hashem, M. Youssef, and K. A. Harras, "WiNar: RTT-based Sub-meter Indoor Localization using Commercial Devices," in *2020 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, Mar. 2020, pp. 1–10, iSSN: 2474-249X.

[10] I. Sharp and K. Yu, *Wireless Positioning: Principles and Practice*, ser. Navigation: Science and Technology. Singapore: Springer, 2019. [Online]. Available: http://link.springer.com/10.1007/978-981-10-8791-2

[11] E. Commission, D.-G. for the Information Society, and Media, *COST Action 231 : Digital mobile radio towards future generation systems: Final Report*. Publications Office, 1999.

[12] "Propagation data and prediction methods for the planning of indoor radiocommunication systems and radio local area networks in the frequency range 300 MHz to 100 GHz," International Telecommunication Union (ITU), Tech. Rep. P.1238-8, 2015.

[13] F. Zargoun, I. Henawy, and N. Ziedan, "Effects of Walls and Floors in Indoor Localization Using Tracking Algorithm," *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 3, 2016. [Online]. Available: http://thesai.org/Publications/ViewPaper?Volume=7&Issue=3&Code=ijacsa&SerialNo=5

[14] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, p. 381–395, jun 1981. [Online]. Available: https://doi.org/10.1145/358669.358692

[15] R. Raguram, O. Chum, M. Pollefeys, J. Matas, and J.-M. Frahm, "Usac: A universal framework for random sample consensus," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 2022–2038, 2013.

[16] D. A. Cox, J. Little, and D. O'Shea, *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*, ser. Undergraduate Texts in Mathematics. Cham: Springer International Publishing, 2015. [Online]. Available: http://link.springer.com/10.1007/978-3-319-16721-3

[17] D. A. Cox, J. B. Little, and D. O'Shea, *Using algebraic geometry*, 2nd ed., ser. Graduate texts in mathematics. New York: Springer, 2005, no. 185.

[18] V. Larsson, K. Astrom, and M. Oskarsson, "Polynomial Solvers for Saturated Ideals," in *2017 IEEE International Conference on Computer Vision (ICCV)*. Venice: IEEE, Oct. 2017, pp. 2307–2316. [Online]. Available: http://ieeexplore.ieee.org/document/8237513/

[19] M. Abramowitz, *Handbook of Mathematical Functions, With Formulas, Graphs, and Mathematical Tables,*. New York, NY, USA: Dover Publications, Inc., 1974, ch. 3.8. Algebraic Equations, pp. 17–18.

[20] D. R. Grayson and M. E. Stillman, "Macaulay2, a software system for research in algebraic geometry," Available at http://www.math.uiuc.edu/Macaulay2/.

[21] Z. Kukelova, M. Bujnak, and T. Pajdla, "Automatic Generator of Minimal Problem Solvers," in *Computer Vision – ECCV 2008*, D. Forsyth, P. Torr, and A. Zisserman, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, vol. 5304, pp. 302–315, series Title: Lecture Notes in Computer Science. [Online]. Available: http://link.springer.com/10.1007/978-3-540-88690-7_23

[22] V. Larsson, "Computational Methods for Computer Vision : Minimal Solvers and Convex Relaxations," thesis/docmono, Lund University, 2018, iSBN: 9789177536956 9789177536963. [Online]. Available: http://lup.lub.lu.se/record/cc1ae2a2-409d-414c-87e0-ec381d22649d

[23] V. Larsson, M. Oskarsson, K. Astrom, A. Wallis, T. Pajdla, and Z. Kukelova, "Beyond Grobner Bases: Basis Selection for Minimal Solvers," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Salt Lake City, UT, USA: IEEE, Jun. 2018, pp. 3945–3954. [Online]. Available: https://ieeexplore.ieee.org/document/8578513/

[24] V. Larsson, K. Astrom, and M. Oskarsson, "Efficient Solvers for Minimal Problems by Syzygy-Based Reduction," 2017, pp. 820–829. [Online]. Available: https://openaccess.thecvf.com/content_cvpr_2017/html/Larsson_Efficient_Solvers_for_CVPR_2017_paper.html

[25] B. Li and V. Larsson, "Gaps: Generator for automatic polynomial solvers," *arXiv preprint arXiv:2004.11765*, 2020.

[26] D. E. Manolakis, "Efficient solution and performance analysis of 3-D position estimation by trilateration," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, no. 4, pp. 1239–1248, Oct. 1996, conference Name: IEEE Transactions on Aerospace and Electronic Systems.

[27] I. D. Coope, "Reliable computation of the points of intersection of n spheres in R^n," *ANZIAM Journal*, vol. 42, pp. C461–C477, Dec. 2000. [Online]. Available: https://journal.austms.org.au/ojs/index.php/ANZIAMJ/article/view/608

[28] I. Dokmanic, R. Parhizkar, J. Ranieri, and M. Vetterli, "Euclidean Distance Matrices: Essential theory, algorithms, and applications," *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 12–30, Nov. 2015, conference Name: IEEE Signal Processing Magazine.

[29] P. H. Schönemann, "On metric multidimensional unfolding," *Psychometrika*, vol. 35, no. 3, pp. 349–366, Sep. 1970. [Online]. Available: https://doi.org/10.1007/BF02310794

[30] Y. Kuang, S. Burgess, A. Torstensson, and K. Åström, "A complete characterization and solution to the microphone position self-calibration problem," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 3875–3879, iSSN: 2379-190X.

[31] Y. Kuang and K. Åström, "Stratified sensor network self-calibration from TDOA measurements," in *21st European Signal Processing Conference (EUSIPCO 2013)*, Sep. 2013, pp. 1–5, iSSN: 2076-1465.

[32] K. Batstone, G. Flood, T. Beleyur, V. Larsson, H. R. Goerlitz, M. Oskarsson, and K. Åström, "Robust Self-calibration of Constant Offset Time-difference-of-arrival," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2019, pp. 4410–4414, iSSN: 2379-190X.

[33] G. Wahba, "A Least Squares Estimate of Satellite Attitude," *SIAM Review*, vol. 7, no. 3, pp. 409–409, Jul. 1965. [Online]. Available: http://epubs.siam.org/doi/10.1137/1007077

[34] W. Kabsch, "A solution for the best rotation to relate two sets of vectors," *Acta Crystallographica Section A*, vol. 32, no. 5, pp. 922–923, 1976, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1107/S0567739476001873. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1107/S0567739476001873

[35] B. K. P. Horn, H. M. Hilden, and S. Negahdaripour, "Closed-form solution of absolute orientation using orthonormal matrices," *Journal of the Optical Society of America A*, vol. 5, no. 7, p. 1127, Jul. 1988. [Online]. Available: https://opg.optica.org/abstract.cfm?URI=josaa-5-7-1127

[36] J. C. Gower and G. B. Dijksterhuis, "29Orthogonal Procrustes problems," in *Procrustes Problems*. Oxford University Press, Jan.

2004, _eprint: https://academic.oup.com/book/0/chapter/146378375/chapter-ag-pdf/45002613/book_4410_section_146378375.ag.pdf. [Online]. Available: https://doi.org/10.1093/acprof:oso/9780198510581.003.0004

[37] T. Zinßer, J. Schmidt, and H. Niemann, "Point set registration with integrated scale estimation," in *International conference on pattern recognition and image processing*, 2005, pp. 116–119.

[38] F. Pomerleau, F. Colas, R. Siegwart *et al.*, "A review of point cloud registration algorithms for mobile robotics," *Foundations and Trends® in Robotics*, vol. 4, no. 1, pp. 1–104, 2015.

[39] D. E. Badawy, V. Larsson, M. Pollefeys, and I. Dokmanić, "Localizing unsynchronized sensors with unknown sources," 2021. [Online]. Available: https://arxiv.org/abs/2102.03565

[40] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 4, pp. 320–327, 1976.

Part II

Scientific Publications

Paper I

Registration and merging maps with uncertainties

Martin Larsson[1], Kalle Åström[1], Magnus Oskarsson[1]

[1]*Centre for Mathematical Sciences, Lund University, Lund, Sweden*

Abstract: In this paper we address the problem of registering and merging two maps in two dimensions, given covariance estimates of the two maps. We show that if two maps are given in the same coordinate system, then the problem of merging them in a statistically optimal way can be formulated as a linear least squares problem, but if they are given in different coordinate systems as well the problem becomes highly non-linear and non-convex. We show how we can relax the problem slightly in order to optimize over the registration (i.e. putting the two maps in the same coordinate system) and at the same time optimize over the merged map. The approach is based on finding all stationary points of the optimization problem and evaluating these to choose the global optimum. We show on synthetic data that in many cases the proposed approach gives better results than naively registering and merging the maps. We also show results on real data, where we merge maps given by time-of-arrival measurements, and in these cases simpler linear methods perform just a good as the proposed method.

Keywords: rigid registration; merging; time-of-arrival; SLAM; covariance; mapping

This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation, by the strategic research projects ELLIIT and eSSENCE and by the Swedish Foundation for Strategic Research project "Semantic Mapping and Visual Navigation for Smart Robots" (grant no. RIT15-0038).

Reprinted from *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, M. Larsson, K. Åström, and M. Oskarsson, Registration and Merging Maps with Uncertainties, 206–212, Copyright 2018, with permission from IEEE.

1 Introduction

In this paper we will study the problem of aligning and merging two maps. In our setting, the map is a set of points in 2D. We will also assume that with each map comes an estimate of its uncertainty, represented by a covariance matrix. The goal is now to find a new map that combines the information in the two maps in an optimal way, given the uncertainties and positions of the maps. This is general problem which is frequently occurring in many types of mapping applications. The maps could represent landmark features used in localization systems based on different sensor modalities. Examples are feature points extracted from images, GPS satellite positions, WiFi-locations, Bluetooth beacons, laser landmarks or microphone positions. Figure 1 shows a few examples of sensor technologies used for map estimation. Our own interest and motivation for solving it comes from time-of-arrival estimation and calibration problems. A typical scenario is when several maps have been created of the same surroundings, at different times. In order to get a more accurate map we would like to use the information from two different maps in a statistically correct way. One of the main difficulties and also the reason behind the non-linearity in merging maps in this way is that we need to both register the point sets (since they in general are estimated in different coordinate systems) *and* merge them.

Registration of point sets has a long standing tradition. It is well known that, finding the optimal rigid transformation between two given point sets has a closed form solution, using singular value decomposition (SVD). This has been shown at least two times independently, by Horn in [5] and by Kabsch in [6]. The method works for similarity transformation as well and in [7] it was shown how to also handle correpondences with planes and lines. If the correpondences between the point sets are unknown the problem becomes much harder, and there exist numerous variants of solutions to this, most based on variants of the iterative closest point (ICP) algorithm [8]. In this paper we assume the correspondences are known as would be the case when working with e.g. WiFi access points, ultra-wideband (UWB) transmitters or other landmarks that have unique identifiers.

The rest of the paper is structured as follows. In Section 2 we provided some background to the problem and explain the maps we are considering. In Section 3 we present our proposed solution to the problem and in Section 4 we compare it to other methods using synthetic and real data. Finally, in Section 5 we summarize and provided our conclusions.

Figure 1: Examples of sensor technologies for map estimation. Top-left: Structure from motion using multiple images, [1]. Top-right: Mobile phones that measure distances to UWB tags, [2]. Bottom-left: Microphones measure difference of distances to ambient sound, [3]. Bottom-right: Massive MIMO units measure distances to radio transmitters, [4].

2 Background

The idea of map estimation using non-linear least-squares optimization is quite general and works for different sensor modalities. However, for simplicity we will describe this in terms of distance measurements from sensor positions $s_j \in \mathbb{R}^2$ to map points $x_i \in \mathbb{R}^2$ in the plane. Each noise-free measurement $d_{ij} = ||x_i - s_j||$ is a non-linear function of the map features x_i and sensor features s_j. The actual measurements $\tilde{d_{ij}}$ are typically noisy and can be modelled as

$$\tilde{d_{ij}} = ||x_i - s_j|| + e_{ij}, \tag{1}$$

where e_{ij} are stochastic variables. The complete sets of sensor positions and map points are represented by the vectors $s \in \mathbb{R}^{2k}$ and $m \in \mathbb{R}^{2n}$ respectively, where $s = (s_1^T, \ldots, s_k^T)^T$ and $m = (x_1^T, \ldots, x_n^T)$. The vector $z = \begin{pmatrix} m \\ s \end{pmatrix}$ is used for all unknown parameters. Assuming that the errors are independent and Gaussian, the maximum likelihood estimate is found by minimizing $r(z)^T r(z)$, where r is the vector of all residuals

$$r = (||x_1 - s_1|| - \tilde{d}_{11}, \ldots, ||x_n - s_k|| - \tilde{d}_{nk})^T. \tag{2}$$

Estimation of the parameters often involve obtaining initial estimates of z_0, e.g., using minimal solvers and then local optimization based on the calculated residual $r(z_0)$ and the derivatives $J(z_0)$ of the residual with respect to all unknowns at the current estimate. The next estimate is found by

$$z_{new} = z_0 - (J^T J)^{-1}(J^T r). \tag{3}$$

At the optimum, the Jacobean J provides an estimate of the covariance of the estimated map m and sensor points s according to

$$C_z = \sigma^2 (J^T J), \tag{4}$$

where σ is the standard deviation of the stochastic variables e_{ij}. From the Jacobian we can also obtain the estimate of the covariance C of the map m alone. This is shown in Section 3.4.

This gives us the setting that we will investigate in this paper, namely the maps that we consider are point sets in two dimensions, i.e., n points $x_i \in \mathbb{R}^2, i = 1, \ldots, n$. The whole map is represented by the vector $m \in \mathbb{R}^{2n}$ and its uncertainty is represented by the covariance matrix $C \in \mathbb{R}^{2n \times 2n}$. We will in the following sections show how we can formulate and solve the simultaneous merging and registration of two such maps.

3 Merging maps

3.1 Merging without registration

We assume that two estimates m_1 and m_2 (both in $\mathbb{R}^{2n}$) of the same map are given. We also assume that the local error structures of these maps are known. This could typically be given by the linearizations of the estimates from previous non-linear refinement of the two solutions. The covariance matrices of m_1 and m_2 are given by C_1 and C_2 respectively. We will also define the weighted norm of a vector by

$$\|x\|_A^2 = x^T A^{-1} x. \tag{5}$$

We would now like to find the best estimate of the true map m. In this case finding the maximum likelihood estimate of m is given by the solution to the following problem.

Problem 1 (Weighted merging of two maps) *Given two maps m_1 and m_2, with covariances C_1 and C_2, find the merged map m such that*

$$\min_m \|m - m_1\|_{C_1}^2 + \|m - m_2\|_{C_2}^2. \tag{6}$$

The solution to (6) is given by the least squares solution to

$$(C_1^{-1} + C_2^{-1})m = C_1^{-1} m_1 + C_2^{-1} m_2. \tag{7}$$

If we instead represent our uncertainties with L_1 and L_2 representing the linearization around the two estimates, we have the following problem.

Problem 2 (Weighted merging of two maps) *Given two maps m_1 and m_2, and L_1 and L_2, find the merged map m such that*

$$\min_m \|L_1(m - m_1)\|^2 + \|L_2(m - m_2)\|^2. \tag{8}$$

The optimal solution to (8) is again given by the least squares solution, i.e., the solution to

$$\begin{bmatrix} L_1 \\ L_2 \end{bmatrix} m = \begin{bmatrix} L_1 m_1 \\ L_2 m_2 \end{bmatrix}. \tag{9}$$

If $C_i^{-1} = L_i^T L_i$, then (8) is an equivalent formulation to (6).

3.2 Merging and registration

In the previous section we assumed that m_1 and m_2 were already registered and put in a common coordinate system. In many applications this is not the case. We will now address the much more difficult problem when the two maps are unregistered. In this case we set out to find a rigid transformation mapping m_1 onto m_2 simultaneously as we merge the maps, i.e., we want to solve

Problem 3 (Weighted merging and registration of two maps) *Given two maps m_1 and m_2, and L_1 and L_2, find $\hat{R}$, $\hat{t}$ and the merged map m such that*

$$\min_{m,\hat{R},\hat{t}} \|L_1(m - m_1)\|^2 + \|L_2(\hat{R}m + \hat{t} - m_2))\|^2, \tag{10}$$

where $\hat{R}$ and $\hat{t}$ represent the rotation and translation of the transform, respectively, defined by

$$\hat{R} = \begin{bmatrix} R & & & \\ & R & & \\ & & \ddots & \\ & & & R \end{bmatrix}, \quad \hat{t} = \begin{bmatrix} t \\ t \\ \vdots \\ t \end{bmatrix}, \tag{11}$$

where

$$R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \text{ and } t = \begin{bmatrix} t_x \\ t_y \end{bmatrix}. \tag{12}$$

Using the substitutions $a = \cos\theta$ and $b = \sin\theta$ (10) becomes polynomial in the unknowns a, b, t and m. Adding a Lagrangian term for the constraint $a^2 + b^2 = 1$ and differentiating results in a polynomial system whose solutions are the stationary points of (10). Given the stationary points the minimum can be found by evaluation, however, finding them in the general case is not easy.

The problem is non-linear and does not scale well with the number of points n. By generating random interger instances of the problem we can use Macaulay2 [9] to inspect the corresponding solution set (affine variety). We find that there are a finite number of solution but the number increases with n. Table 1 shows the number of stationary points found by Macaulay2 for various n. As can be seen the relationship is linear and assuming this holds the problem will be unfeasible to solve for large n, since in general actually solving the polynomial system will increase in difficulty with increasing number of solutions.

Table 1: The Number of Solutions When Solving (10) for Different Number of Points n

n	# solutions
2	6
3	14
4	22
5	30
6	38
7	46
8	54

3.3 Approximate merging and registration

In order to address the difficulties from the previous section we will now relax our problem formulation slightly. We assume that the two map estimates m_1 and m_2 are measurements of two true maps m and m' with

$$m_1 = m + \epsilon_1, \tag{13}$$
$$m_2 = m' + \epsilon_2, \tag{14}$$

where ϵ_i is drawn from a zero-mean multivariate Gaussian distribution with covariance C_i. We assume that the true maps are represented in two different coordinate systems related by a translation and a rotation so that

$$m' = \hat{R}m + \hat{t} \Leftrightarrow m = \hat{R}^T(m' - \hat{t}). \tag{15}$$

Using (15) to substitute m and taking the difference between (13) and (14) gives

$$\hat{R}m_1 + \hat{t} - m_2 = \hat{R}\epsilon_1 - \epsilon_2 \equiv \epsilon_3. \tag{16}$$

Here, ϵ_3 will follow a zero-mean Gaussian distribution with covariance $C_3 = \hat{R}C_1\hat{R}^T + C_2$. If we are given two maps m_1 and m_2 and want to find the most likely $\hat{R}$ and $\hat{t}$ we should maximize the probability of getting the error ϵ_3, i.e.,

$$\max_{\hat{R},\hat{t}} P(\epsilon_3) \Leftrightarrow \max_{\hat{R},\hat{t}} e^{-(\hat{R}m_1+\hat{t}-m_2)^T C_3^{-1}(\hat{R}m_1+\hat{t}-m_2)}. \tag{17}$$

Finding the maximum of this is equivalent to minimizing the log-likelihood, i.e.,

$$\min_{\hat{R},\hat{t}} (\hat{R}m_1 + \hat{t} - m_2)^T C_3^{-1}(\hat{R}m_1 + \hat{t} - m_2). \tag{18}$$

The problem described in (18) is highly non-linear and non-convex. Indeed, the problem suffers from the same issues as (10) with the same number of solutions as presented in Table 1. Note that C_3 also depends on the rotation $\hat{R}$.

A simpler problem is to replace C_3^{-1} with a matrix P that is independent of the rotation $\hat{R}$. Then (18) becomes

$$\min_{\hat{R},\hat{t}} (\hat{R}m_1 + \hat{t} - m_2)^T P(\hat{R}m_1 + \hat{t} - m_2). \tag{19}$$

Solving (19) for some initial P, we can then update P as $P = (\hat{R}C_1\hat{R}^T + C_2)^{-1}$ and solve (19) again. This approach can then be iterated until convergence. Once the registration is found the merged points m can be found by solving the system

$$\begin{bmatrix} L_1 \\ L_2\hat{R} \end{bmatrix} m = \begin{bmatrix} L_1 m_1 \\ L_2(m_2 - \hat{t}) \end{bmatrix}. \tag{20}$$

To solve (19) we rewrite the problem as a polynomial system that can be solved using the *action matrix method* [10]. We present three such systems and hence three solvers are constructed, that are increasingly faster and more stable. We start by rewriting the expression in (19) as

$$f = (Ar + Bt - m_2)^T P(Ar + Bt - m_2), \tag{21}$$

where

$$A = \begin{bmatrix} m_1^{(1)} & -m_1^{(2)} \\ m_1^{(2)} & m_1^{(1)} \\ \vdots & \vdots \\ m_1^{(n-1)} & -m_1^{(n)} \\ m_1^{(n)} & -m_1^{(n-1)} \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ \vdots & \vdots \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{22}$$

$$r = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \tag{23}$$

and $m_1^{(k)}$ denotes the kth element of the vector m_1. Differentiating with respect to t and θ results in

$$\nabla_t f = B^T P(Ar + Bt - m_2), \tag{24}$$

$$\nabla_\theta f = r^T D^T A^T P(Ar + Bt - m_2), \tag{25}$$

where

$$D = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}. \tag{26}$$

Using the substitution $a = \cos\theta$ and $b = \sin\theta$ and adding the constraint $a^2 + b^2 = 1$ we get the first polynomial system

$$\begin{cases} \nabla_t f = 0 \\ \nabla_\theta f = 0 \\ a^2 + b^2 = 1, \end{cases} \tag{27}$$

which has at most four solutions when finitely many. Note that by directly differentiating with respect to θ instead of with respect to a and b we get lower degree of our polynomials. This is because we in this case do not have to add the normalization constraint on a and b in the Lagrangian, and the derivative of r with respect to θ is linear in r, i.e., $r'(\theta) = Dr(\theta)$.

The system can be reduced by eliminating the translation. Let $E = (B^T PB)^{-1}B^T P$ and $F = D^T A^T P$. Since $\nabla_t f = 0$ we can solve for the translation and get $t = E(m_2 - Ar)$. Insertion of this into $\nabla_\theta f$ results in

$$\begin{aligned} \nabla_\theta f &= r^T F(Ar + Bt - m_2) \\ &= r^T F(A - BEA)r + r^T F(BEm_2 - m_2) \\ &= r^T \begin{bmatrix} c_1 & c_2 \\ c_3 & -c_1 \end{bmatrix} r + r^T \begin{bmatrix} c_4 \\ c_5 \end{bmatrix} = p(a,b) \end{aligned} \tag{28}$$

for some real numbers $c_1, \ldots, c_5$. Similar to before, $p(a,b)$ together with the constraint $a^2 + b^2 = 1$ form our second polynomial system, now with two equations in two unknowns:

$$\begin{cases} p(a,b) = 0 \\ a^2 + b^2 = 1. \end{cases} \tag{29}$$

The system can be reduced even further into a single polynomial in only b. By separating the even and odd powers of a in $p(a,b)$ and utilizing that $a^2 = b^2 - 1$, we define $p(a,b) = p_1(b) + ap_2(b)$, where

$$p_1(b) = -2c_1 b^2 + c_5 b + c_1, \tag{30}$$

$$p_2(b) = (c_2 + c_3)b + c_4. \tag{31}$$

Additionally, by multiplying with a we get $ap(a,b) = ap_1(b) + (1 - b^2)p_2(b)$. Since $p(a,b) = ap(a,b) = 0$ we construct the system

$$\begin{bmatrix} p_1(b) & p_2(b) \\ (1 - b^2)p_2(b) & p_1(b) \end{bmatrix} \begin{bmatrix} 1 \\ a \end{bmatrix} = 0, \tag{32}$$

which has a solution if and only if the determinant is zero, i.e., if

$$p_1(b)^2 - (1 - b^2)p_2(b)^2 = 0. \tag{33}$$

This is a quartic polynomial in only b and can consequently be solved in closed-form [11]. Given a solution for b we can then get a as $a = -p_1(b)/p_2(b)$ provided $p_2(b) \neq 0$. The original system in (27) is now reduced to a single univariate polynomial. This is our third polynomial system.

The two systems (27) and (29) can be solved using the action matrix method and solvers for these kinds of problems can be generated automatically [10, 12]. Since (33) is univariate it can be solved simply by finding the eigenvalues of the companion matrix or as mentioned above, using a closed-form solver. The univariate solver will have slightly better numerical accuracy since we have done all the eliminations algebraically exact, as opposed to the action matrix solvers that include online elimination steps based on Gaussian elimination.

3.4 Merging time-of-arrival data

In this section we show how to adapt time-of-arrival (ToA) datasets to the presented merging scheme. As first described in Section 2, let $m \in \mathbb{R}^{2n}$ denote the column-stacked coordinates of a set of anchors $\{x_i\}_1^n$. Similarly, let s denote the coordinates of a set of scan points $\{s_j\}_1^k$ and let $\tilde{d}_{ij}$ be a distance measurement between anchor x_i and scan point s_j. Also, assume that the optimal solution to the minimum least squares problem

$$\arg\min_{m,s} \sum_{ij} \left(\|x_i - s_j\| - \tilde{d}_{ij} \right)^2 \tag{34}$$

is denoted $m^\star$ and $s^\star$, and has the residuals $r_{opt} = r(m^\star, s^\star)$ where

$$r_{ij}(m, s) = \|x_i - s_j\| - \tilde{d}_{ij}. \tag{35}$$

Then the residuals can be linearized around $m^\star$ and $s^\star$ as

$$r(m, s) \approx r_{opt} + J \begin{bmatrix} m - m^\star \\ s - s^\star \end{bmatrix}, \tag{36}$$

where J is the Jacobian of $r(m, s)$ with respect to m and s. The problem can be reformulated as

$$r(m, s) \approx r_{opt} + \begin{bmatrix} J_m & J_s \end{bmatrix} \begin{bmatrix} \Delta m \\ \Delta s \end{bmatrix}. \tag{37}$$

Due to the nature of the ToA problem our coordinate system has three degrees of freedom (translation and rotation). Hence, to fix the coordinate system, we remove three columns from J_s, fixing the corresponding coordinates. To express the residuals as functions in only m we eliminate the scan points by minimizing the residual norm over Δs

$$\min_{\Delta s} \|r_{opt} + J_m \Delta m + J_s \Delta s\|^2 \tag{38}$$

which has the closed form solution

$$\begin{aligned} \Delta s &= -(J_s^T J_s)^{-1} (\underbrace{J_s^T r_{opt}}_{=0} + J_s^T J_m \Delta m) \\ &= -(J_s^T J_s)^{-1} J_s^T J_m \Delta m. \end{aligned} \tag{39}$$

Insertion into (37) yields

$$r(m, s) \approx r_{opt} + \underbrace{(I - J_s (J_s^T J_s)^{-1} J_s^T) J_m}_{=U} \Delta m, \tag{40}$$

where I denotes the identity matrix of proper size. In some cases the matrix U can be large and cumbersome to deal with. We therefor decompose it into $U = QL$ where L is an upper triangular matrix and Q is a unitary matrix. Consequently,

$$\begin{aligned} \|r(m, s)\|^2 &\approx \|Q^T r_{opt} + L\Delta m\|^2 \\ &= \|r_{opt}\|^2 + \|L\Delta m\|^2. \\ &\quad (r_{opt}^T Q L \Delta m = 0) \end{aligned} \tag{41}$$

If two datasets are to be merged the resulting anchor points can be found by minimizing the sum of the residual norms:

$$\min_m \|r_1(m, s)\|^2 + \|r_2(m, s')\|^2. \tag{42}$$

Using the approximation in (41) this is equivalent to the merging in (8), i.e., the L found above represents the linearization around the estimated anchor positions as in the previous sections.

4 Experiments

In Section 4.1 we investigate the numerical stability of the three non-linear solvers proposed in Section 3.3. In Sections 4.2 and 4.3 we compare five different methods for

registering and merging maps using simulated and real data. The five methods are described below.

Problem 3 can be expressed as a separable non-linear least squares problem. Given the rotation θ, (10) is linear in m and t and can be solved using the linear system

$$\underbrace{\begin{bmatrix} L_1 & 0 \\ L_2\hat{R} & L_2B \end{bmatrix}}_{H} \begin{bmatrix} m \\ t \end{bmatrix} = \underbrace{\begin{bmatrix} L_1 m_1 \\ L_2 m_2 \end{bmatrix}}_{y}. \tag{43}$$

Note that $\hat{R}$ and consequently H is dependent on θ. The θ that minimizes (10) is then found by maximizing the univariate non-linear expression [13]

$$\max_{\theta} y^T H (H^T H)^{-1} H^T y. \tag{44}$$

This can be done using standard iterative optimization methods. We call this method NLS.

In the second method we use singular value decomposition to find the rigid transformation (see e.g. [5]). After the two point sets are registered (9) is used to perform the merge. We call this method SVD_{Rt}. The rotation found in this method is also used as the initial θ in NSL.

If we take only the rotation from SVD_{Rt} and then perform the merging using (43) we get our third method called SVD_{R}. SVD_{Rt} and SVD_{R} are both linear methods but differ in that SVD_{R} takes into consideration the uncertainty of the points when finding the translation in (43).

The fourth method is a registration using SVD followed by a naive merging scheme where the covariances are ignored and the corresponding points of the two datasets are simply averaged: $m = (m_1 + m_2)/2$. We call this method AVG.

The fifth and proposed method is the univariate registration explained in Section 3.3 followed by merging using (9).

4.1 Numerical stability

To test the numerical stability of the three solvers proposed in Section 3.3, we generate random datasets consisting of $n = 10$ 2D points m_1 with coordinates in the range $[0, 10]$, ground truth transformation (R, t) with $\theta \in [0, 2\pi]$ and $t_x, t_y \in [-10, 10]$, and covariance matrices C_1 and C_2, where $C_i = K_i^T K_i$ and the elements of K_i are drawn from

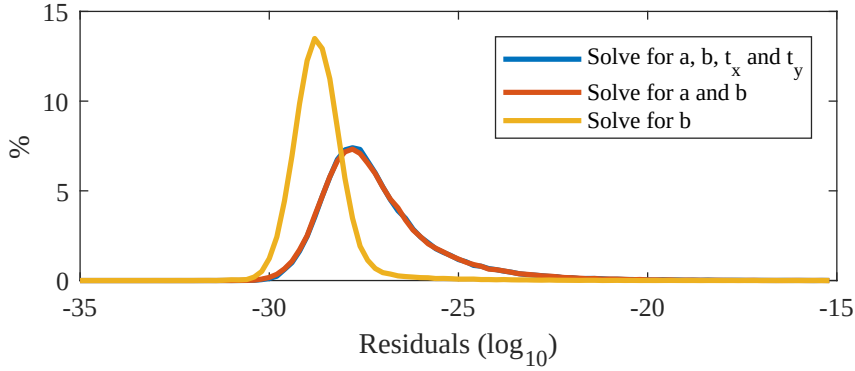


Figure 2: Residual error histogram for 100,000 runs of the three proposed solvers. The two first solvers (blue and red) overlap almost completely.

the range $[-\sigma, \sigma]$ with $\sigma = 1$. From these we get the second point set $m_2 = Rm_1 + t$ and the precision matrix $P = (RC_1R^T + C_2)^{-1}$. Since we have not explicitly introduced any errors, the residual in (19) can be attributed to numerical errors in the optimization. Figure 2 shows residual histograms for 100,000 runs of the three solvers. One can see that all solvers give residuals that are close to machine precision.

When the solvers for (27) and (29) are automatically generated Macaulay2 is used to first solve a random integer instance of the problem. Knowledge of this specific instance is then used when generating the solvers [10]. As a consequence of this, the solvers produced might only work for similar instances of the problem. This is the case for the two solvers produced here. When the covariance in the anchor points are isotropic and independent there are only two solutions to (27) and (29) instead of four, which the generated solvers fail to handle. The univariate solver however do not suffer from these issues while being just as numerically stable.

4.2 Simulated data

We test the proposed registration and merging scheme on simulated point sets with Gaussian errors. Similar to the previous section we generate random ground truth positions m with coordinates in the range $[0, 10]$ and transform them as $m' = Rm + t$ where $t_x, t_y \in [-10, 10]$. Gaussian noise is then added to the positions: $m_1 = m + e_1$ and $m_2 = m' + e_2$, where e_i are sampled from $\mathcal{N}(0, C_i)$. The random covariances are calculated as $C_i = K_i^T K_i$ where elements from K_i are in the range $[-\sigma, \sigma]$ for some constant σ. Figure 3 shows the RMS error in the merged positions for various values of σ. All

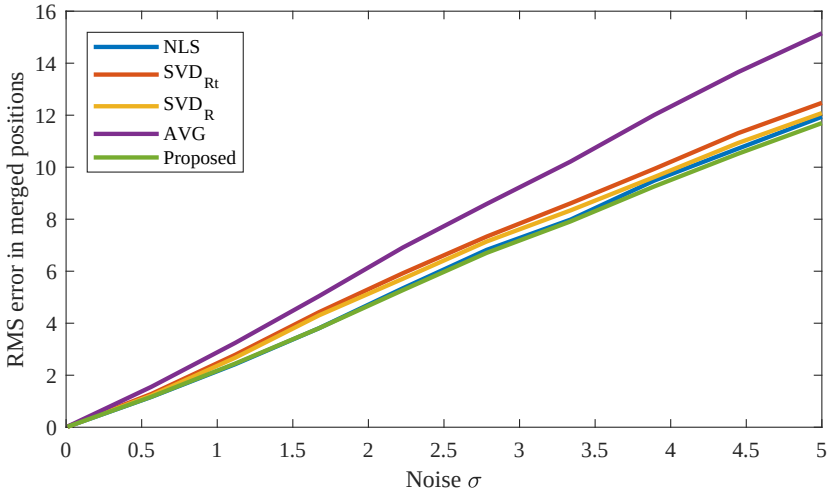


Figure 3: RMS error in merged position for various amounts of noise.

methods but the naive AVG produces similar results.

A second test was performed to see how the methods compare as the anisotropy of the Gaussian noise increases. To this end we define new covariance matrices

$$C_i = \begin{bmatrix} S_{i1}\Lambda S_{i1}^T & & & \\ & S_{i2}\Lambda S_{i2}^T & & \\ & & \ddots & \\ & & & S_{in}\Lambda S_{in}^T \end{bmatrix}, \tag{45}$$

where $S_{ij} \in \mathbb{R}^{2\times 2}$ are random rotation matrices and

$$\Lambda = \begin{bmatrix} \sqrt{\eta} & 0 \\ 0 & 1/\sqrt{\eta} \end{bmatrix}, \tag{46}$$

where η is the ratio between the minor and major axes of the Gaussian ellipses called the anisotropy ratio. Figure 4 shows the RMS error in the merged positions for various values of η. For isotropic errors ($\eta = 1$) all methods find the optimal solution. For higher anisotropy ratios the non-linear NLS and proposed methods perform similarly and better than the remaining linear ones. Note how SVD_R performs better than SVD_Rt. There is clearly a benefit in solving the translation during the merging step in (43) rather than during the registration.

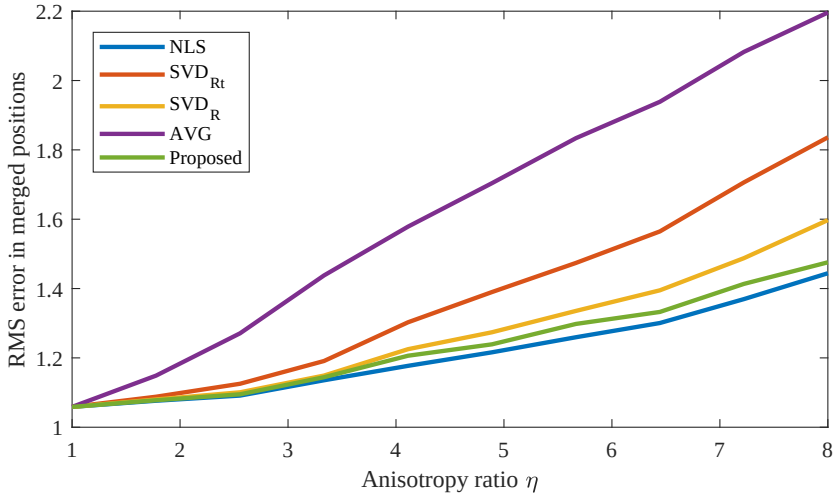


Figure 4: RMS error in merged position for various level of anisotropy.

4.3 Real data

A series of real ToA datasets were captured using an ultra-wideband (UWB) positioning system. Six anchors where placed in a plane covering an area of approximately 3×6 m. The ground truth positions of the anchors were found by accurately measuring the pairwise distance between them. A receiver was then moved in the same plane while distance measurements were recorded. From these measurements the anchors m and scan points s were estimated by solving (34) and the remainder of Section 3.4 was carried out to produce the upper triangular matrix L.

We present eight datasets constructed in this fashion. In three of them, called S1-S3, the receiver was moved in a straight line through the anchor setup. Three similar datasets, called S4-S6, were created where the receiver was moved in a straight line perpendicular to S1-S3. Finally, two longer datasets, called L1 and L2, were created where the receiver was moved freely around the anchor setup (see Figure 5). Pairs of these datasets were registered and merged using the five discussed methods. Table 2 shows the resulting RMS errors in merged position measured against the ground truth. All methods but the naive AVG produced similar errors.

Figure 6 shows the registered and merged anchors when using the proposed method on S1 and S4. The fact that the receiver moved in a straight line resulted in anisotropic covariances that are taken into account when merging. Similarly, Figure 7 shows the registered and merged anchors from datasets S1 and L1. The longer datasets resulted

Table 2: RMS Error (meters) in Anchor Positions When Registering and Merging Various Datasets.

Dataset	m_1	m_2	NLS	SVD_{Rt}	SVD_{R}	AVG	Proposed
S1 - S4	0.44	0.41	0.28	0.28	0.28	0.35	0.28
S2 - S5	0.42	0.39	0.26	0.27	0.27	0.34	0.27
S3 - S6	0.42	0.38	0.25	0.25	0.25	0.33	0.25
S1 - L1	0.44	0.18	0.18	0.18	0.18	0.28	0.18
L2 - L1	0.18	0.18	0.17	0.17	0.17	0.17	0.17

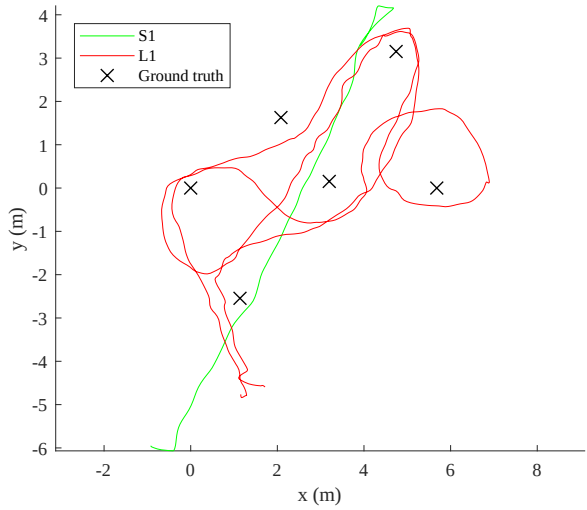


Figure 5: The estimated scan points and ground truth anchor positions for the datasets S1 (green) and L1 (red).

in lower and more isotropic covariances and, as a consequence, the merged points are drawn towards the more reliable L1.

5 Conclusions

In this paper we presented five different methods for performing rigid registration of 2D point sets that takes into consideration the covariance of the points. We also presented a merging scheme for combining multiple such point sets in an optimal way and showed how this can be applied to ToA data. Synthetic data showed that the proposed method performed better than simpler linear methods for very anisotropic noise. When the

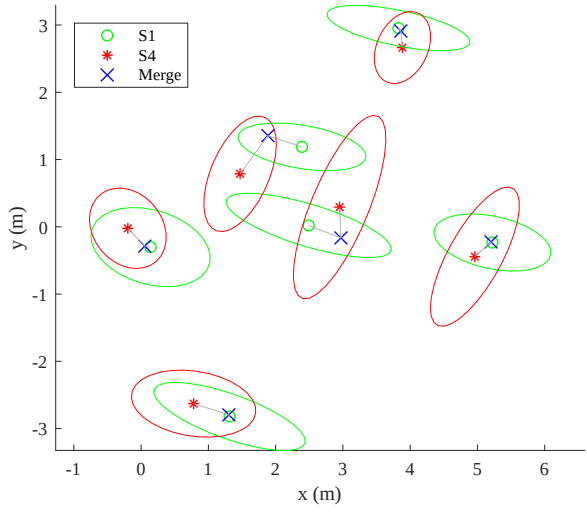


Figure 6: The registered and merged anchors (blue) resulting from datasets S1 (green) and S4 (red) using the proposed method. The ellipses illustrate the covariance of the anchors.

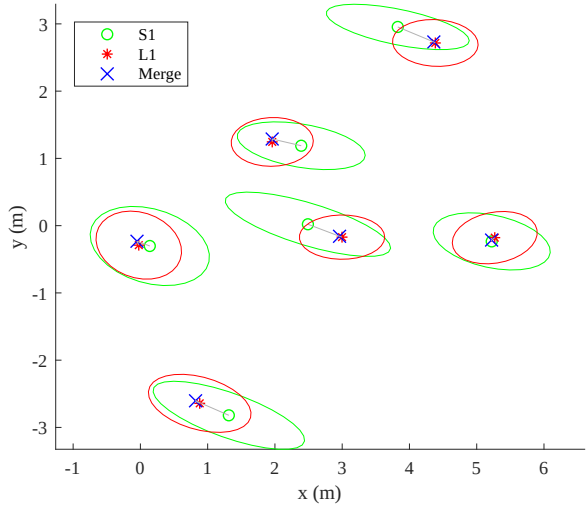


Figure 7: The registered and merged anchors (blue) resulting from datasets S1 (green) and L1 (red) using the proposed method. The ellipses illustrate the covariance of the anchors.

noise is more isotropic real and synthetic data showed that all methods but the naive AVG method performed similarly.

Even though the full problem in (10) is highly non-linear, as indicated by Table 1, simple methods produce good results. In particular, if the rotation of the transformation is found without considering the uncertainties of the points, then the problem becomes linear (SVD_R) and can be solved efficiently with decent results even for anisotropic noise.

However, it is possible that in higher dimensions the non-linearity increases and more complicated non-linear methods, such as the proposed or NLS methods, are required. We did not investigate this as the proposed method does not easily extend to higher dimensions.

References

[1] C. Olsson and O. Enqvist, "Stable structure from motion for unordered image collections," in *Scandinavian Conf. on Image Analysis*, 2011.

[2] K. Batstone, M. Oskarsson, and K. Åström, *Robust Time-of-Arrival Self Calibration with Missing Data and Outliers*. United States: Institute of Electrical and Electronics Engineers Inc., 9 2016, pp. 2370–2374.

[3] Z. Simayijiang, S. Segerblom Rex, Y. Kuang, F. Andersson, and K. Åström, "An automatic system for acoustic microphone geometry calibration based on minimal solvers," *ARXIV*, 10 2016.

[4] X. Li, K. Batstone, K. Åström, M. Oskarsson, C. Gustafson, and F. Tufvesson, *Robust Phase-Based Positioning Using Massive MIMO with Limited Bandwidth*. IEEE–Institute of Electrical and Electronics Engineers Inc., 6 2017.

[5] B. Horn, H. M. Hilden, and S. Negahdaripour, "Closed-form solution of absolute orientation using orthonormal matrices," *Journal of the Optical Society of America A*, vol. 5, no. 7, pp. 1127–1135, 1988.

[6] W. Kabsch, "A solution for the best rotation to relate two sets of vectors," *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, vol. 32, no. 5, pp. 922–923, 1976.

[7] C. Olsson, F. Kahl, and M. Oskarsson, "The registration problem revisited: Optimal solutions from points, lines and planes," in *CVPR*, vol. I, New York City, USA, 2006, pp. 1206–1213.

[8] F. Pomerleau, F. Colas, R. Siegwart *et al.*, "A review of point cloud registration algorithms for mobile robotics," *Foundations and Trends® in Robotics*, vol. 4, no. 1, pp. 1–104, 2015.

[9] D. R. Grayson and M. E. Stillman, "Macaulay2, a software system for research in algebraic geometry," Available at http://www.math.uiuc.edu/Macaulay2/.

[10] V. Larsson, K. Åström, and M. Oskarsson, "Efficient solvers for minimal problems by syzygy-based reduction," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 2383–2392.

[11] M. Abramowitz, *Handbook of Mathematical Functions, With Formulas, Graphs, and Mathematical Tables,*. New York, NY, USA: Dover Publications, Inc., 1974, ch. 3.8. Algebraic Equations, pp. 17–18.

[12] Z. Kukelova, M. Bujnak, and T. Pajdla, "Automatic generator of minimal problem solvers," in *European Conference on Computer Vision*. Springer, 2008, pp. 302–315.

[13] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1993.

Paper II

Upgrade Methods for Stratified Sensor Network Self-calibration

Martin Larsson[1,2], Gabrielle Flood[1], Magnus Oskarsson[1], Kalle Åström[1]

[1] *Centre for Mathematical Sciences, Lund University, Lund, Sweden*
[2] *Combain Mobile AB*

Abstract: Estimating receiver and sender positions is often solved using a stratified, two-tiered approach. In the first step the problem is converted to a low-rank matrix estimation problem. The second step can be seen as an affine upgrade. This affine upgrade is the focus of this paper. In the paper new efficient algorithms for solving for the upgrade parameters using minimal data are presented. It is also shown how to combine such solvers as initial estimates, either directly or after a hypothesis and test step, in optimization of likelihood. The system is verified on both real and synthetic data.

Keywords: Time-of-arrival, Time-difference-of-arrival, RANSAC, Minimal Problems, Calibration

This work was partially supported by the following funding bodies: strategic research projects ELLIIT and eSSENCE, Swedish Foundation for Strategic Research project - Semantic Mapping and Visual Navigation for Smart Robots - (grant no. RIT15-0038), Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation.

Reprinted from *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, M. Larsson, G. Flood, M. Oskarsson, and K. Åström, Upgrade Methods for Stratified Sensor Network Self-Calibration, 4851-4855, Copyright 2020, with permission from IEEE.

1 Introduction

The problem of estimating receiver-sender node positions from radio or sound signals is a key issue in different applications such as microphone array calibration, radio antenna array calibration, mapping and positioning [1]. If all senders and receivers are synchronized, it is possible to obtain absolute distance measurements between senders and receivers. These measurements can be used for self-calibration and such problems (Time-of-Arrival problems, TOA) have been studied in, e.g., [2–17]. Some variants of this TOA problem are (i) TDOA - if the receivers are synchronized, whereas the senders are unsynchronized [18–20]; (ii) (COTDOA) - constant offset time-difference-of-arrival [21]; and (iii) UTDOA - if neither senders nor receivers are calibrated [22].

A popular strategy to analyze and solve these tasks is to follow a two-tiered stratified approach [18, 21, 23]. The first part of this approach is based on solving a relaxed version of the problem. For this part it is possible to estimate the unknown offsets in the TDOA, COTDOA and UTDOA cases. The first part also involves estimating a low rank decomposition of a matrix derived from the measurements and offsets [24, 25]. The rank here depends on the minimum of the dimensions of the affine spans of the receivers and of the senders. Thus, for the general 3D case the rank is three, whereas the rank is two if, e.g., the receivers are coplanar and one if, e.g., the senders are colinear.

The second part of the two-tiered approach can be seen as an affine upgrade and it involves estimating a few parameters. This part is common for the cases of TOA, TDOA, COTDOA and UTDOA, since the offsets have already been estimated in the first part. Here, it is also possible to handle degenerate cases such as when the dimensions of the affine spans of the receivers and senders are different [26, 27]. The stratified approach is an efficient way of separating the full calibration problem into several well-defined sub-problems.

In this paper we study the upgrade problem, i.e., the second part of the two-tiered approach, and provide solvers for the most interesting minimal cases.[1] We also demonstrate how such solvers can be used in combination with nonlinear least squares optimization, [28, 29], to produce efficient algorithms for these upgrades. This can be used for all above mentioned stratified sensor network self-calibration problems.

[1]The solvers were implemented in C++ and MATLAB and the code is available at https://github.com/martinkjlarsson/upgrade-methods.

2 A Stratified approach to self calibration

The problem we address involves m receiver positions $\boldsymbol{r}_i \in \mathbb{R}^3$, $i = 1, \ldots, m$, n sender positions $\boldsymbol{s}_j \in \mathbb{R}^3$, $j = 1, \ldots, n$, and possibly unknown offsets. This could for example represent the microphone positions and locations of sound emissions, respectively. The arrival time of a sound j to receiver i is denoted t_{ij} and the time that sound j is emitted is τ_j. Multiplying the travel time $t_{ij} - \tau_j$ with the speed v of the signal, we obtain the distance between senders and receivers

$$v(t_{ij} - \tau_j) = \|\boldsymbol{r}_i - \boldsymbol{s}_j\|_2, \tag{1}$$

where $\|.\|_2$ denotes the ℓ^2-norm. The speed v is throughout the paper assumed to be known and constant. Assume that we have measurements z_{ij} of $v(t_{ij} - \tau_j)$. Then we have

$$z_{ij} = \|\boldsymbol{r}_i - \boldsymbol{s}_j\|_2. \tag{TOA}$$

Estimating $\boldsymbol{r}_i$ and $\boldsymbol{s}_j$ from z_{ij} is known as the node calibration problem. For the cases of COTDOA, TDOA and UTDOA the measurement equations are similar:

$$z_{ij} = \|\boldsymbol{r}_i - \boldsymbol{s}_j\|_2 + o, \tag{COTDOA}$$

$$z_{ij} = \|\boldsymbol{r}_i - \boldsymbol{s}_j\|_2 + o_j, \tag{TDOA}$$

$$z_{ij} = \|\boldsymbol{r}_i - \boldsymbol{s}_j\|_2 + q_i + o_j. \tag{UTDOA}$$

The first part of the stratified approach involves estimating the offsets and will not be covered in this paper. We can thus assume that these offsets are known and subtract them from z_{ij} to get the actual distances d_{ij}. Additionally, since the distances are assumed to be positive we can without loss of generality square them and for all the cases above we get the problem

$$d_{ij}^2 = \|\boldsymbol{r}_i - \boldsymbol{s}_j\|_2^2 = \boldsymbol{r}_i^T \boldsymbol{r}_i - 2\boldsymbol{r}_i^T \boldsymbol{s}_j + \boldsymbol{s}_j^T \boldsymbol{s}_j. \tag{2}$$

In addition to finding the offsets, the first step also includes finding a solution to the following relaxed problem

$$d_{ij}^2 = -2\boldsymbol{u}_i^T \boldsymbol{v}_j + a_j + b_i, \tag{3}$$

where $\boldsymbol{u}_i$ and $\boldsymbol{v}_j$ are columns of two sought matrices $\mathbf{U} \in \mathbb{R}^{3\times m}$ and $\mathbf{V} \in \mathbb{R}^{3\times n}$, respectively, and $\boldsymbol{a}^T \in \mathbb{R}^n$ and $\boldsymbol{b} \in \mathbb{R}^m$ only depend on data. Due to noise there is typically not an exact solution to (3). Furthermore, the matrix $-2\mathbf{U}^T\mathbf{V}$ forms a low-rank approximation of the so called double compaction matrix $\mathbf{M}$ [18], where the rank is determined by the smallest dimension of the affine spans of the receivers and senders. For our purposes the rank will be three. The second part in the stratified approach is to take a solution to the relaxed problem and upgrade it to find the receivers and senders. This second part is the main focus of the paper.

3 Upgrade

Let $\mathbf{R}$ and $\mathbf{S}$ be the matrices whose columns are $\boldsymbol{r}_i$ and $\boldsymbol{s}_j$, respectively. The problem of upgrading a relaxed solution $(\mathbf{U}, \mathbf{V}, \boldsymbol{a}, \boldsymbol{b})$ to a solution in $(\mathbf{R}, \mathbf{S})$ was first introduced in [11] and later improved upon in [17]. In this section we generalize the problem slightly and provide conditions which must be satisfied in the relaxed solution for the upgrade to work. We will assume that the receivers and senders are points in 3D but the scheme generalizes to any dimension. We start by introducing the matrices

$$\mathbf{C}_r = \mathbf{I} - \boldsymbol{w}_r \mathbf{1}^T, \tag{4}$$

$$\mathbf{C}_s = \mathbf{I} - \boldsymbol{w}_s \mathbf{1}^T, \tag{5}$$

where $\boldsymbol{w}_r \in \mathbb{R}^m$ and $\boldsymbol{w}_s \in \mathbb{R}^n$ are vectors such that $\boldsymbol{w}_r^T \mathbf{1} = \boldsymbol{w}_s^T \mathbf{1} = 1$. Additionally, let $\boldsymbol{r}_0 = \mathbf{R}\boldsymbol{w}_r$ and $\boldsymbol{s}_0 = \mathbf{S}\boldsymbol{w}_s$ be affine combinations of the receiver and sender positions, respectively. The double compaction matrix, $\mathbf{M}$, can then be expressed in the distances, $\mathbf{D}$, as

$$\mathbf{M} = \mathbf{C}_r^T \mathbf{D}^{\circ 2} \mathbf{C}_s = \mathbf{D}^{\circ 2} - \mathbf{1}\boldsymbol{a} - \boldsymbol{b}\mathbf{1}^T + c\mathbf{1}\mathbf{1}^T \tag{6}$$

where $\boldsymbol{a} = \boldsymbol{w}_r^T \mathbf{D}^{\circ 2}$, $\boldsymbol{b} = \mathbf{D}^{\circ 2} \boldsymbol{w}_s$ and $c = \boldsymbol{w}_r^T \mathbf{D}^{\circ 2} \boldsymbol{w}_s$. Here $\mathbf{D}^{\circ 2}$ denotes the element-wise square of $\mathbf{D}$. By inserting (2) we get, after some simplifications,

$$\mathbf{M} = -2(\mathbf{R} - \boldsymbol{r}_0 \mathbf{1}^T)^T (\mathbf{S} - \boldsymbol{s}_0 \mathbf{1}^T), \tag{7}$$

$$a_j = \boldsymbol{w}_r^T \operatorname{diag}(\mathbf{R}^T \mathbf{R}) - 2\boldsymbol{r}_0^T \boldsymbol{s}_j + \boldsymbol{s}_j^T \boldsymbol{s}_j, \tag{8a}$$

$$b_i = \boldsymbol{r}_i^T \boldsymbol{r}_i - 2\boldsymbol{r}_i^T \boldsymbol{s}_0 + \boldsymbol{w}_s^T \operatorname{diag}(\mathbf{S}^T \mathbf{S}), \tag{8b}$$

$$c = \boldsymbol{w}_r^T \operatorname{diag}(\mathbf{R}^T \mathbf{R}) - 2\boldsymbol{r}_0^T \boldsymbol{s}_0 + \boldsymbol{w}_s^T \operatorname{diag}(\mathbf{S}^T \mathbf{S}). \tag{8c}$$

From (7) we see that, by decomposing $\mathbf{M} = -2\mathbf{U}^T\mathbf{V}$ using, e.g., singular value decomposition, we can find the receiver and sender positions up to some full rank transformation $\mathbf{L}$ and reference points $\boldsymbol{r}_0$ and $\boldsymbol{s}_0$, such that

$$\mathbf{R} = \mathbf{L}^{-T}\mathbf{U} + \boldsymbol{r}_0 \mathbf{1}^T \quad \text{and} \quad \mathbf{S} = \mathbf{L}\mathbf{V} + \boldsymbol{s}_0 \mathbf{1}^T. \tag{9}$$

3.1 Conditioning the relaxed problem

Due to the larger gauge freedom in the relaxed problem than in the original problem some constraints need to be added before it fits into the upgrading scheme. Firstly, we see from (9) that

$$\boldsymbol{u}_0 \triangleq \mathbf{U}\boldsymbol{w}_r = \mathbf{L}^T(\mathbf{R} - \boldsymbol{r}_0 \mathbf{1}^T)\boldsymbol{w}_r = \mathbf{0}, \tag{10}$$

$$\boldsymbol{v}_0 \triangleq \mathbf{V}\boldsymbol{w}_s = \mathbf{L}^{-1}(\mathbf{S} - \boldsymbol{s}_0 \mathbf{1}^T)\boldsymbol{w}_s = \mathbf{0}. \tag{11}$$

This results from our definition of $\mathbf{M}$ and is not true for a general $\mathbf{U}$ and $\mathbf{V}$. However, we can ensure that the conditions are met by translating $\mathbf{U}$ and $\mathbf{V}$:

$$\mathbf{U} \to \mathbf{U} - \boldsymbol{u}_0 \mathbf{1}^T, \quad \mathbf{V} \to \mathbf{V} - \boldsymbol{v}_0 \mathbf{1}^T \tag{12}$$

and compensating $\boldsymbol{a}$, $\boldsymbol{b}$ and c accordingly

$$\boldsymbol{a} \to \boldsymbol{a} - 2\boldsymbol{u}_0^T \mathbf{V}, \ \boldsymbol{b} \to \boldsymbol{b} - 2\mathbf{U}^T \boldsymbol{v}_0, \ c \to c - 2\boldsymbol{u}_0^T \boldsymbol{v}_0. \tag{13}$$

Secondly, from (6) we see that $c = \boldsymbol{a}\boldsymbol{w}_s = \boldsymbol{w}_r^T \boldsymbol{b}$ which can be ensured by adding appropriate constants to $\boldsymbol{a}$, $\boldsymbol{b}$ and c, making sure (3) still holds.

3.2 Solving for the upgrade parameters

We are now left with finding the unknowns $\mathbf{L}$, $\boldsymbol{r}_0$ and $\boldsymbol{s}_0$ using the equations in (8). Any solution to $\mathbf{R}$ and $\mathbf{S}$ is only determined up to a rigid transform. To fix the translational part of the transform we let $\boldsymbol{r}_0 = \mathbf{0}$. We then parameterize the remaining unknowns as $\boldsymbol{s}_0 = \mathbf{L}\boldsymbol{q}$ and $\mathbf{H} = (\mathbf{L}^T\mathbf{L})^{-1}$ where $\boldsymbol{q} \in \mathbb{R}^3$ and where $\mathbf{H} \in \mathbb{R}^{3\times 3}$ is a symmetric matrix.

To simplify the equations we will henceforth assume that $\boldsymbol{w}_r$ and $\boldsymbol{w}_s$ are zero vectors except for one element which is set to 1, i.e., $\boldsymbol{r}_0 = \boldsymbol{r}_i$ for some $1 \le i \le m$ and $\boldsymbol{s}_0 = \boldsymbol{s}_j$ for some $1 \le j \le n$. The equations in (8) can now be written as

$$a_j = (\boldsymbol{v}_j + \boldsymbol{q})^T \mathbf{H}^{-1} (\boldsymbol{v}_j + \boldsymbol{q}), \tag{14a}$$

$$b_i = \boldsymbol{u}_i^T \mathbf{H} \boldsymbol{u}_i - 2\boldsymbol{u}_i^T \boldsymbol{q} + \boldsymbol{q}^T \mathbf{H}^{-1} \boldsymbol{q}, \tag{14b}$$

$$c = \boldsymbol{q}^T \mathbf{H}^{-1} \boldsymbol{q}. \tag{14c}$$

$\mathbf{H}$ is symmetric and can together with $\boldsymbol{q}$ be parameterized in nine unknowns. However, the equations above are not independent due to the two linear constraints $c = \boldsymbol{a}\boldsymbol{w}_s = \boldsymbol{w}_r \boldsymbol{b}$. Consequently, we need $m + n + 1 \ge 11$ for the problem to be well-defined.

If we subtract c from the first two equations,

$$a_j - c = \boldsymbol{v}_j^T \mathbf{H}^{-1} \boldsymbol{v}_j - 2\boldsymbol{v}_j^T \mathbf{H}^{-1} \boldsymbol{q}, \tag{15a}$$

$$b_i - c = \boldsymbol{u}_i^T \mathbf{H} \boldsymbol{u}_i - 2\boldsymbol{u}_i^T \boldsymbol{q}, \tag{15b}$$

$$c = \boldsymbol{q}^T \mathbf{H}^{-1} \boldsymbol{q}, \tag{15c}$$

we get $m - 1$ linear constraints on the unknowns from (15b). With $m \ge 10$ the problem becomes linear and if $m < 10$, $\mathbf{H}$ and $\boldsymbol{q}$ can be parameterized in $10 - m$ unknowns α_k for $k = 1, \ldots, 10 - m$:

$$\mathbf{H} = \mathbf{H}_0 + \sum_{k=1}^{10-m} \alpha_k \mathbf{H}_k, \quad \boldsymbol{q} = \boldsymbol{q}_0 + \sum_{k=1}^{10-m} \alpha_k \boldsymbol{q}_k. \tag{16}$$

If we multiply the nonlinear equations (15a) and (15c) with $\det(\mathbf{H})$ they become polynomial in α_k and can be solved, e.g, using action matrix methods [9]. We might have introduced additional solutions for the case where $\det(\mathbf{H}) = 0$. However, these can be removed using saturation as described in [17]. Once $\mathbf{H}$ is solved for, $\mathbf{L}$ can be found using Cholesky decomposition and finally the receiver and sender positions can be attained using (9).

4 Minimal Solvers

In [11] two problems are considered. The first involves six receivers and four senders and results in five linear equations corresponding to (15b), three nonlinear equations corresponding to (15a) and one nonlinear equation corresponding to (15c). Because of this, we introduce a new notation and denote this problem with `531`. Similarly, the second problem in [11] involves five receivers and five senders which we would denote `441`.

Problem `531` is minimal in the sense that there are 24 distance measurements and 24 degrees of freedom in $\mathbf{R}$ and $\mathbf{S}$. However, problems `531` and `441` are both minimal in the sense that they have nine equations and nine unknowns in (15). There is indeed a total of 19 possible minimal configurations of the equations, 10 of which are listed in Table 1. The first solver, `900`, is linear and subsequent solvers get increasingly nonlinear. Note that, although we used the same scheme as in [17], our templates for setting up the action matrices for solvers `531` and `441` are slightly smaller.

5 Validation

5.1 Numerical stability of solvers

To test the numerical stability of the solvers we generated 10 random receiver and 10 random sender positions within a unit cube, $\mathbf{R}, \mathbf{S} \in [0, 1]^{3\times 10}$, from which we could calculate the distance matrix $\mathbf{D}$. The relaxed version of the problem ($\mathbf{U}$, $\mathbf{V}$, $\boldsymbol{a}$, $\boldsymbol{b}$ and c) could then be found as described in Section 3. For every solver in Table 1 a minimal sample of the relaxed problem was taken and solved. From the estimated receiver and sender positions the RMS error in the estimated distances was calculated. Figure 1 shows histograms of the RMS errors over multiple runs. As can be seen the `900`, `810` and `801` solvers performed best and the numerical stability generally worsens as more nonlinear equations are added.

Table 1: The number of solutions and template sizes for the solvers, with and without saturation, together with their execution time.

	# Solutions		Template size		Exec.
Solver	no sat.	sat.	no sat.	sat.	time
900	1	-	-	-	39 µs
810	3	3	-	-	130 µs
801	4	4	-	-	130 µs
720	9	9	12 × 21	12 × 21	170 µs
711	12	12	16 × 28	16 × 28	210 µs
630	21	17	88 × 109	112 × 129	600 µs
621	30	26	122 × 152	156 × 182	1.2 ms
540	∞	21	-	310 × 331	5.3 ms
531	∞	38	-	493 × 531	19 ms
441	∞	42	-	817 × 859	72 ms

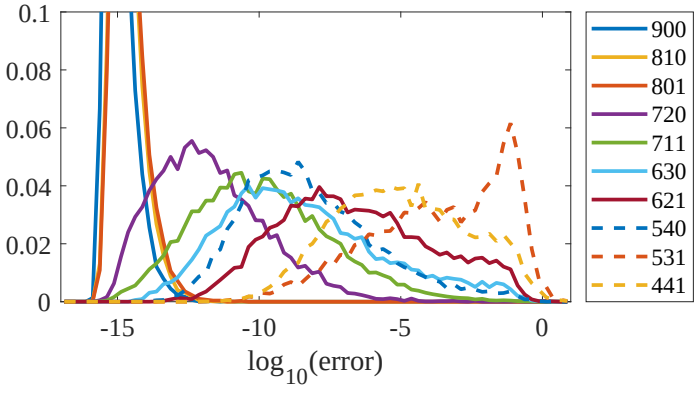


Figure 1: Error histograms for all solvers in Table 1 when provided with noise free data. Note that the graphs for 900, 810 and 801 extend beyond the plot.

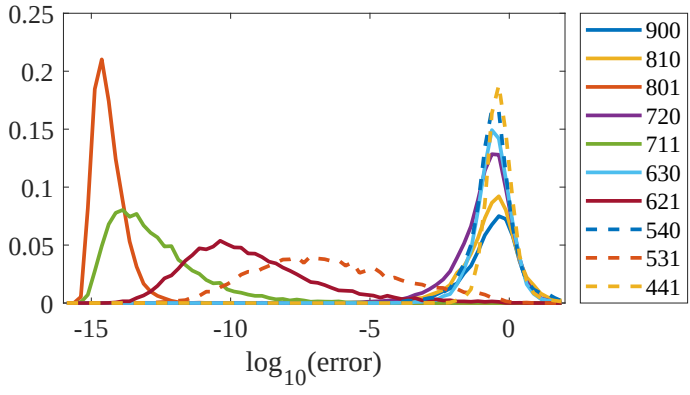


Figure 2: Error histograms for all solvers in Table 1 when provided with noise free data from points located on a sphere.

5.2 Degenerate configurations

The solvers behave differently when it comes to certain degenerate configurations of **R** and **S**. For example, during testing we observed that the rank of the linear system resulting from (15b) never exceeds eight when the receivers are confined to certain two-dimensional manifolds (*e.g.* ellipsoids, paraboloids, hyperboloids). Consequently, the linear 900 solver performs poorly for such configurations. More generally, one could imagine a distance matrix **D** for which there are several possible embeddings of **R** and **S**. If the number of embeddings exceeds the number of solutions of a solver, that solver might not find the correct embedding.

Figure 2 shows RMS distance errors for the solvers in Table 1 when the receivers and senders are located on a unit sphere. It can be seen that only solvers that include equation (15c) are stable.

5.3 Minimal solvers in a RANSAC system

In this section we show how our upgrade solvers can be used in a simple system. We assume that we have a solution to the first part in the stratified approach, i.e., we have a low rank approximation solution (**U**, **V**, $\boldsymbol{a}$, $\boldsymbol{b}$ and c), and want to upgrade this solution to actual receiver and sender positions. From the given (**U**, **V**) (and a chosen minimal solver) we sample minimal configurations and solve using the chosen upgrade solver. From the solution we can estimate the distance errors. We then iterate a small number of times and choose the best solution. The results of this can be seen in Figure 3 for three

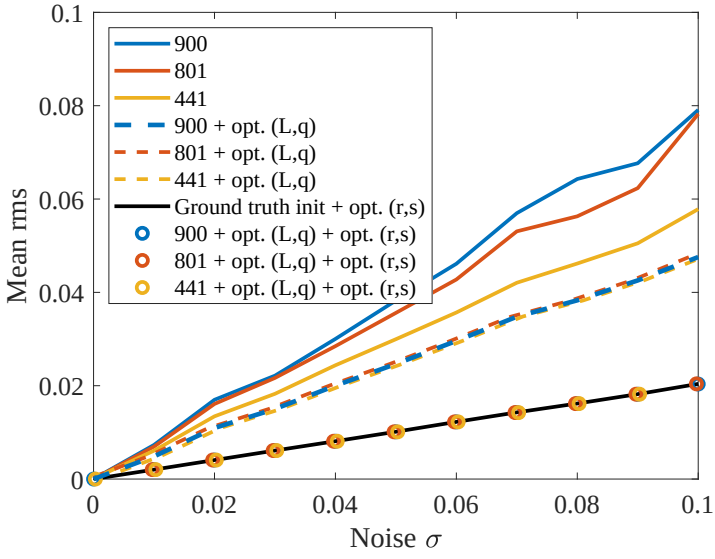


Figure 3: Results using different minimal upgrade solvers as initialization, with subsequent nonlinear optimization on synthetic data.

example solvers. Here we have used synthetic data, and show the results for varying levels of added noise. The graph also shows the results after nonlinear optimization over the upgrade parameters $\mathbf{L}$ and $\boldsymbol{q}$ respectively after subsequent nonlinear optimization over the receiver and sender positions. The rationale for optimizing over $\mathbf{L}$ and $\boldsymbol{q}$ first, is that this is in general faster and more robust, since it only involves nine parameters. For larger problems, optimizing over the full receiver and sender positions would involve hundreds or thousands of parameters.

5.4 Real data from UWB

We evaluate the solvers on real TOA datasets gathered with an ultra-wideband (UWB) setup. Six senders were kept stationary in an area of 3×3×2 meters while a receiver was moved through the setup. Ground truth positions were gathered using an optical motion capture system. The noise in the UWB measurements corresponds roughly to $\sigma = 0.26$. The RANSAC scheme discussed in the previous subsection was used to find a good initialization with three selected solvers. Table 2 shows the RMS errors in sender positions from the solver initialization, after nonlinear optimization over $\mathbf{L}$ and $\boldsymbol{q}$, and after nonlinear optimization over $\mathbf{L}$ and $\boldsymbol{q}$ followed by $\mathbf{R}$ and $\mathbf{S}$. None of the solvers performed best for all datasets and after optimization they all performed similarly.

Table 2: RMS errors (meters) in sender positions for three real UWB datasets.

	Init			Opt. $(\mathbf{L}, \boldsymbol{q})$			Opt. $(\mathbf{L}, \boldsymbol{q}), (\mathbf{R}, \mathbf{S})$		
Data	900	801	441	900	801	441	900	801	441
1	0.75	1.85	1.01	0.96	0.95	0.96	0.33	0.32	0.33
2	0.62	0.73	0.48	0.38	0.38	0.38	0.28	0.28	0.28
3	0.41	0.40	0.57	0.49	0.49	0.49	0.21	0.21	0.21

6 Conclusions

In this paper, several novel solvers have been constructed to efficiently solve the upgrade step in a two-tiered stratified approach to solving TOA, TDOA and COTDOA problems. These have been verified using simulated data to test the solver and real experimental data to test our algorithms in realistic scenarios.

For future work, it would be interesting to further study how best to combine low rank estimation problems for TOA, TDOA and similar problems with affine upgrade methods. This would make it possible to produce systems that could solve a wide variety of estimation problems (TOA, TDOA, UTDOA) with a wide variety of assumptions on senders and receivers, e.g., spanning 3D or being coplanar or colinear.

References

[1] A. Plinge, F. Jacob, R. Haeb-Umbach, and G. A. Fink, "Acoustic microphone geometry calibration: An overview and experimental evaluation of state-of-the-art algorithms," *IEEE Signal Processing Magazine*, vol. 33, no. 4, pp. 14–29, 2016.

[2] S. T. Birchfield and A. Subramanya, "Microphone array position calibration by basis-point classical multidimensional scaling," *IEEE transactions on Speech and Audio Processing*, vol. 13, no. 5, 2005.

[3] D. Niculescu and B. Nath, "Ad hoc positioning system (aps)," in *Global Communications Conference (GLOBECOM)*, 2001.

[4] V. C. Raykar, I. V. Kozintsev, and R. Lienhart, "Position calibration of microphones and loudspeakers in distributed computing platforms," *IEEE transactions on Speech and Audio Processing*, vol. 13, no. 1, 2005.

[5] M. Crocco, A. Del Bue, M. Bustreo, and V. Murino, "A closed form solution to the microphone position self-calibration problem," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2012.

[6] J. C. Chen, R. E. Hudson, and K. Yao, "Maximum likelihood source localization and unknown sensor location estimation for wideband signals in the near-field," *IEEE transactions on Signal Processing*, vol. 50, 2002.

[7] P. Pertila, M. Hamalainen, and M. Mieskolainen, "Passive temporal offset estimation of multichannel recordings of an ad-hoc microphone array," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 21, no. 11, pp. 2393–2402, Nov. 2013.

[8] M. Pollefeys and D. Nister, "Direct computation of sound and microphone locations from time-difference-of-arrival data," in *International conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2008.

[9] H. Stewénius, "Gröbner basis methods for minimal problems in computer vision," Ph.D. dissertation, Lund University, APR 2005.

[10] Y. Kuang, E. Ask, S. Burgess, and K. Åström, "Understanding toa and tdoa network calibration using far field approximation as initial estimate," in *ICPRAM*, 2012.

[11] Y. Kuang, S. Burgess, A. Torstensson, and K. Åström, "A complete characterization and solution to the microphone position self-calibration problem," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2013.

[12] X. Li, E. Leitinger, M. Oskarsson, K. Åström, and F. Tufvesson, "Massive mimo-based localization and mapping exploiting phase information of multipath components," *IEEE Transactions on Wireless Communications*, vol. 18, no. 9, pp. 4254–4267, 2019.

[13] K. Batstone, M. Oskarsson, and K. Åström, "Towards real-time time-of-arrival self-calibration using ultra-wideband anchors," in *International conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2017.

[14] S. Burgess, K. Åström, M. Högström, B. Lindquist, and R. Ljungberg, "Smartphone positioning in multi-floor environments without calibration or added infrastructure," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2016.

[15] Z. Simayijiang, S. Burgess, Y. Kuang, and K. Åström, "Toa-based self-calibration of dual-microphone array," *IEEE Journal on Selected Topics in Signal Processing*, vol. 9, no. 5, pp. 791–801, 2015.

[16] ——, "Minimal solutions for dual microphone rig self-calibration," in *European Signal Processing Conference (EURASIP)*, 2014.

[17] V. Larsson, K. Åström, and M. Oskarsson, "Polynomial solvers for saturated ideals," in *International Conference on Computer Vision (ICCV)*. IEEE, 2017.

[18] Y. Kuang and K. Astrom, "Stratified sensor network self-calibration from tdoa measurements," in *European Signal Processing Conference (EUSIPCO)*, 2013.

[19] Z. Simayijiang, S. Segerblom Rex, Y. Kuang, F. Andersson, and K. Åström, "An automatic system for acoustic microphone geometry calibration based on minimal solvers," *ARXIV*, 10 2016.

[20] L. Wang, T. Hon, J. D. Reiss, and A. Cavallaro, "Self-localization of ad-hoc arrays using time difference of arrivals," *IEEE Transactions on Signal Processing*, vol. 64, no. 4, pp. 1018–1033, Feb 2016.

[21] K. Batstone, G. Flood, T. Beleyur, V. Larsson, H. Goerlitz, M. Oskarsson, and K. Astrom, "Robust self-calibration of constant offset time-difference-of-arrival," in *International conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019.

[22] S. Burgess, Y. Kuang, and K. Åström, "Node localization in unsynchronized time of arrival sensor networks," in *International Conference on Pattern Recognition (ICPR)*, 2012.

[23] K. Batstone, M. Oskarsson, and K. Åström, "Robust time-of-arrival self calibration with missing data and outliers," in *European Signal Processing Conference (EUSIPCO)*, 2016.

[24] Z. Simayijiang, F. Andersson, and K. Åström, "Offset estimation for microphone localization using alternating projections," in *Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2016.

[25] F. Jiang, Y. Kuang, and K. Åström, "Time delay estimation for tdoa self-calibration using truncated nuclear norm regularization," in *International conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013.

[26] S. Burgess, Y. Kuang, and K. Åström, "Toa sensor network self-calibration for receiver and transmitter spaces with difference in dimension," *Signal Processing*, vol. 107, no. Online 11 June 2014, pp. 33–42, 2015.

[27] E. Ask, Y. Kuang, and K. Åström, "A unifying approach to minimal problems in collinear and planar tdoa sensor network self-calibration," in *European Signal Processing Conference (EUSIPCO)*, 2014.

[28] R. Biswas and S. Thrun, "A passive approach to sensor network localization," in *International Conference on Intelligent Robots and Systems (IROS)*, 2004.

[29] J. Wendeberg, F. Hoflinger, C. Schindelhauer, and L. Reindl, "Anchor-free tdoa self-localization," in *International conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2011.

Paper III

Fast and Robust Stratified Self-Calibration Using Time-Difference-of-Arrival Measurements

Martin Larsson[1,2], Gabrielle Flood[1], Magnus Oskarsson[1], Kalle Åström[1]

[1] *Centre for Mathematical Sciences, Lund University, Lund, Sweden*
[2] *Combain Mobile AB*

Abstract: In this paper we study the problem of estimating receiver and sender positions using time-difference-of-arrival measurements. For this, we use a stratified, two-tiered approach. In the first step the problem is converted to a low-rank matrix estimation problem. We present new, efficient solvers for the minimal problems of this low-rank problem. These solvers are used in a hypothesis and test manner to efficiently remove outliers and find an initial estimate which is used for the subsequent step. Once a promising solution is obtained for a sufficiently large subset of the receivers and senders, the solution can be extended to the remaining receivers and senders. These steps are then combined with robust local optimization using the initial inlier set and the initial estimate as a starting point. The proposed system is verified on both real and synthetic data.

Keywords: TDOA, self-calibration, minimal problems, RANSAC

This work was partially supported by the strategic research projects ELLIIT and eSSENCE, the Swedish Foundation for Strategic Research project, Semantic Mapping and Visual Navigation for Smart Robots (grant no. RIT15-0038) and Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation. The authors gratefully acknowledge Lund University Humanities Lab.

Reprinted from *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, M. Larsson, G. Flood, M. Oskarsson, and K. Åström, Fast and Robust Stratified Self-Calibration Using Time-Difference-Of-Arrival Measurements, 4640-4644, Copyright 2021, with permission from IEEE.

1 Introduction

Precise localization of sender/receiver node positions using radio or sound signals is a key enabler in numerous applications such as microphone array calibration, speaker diarization, beam-forming, radio antenna array calibration, mapping and positioning [1]. There are several variants of this problem, for example (i) TOA, (ii) TDOA, (iii) COTDOA and (iv) UTDOA. The time-of-arrival (TOA) problem refers to the problem where measurements of absolute distances between senders and receivers can be obtained [2–4]. One example of this is when senders and receivers are jointly synchronized. The time-difference-of-arrival problem (TDOA) is the problem when the receivers are synchronized, whereas the senders are unsynchronized, or vice versa [5, 6]. The constant offset time-difference-of-arrival problem (COTDOA) is similar to the TDOA problem, but the unknown offset is constant [7]. Finally the UTDOA refers to the problem where neither senders nor receivers are synchronized [8].

In addition, the self-calibration problem becomes fundamentally different depending on the respective dimension of the affine hull of the senders and of the receivers. The senders and receivers can, for example, separately be confined to a line, a plane or span 3D space [9,10].

Considering each combination of calibration type (TOA, TDOA, COTDOA, UTDOA) with each combination of sender/receiver dimensionality within a common framework is a challenge. One strategy to understand and solve the self-calibration problem is to follow a two-tiered stratified approach [6, 7]. The first part of this approach is based on solving a relaxed version of the problem where, in the case of TDOA, COTDOA and UTDOA, any offsets are solved for. The second part consists of upgrading a relaxed solution to a solution to the original problem. This was recently studied in [11].

A different approach for performing robust TDOA self-calibration was proposed in [12], where subsets of the TDOA measurements were selected to calculate candidate TOA measurements. After poor candidates were discarded the median of the remaining ones was used to perform TOA self-calibration. In [13] the redundancy of the full set of TDOA measurements was exploited using low-rank approximation to perform denoising, fill in missing data and remove outliers. However, no system for self-calibration was proposed.

In this paper we follow the stratified approach, focusing on the TDOA case in 3D. Our contribution here is twofold. First, we improve on existing minimal solvers for finding the TDOA offsets [6], making them notably faster and reducing their memory requirements. Second, utilizing these improved solvers in efficient RANSAC [14] methods we produce a system for TDOA self-calibration that is robust to noise, missing data and

outliers[1]. We also verify the solvers and system using synthetic and real data.

2 Stratified Self-Calibration

The problem we address involves m receiver positions $\boldsymbol{r}_i \in \mathbb{R}^3$, $i = 1, \ldots, m$ and n sender positions $\boldsymbol{s}_j \in \mathbb{R}^3$, $j = 1, \ldots, n$. These could for example represent the microphone positions and locations of sound emissions, respectively. The arrival time of a sound j to receiver i is denoted t_{ij} and the time that sound j is emitted is denoted τ_j. Multiplying the travel time $t_{ij} - \tau_j$ with the speed v of the signal, we obtain the distance between sender and receiver

$$d_{ij} = z_{ij} - o_j = \|\boldsymbol{r}_i - \boldsymbol{s}_j\|, \tag{1}$$

where $z_{ij} = vt_{ij}$, $o_j = v\tau_j$ and $\|.\|$ denotes the ℓ^2-norm. Let $\hat{z}_{ij}$ be noisy measurements of z_{ij} that suffer from small approximately Gaussian noise, outliers with substantially larger errors and missing data. Estimating $\boldsymbol{r}_i$, $\boldsymbol{s}_j$ and o_j from $\hat{z}_{ij}$ is known as the TDOA node calibration problem.

We will use the notation $\theta_1 = \{\mathbf{R}, \mathbf{S}, \boldsymbol{o}\}$ for the unknown parameters, where $\boldsymbol{r}_i$ and $\boldsymbol{s}_j$ are columns of $\mathbf{R}$ and $\mathbf{S}$, respectively, and $\boldsymbol{o}$ is the vector of offsets. We will also let $\hat{\mathbf{Z}} \in \mathbb{R}^{m \times n}$ denote the matrix with entries $\hat{z}_{ij}$ and let $\mathcal{W}_{\text{in}}$ denote the index set where $(i, j) \in \mathcal{W}_{\text{in}}$ indicates that $\hat{z}_{ij}$ is not missing and is an inlier. Given the measurements $\hat{\mathbf{Z}}$ and an initial solution θ_1 the refinement of the estimate can be found by local optimization methods, e.g., Levenberg-Marquardt [15,16], by minimizing

$$f(\theta_1) = \sum_{(i,j) \in \mathcal{W}_{\text{in}}} L\left(\hat{z}_{ij} - (\|\mathrm{r}_i - \mathrm{s}_j\| + o_j)\right), \tag{2}$$

where $L(\cdot)$ is a loss function, e.g., the quadratic loss or the robust Huber loss [17].

The stratified approach is based on a relaxation of the problem, that exploits the fact that $\mathbf{D}^{\circ 2}$ has rank 5 [5], where $\mathbf{D}^{\circ 2} \in \mathbb{R}^{m \times n}$ is the matrix with entries $d_{ij}^2 = (z_{ij} - o_j)^2$. Further simplifications use the double compaction method [6]. The double compaction matrix $\mathbf{M} \in \mathbb{R}^{m \times n}$ is defined as the matrix with elements $M_{ij} = (z_{ij} - o_j)^2 - a_i - b_j$, where $\boldsymbol{a}$ and $\boldsymbol{b}$ are, apart from a scalar offset, affine combinations of the columns and rows of $\mathbf{D}^{\circ 2}$, respectively (see [11]). The matrix $\mathbf{M}$ can be shown to have rank 3, i.e., it can be expressed as $\mathbf{M} = \mathbf{U}^T\mathbf{V}$, where $\mathbf{U} \in \mathbb{R}^{3 \times m}$ and $\mathbf{V} \in \mathbb{R}^{3 \times n}$. The relaxed problem thus involves a set of parameters $\theta_2 = \{\mathbf{U}, \mathbf{V}, \boldsymbol{b}, \boldsymbol{a}, \boldsymbol{o}\}$. Here the constraints can be written

[1] MATLAB and C++ source code can be found at `https://github.com/martinkjlarsson/tdoa-self-calibration`.

Table 1: Execution times and elimination template sizes for our implementation of five minimal offset solvers.

Rank	m	n	# solutions	Exec. time	Template size
2	7	4	1	37 µs	14 × 15
2	5	6	5	810 µs	37 × 42
3	9	5	1	36 µs	29 × 30
3	7	6	5	1.4 ms	52 × 57
3	6	8	14	7.1 ms	320 × 334

as $z_{ij} = \sqrt{\boldsymbol{u}_i^T \boldsymbol{v}_j + a_i + b_j} + o_j$, where $\boldsymbol{u}_i$ and $\boldsymbol{v}_j$ denote columns i and j of $\mathbf{U}$ and $\mathbf{V}$, respectively. Refinement of the parameters can be done by local minimization of

$$f(\theta_2) = \sum_{(i,j)\in\mathcal{W}_{\text{in}}} L\left(\hat{z}_{ij} - \left(\sqrt{\boldsymbol{u}_i^T \boldsymbol{v}_j + a_i + b_j} + o_j\right)\right). \tag{3}$$

3 Minimal Solvers for the Offsets

The first step in the stratified approach is to estimate the unknown offsets $\boldsymbol{o}$. In [6] a technique for solving five minimal problems is presented (see Table 1). Throughout the paper we will use (mr/ns) to denote the problem or solver that requires m receivers and n senders. Two of the problems, (7r/4s) and (9r/5s), are linear, while the remaining three are nonlinear. In this section we propose improvements to the nonlinear ones, using automated tools from [18], making them significantly faster than in [6].

The constraint on $\mathbf{M}$ to be of rank 3 is equivalent to all minors of order 4 being zero. This results in a polynomial system in o_j, which can be solved using action matrix methods [19]. In the resulting solver all polynomial coefficients must be calculated from data, i.e., a large number of polynomials in z_{ij} must be evaluated. The worst solver in this regard is (6r/8s), where there are 5025 coefficients of degrees between four and eight in z_{ij}. To explicitly evaluate these one by one is time consuming and results in code that requires a lot of memory to compile if implemented in, e.g., C++.

However, using Laplace expansion the minors can be written using combinations of lower order minors and consequently, the polynomial system in $\boldsymbol{o}$ has many common subexpressions. This in turn results in common subexpressions in the coefficients which we eliminate to decrease the execution time and memory requirements of the solvers. Table 1 shows the execution time of our solvers implemented in MATLAB. These are

significantly faster than the original solvers in [6], where times in the order of 500 ms was reported.

4 Minimal Solvers in RANSAC

We propose the use of the fast minimal solvers in a hypothesize and test framework to obtain (i) an initial estimate of the offsets $\boldsymbol{o}$ and (ii) an initial inlier set.

We start by randomly picking one of the minimal cases with m' receivers and n' senders. We then randomly select m' rows and n' columns of $\hat{\mathbf{Z}}$ containing no missing data and solve for the corresponding offsets $\boldsymbol{o}'$. For each real solution $\boldsymbol{o}'$ we can find the corresponding $\boldsymbol{a}'$, $\boldsymbol{b}'$ and double compaction matrix $\mathbf{M}'$. From $\mathbf{M}'$ we can extract $\mathbf{U}'$ and $\mathbf{V}'$ using singular value decomposition (SVD).

A partial solution can then be extended by utilizing more columns of $\hat{\mathbf{Z}}$. For each remaining column j we pick 5 measurements randomly from the m' selected rows and solve for $\boldsymbol{v}_j$, b_j and o_j. However, we require there to be at least 6 measurements available, so that there are extra measurements to use for testing the extension and to classify as inliers or outliers according to the residuals in (3).

5 Systems

In this section we put the offset solvers into a robust system for solving TDOA. We use a two-tiered stratified approach as in [6] where we start by constructing a solution $\theta_2 = \{\mathbf{U}, \mathbf{V}, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{o}\}$ which is later upgraded to a solution $\theta_1 = \{\mathbf{R}, \mathbf{S}, \boldsymbol{o}\}$. Some of the components are described in detail below and the system as a whole is summarized in Algorithm 1. For comparison we also implemented a naïve system relying on random initialization, see Algorithm 2.

5.1 Initialization of θ_2

We start by initializing a solution θ_2 as described in Section 4. It is worthwhile finding a reasonable initial solution supported by many inliers, as a good initialization will speed up the remainder of the system.

Algorithm 1 Proposed system

1: Initialize solution θ_2 (Section 4).
2: Local nonlinear optimization over θ_2.
3: Extend rows and columns (Section 5.2).
4: Upgrade relaxed solution θ_2 to solution θ_1 (Section 5.3).
5: Reestimate receiver and sender positions (Section 5.4).
6: Local nonlinear optimization over θ_1 using robust norm.

Algorithm 2 Random initialization system

1: Initialize solution θ_1 randomly.
2: Local nonlinear optimization over θ_1.

5.2 Extending Solution in θ_2

The initialization will most likely yield only a partial solution in the sense that not all $\boldsymbol{u}_i$ and $\boldsymbol{v}_j$ are estimated, and that not all available measurements in $\hat{\mathbf{Z}}$ are used. The solution is extended with additional rows and columns using robust techniques as described in [20]. During this process it is useful to keep the errors down by occasionally refining the solution using local optimization. This has been shown to reduce failures (see e.g. [21, 22]).

5.3 Upgrade Solution in θ_2 to Solution in θ_1

A solution in θ_2 is upgraded to a solution in θ_1 using the minimal solvers presented in [11]. In RANSAC fashion a solver is randomly selected to find the upgrade parameters $\mathbf{L} \in \mathbb{R}^{3\times3}$ and $\boldsymbol{q} \in \mathbb{R}^3$. Receiver and sender positions are then given by the affine transformations $\mathbf{R} = \mathbf{L}^{-T}\mathbf{U}$ and $\mathbf{S} = \mathbf{L}(\mathbf{V} + \boldsymbol{q}\mathbf{1}^T)$, and inliers/outliers are classified according to the residuals in (2).

5.4 Reestimate Rows and Columns in θ_1

At this stage in the process, we can end up with receivers or senders that have not yet been estimated or that have ended up in incorrect locations, e.g., gotten stuck at a local minimum. To mitigate this, all receivers and senders are reestimated using trilateration and multilateration, respectively. If a new node position reduces the residuals in (2) it is kept, otherwise the old position is used.

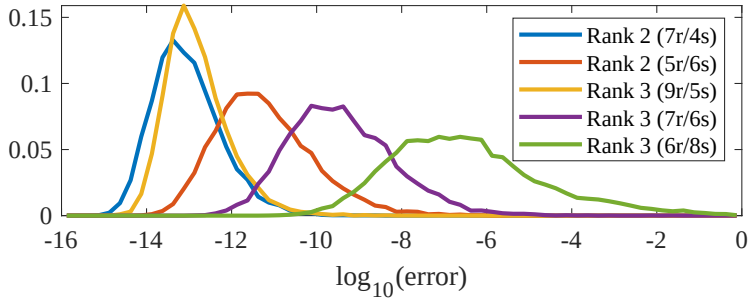


Figure 1: Histograms of the logarithm of the errors for our minimal solvers.

6 Experimental Validation

To test the numerical accuracy and robustness of our minimal solvers we generated 10,000 synthetic problem instances with known offsets. We then ran our solvers and compared the solutions with the ground truth solution. In Figure 1 the resulting histograms of the logarithm of the errors are shown. The linear solvers, (7r/4s) and (9r/5s), performed best and overall the solvers show better numerical stability as the rank and number of solutions decrease.

Note that while the linear solvers seem to be more numerically stable and faster (see Table 1), the other solvers are still useful in scenarios where we do not have a sufficient number of receivers. One could also imagine measurements that admit multiple possible solutions to the offsets, and in those cases the linear solvers will only give one of them.

To investigate the robustness of our systems with respect to outliers and missing data we generated 15 receivers $\mathbf{R}$ and 100 senders $\mathbf{S}$ with each coordinate drawn from $\mathcal{N}(0,1)$, and corresponding offset values $\boldsymbol{o} \in \mathcal{N}(0,1)$. These were used to acquire distance measurements $\hat{\mathbf{Z}}$ according to (1), with additive measurement noise $\epsilon_{ij} \in \mathcal{N}(0, 0.01)$. Additionally, a number of values were deleted from $\hat{\mathbf{Z}}$ to simulate missing data, and some were changed to uniformly distributed values $z_{ij} \in \mathcal{U}(-2, 6)$, to simulate gross outlier measurements.

We then solved for $\theta_1 = \{\mathbf{R}, \mathbf{S}, \boldsymbol{o}\}$ using (i) Algorithm 1 and the (9r/5s) solver, (ii) Algorithm 1 and the (7r/6s) solver, (iii) Algorithm 1 and the (6r/8s) solver, and (iv) Algorithm 2. This was done 100 times to get an estimate on how often the different systems converge. A solution counted as successful if the Euclidean distance between the ground truth receiver position and the corresponding estimated receiver position was at most 0.03 for any of the receivers.

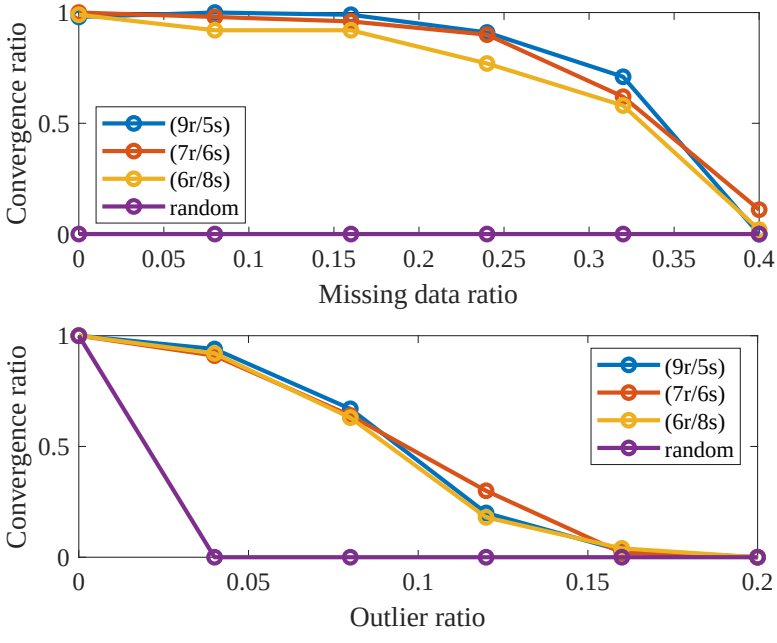


Figure 2: Convergence ratio with respect to ratio of missing data (top) and ratio of outliers (bottom) for four different systems.

Table 2: Results from seven real TDOA datasets including the RMSE in the estimated receiver positions.

Dataset	m	n	RMSE	Est. speed of sound
1	12	89	58 mm	347 m/s
2	12	106	50 mm	347 m/s
3	12	97	45 mm	348 m/s
4	12	105	52 mm	347 m/s
5	12	108	82 mm	348 m/s
6	12	131	64 mm	354 m/s
7	12	115	55 mm	350 m/s

The experiment above was conducted with a fixed outlier ratio of 1 % while the ratio of missing data was varied from 0–40 %. The results from this are shown in the top plot in Figure 2. We then kept the missing data ratio fixed at 1 % while varying the ratio of outliers between 0–20 %. These results can be seen in the bottom plot of Figure 2. It is clear that our systems, from Algorithm 1, outperforms Algorithm 2. The choice of minimal solver only has a small impact on the result, but overall, the (9r/5s) solver is a better choice for the type of data synthesized here considering it is significantly faster than the other two solvers (see Table 1).

We also evaluated our system using real data. The setup consisted of 12 omni-directional microphones (the T-bone MM-1) spanning a volume of $4.0 \times 4.6 \times 1.5$ meters. A speaker was moved through the setup while emitting sound. Ground truth positions for the microphones and speaker positions where found using a Qualisys motion capture system. Seven datasets were gathered in which a chirp sound was played with regular (dataset 1-5) or irregular (dataset 6-7) intervals. The arrival times were found using cross-correlation between the recordings and the original chirp. There was no missing data. The temperature in the room was measured to be 20.1 °C which indicates a speed of sound of $v = 343\,\mathrm{m/s}$. However, we choose to disregard this estimate and consider v, and thus the scale of the solution, unknown. Because of this, the estimated microphones were registered to the ground truth using a similarity transform. The scale component of the transform was then used to estimate the speed of sound. Table 2 shows the RMS errors in the estimated microphone positions and the estimated speed of sound for each dataset. The errors are overall low with a consistent estimate of v that is close to the estimate based of the room temperature. While the true outlier rate is unknown our system classified 14–20 % of the measurements as outliers. The estimated node positions for the fourth dataset are shown in Figure 3. As can be seen, the estimated speaker positions closely follow the ground truth.

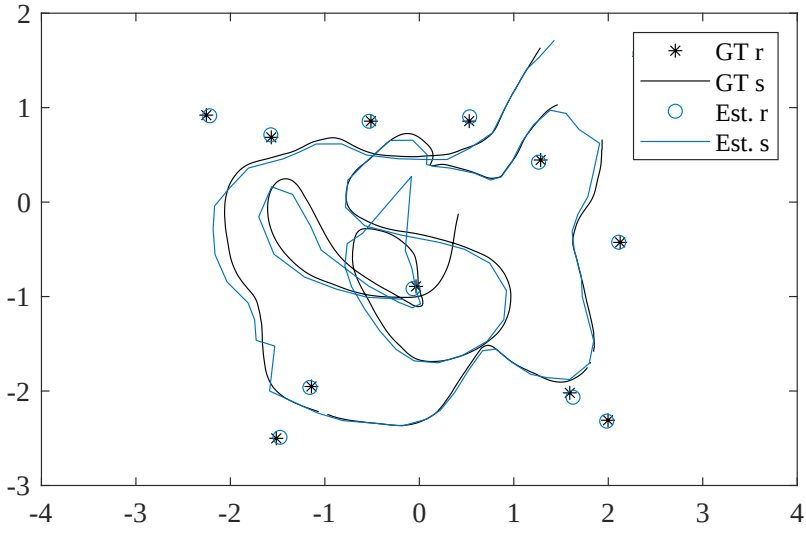


Figure 3: Top view of ground truth and estimated microphone/receiver and speaker/sender positions for the fourth dataset. The scale is in meters.

7 Conclusions

In this paper we have made several improvements to three minimal solvers for estimating offsets from time-difference-of-arrival data. The new solvers require less memory and some are two orders of magnitude faster than the state-of-the-art solvers. In the paper we also develop hypothesis and test algorithms that incorporate these new solvers and develop software systems that combine these with robust nonlinear estimation. The resulting components and systems have been tested on both synthetic and real data, where they demonstrate high quality solutions even in the presence of missing data and outliers. Consolidating further calibration types and dimensionality constraints in one coherent framework is future work.

References

[1] A. Plinge, F. Jacob, R. Haeb-Umbach, and G. A. Fink, "Acoustic microphone geometry calibration: An overview and experimental evaluation of state-of-the-art algorithms," *IEEE Signal Processing Magazine*, vol. 33, no. 4, pp. 14–29, 2016.

[2] M. Crocco, A. Del Bue, M. Bustreo, and V. Murino, "A closed form solution to the microphone position self-calibration problem," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2012.

[3] H. Stewénius, "Gröbner basis methods for minimal problems in computer vision," Ph.D. dissertation, Lund University, 2005.

[4] Y. Kuang, S. Burgess, A. Torstensson, and K. Åström, "A complete characterization and solution to the microphone position self-calibration problem," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2013.

[5] M. Pollefeys and D. Nister, "Direct computation of sound and microphone locations from time-difference-of-arrival data," in *International conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2008.

[6] Y. Kuang and K. Åström, "Stratified sensor network self-calibration from tdoa measurements," in *European Signal Processing Conference (EUSIPCO)*, 2013.

[7] K. Batstone, G. Flood, T. Beleyur, V. Larsson, H. Goerlitz, M. Oskarsson, and K. Åström, "Robust self-calibration of constant offset time-difference-of-arrival," in *International conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019.

[8] S. Burgess, Y. Kuang, and K. Åström, "Node localization in unsynchronized time of arrival sensor networks," in *International Conference on Pattern Recognition (ICPR)*, 2012.

[9] S. Burgess, Y. Kuang, and K. Åström, "Toa sensor network self-calibration for receiver and transmitter spaces with difference in dimension," *Signal Processing*, vol. 107, pp. 33–42, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0165168414002606

[10] E. Ask, Y. Kuang, and K. Åström, "A unifying approach to minimal problems in collinear and planar tdoa sensor network self-calibration," in *European Signal Processing Conference (EUSIPCO)*, 2014.

[11] M. Larsson, G. Flood, M. Oskarsson, and K. Åström, "Robust self-calibration of constant offset time-difference-of-arrival," in *International conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.

[12] T. K. Le and N. Ono, "Robust tdoa-based joint source and microphone localization in a reverberant environment using medians of acceptable recovered toas." Institute of Electrical and Electronics Engineers Inc., 10 2016.

[13] J. Velasco, D. Pizarro, J. Macias-Guarasa, and A. Asaei, "Tdoa matrices: Algebraic properties and their application to robust denoising with missing data," *IEEE Transactions on Signal Processing*, vol. 64, pp. 5242–5254, 10 2016.

[14] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–95, 1981.

[15] K. Levenberg, "A method for the solution of certain non-linear problems in least squares," *Quarterly of applied mathematics*, vol. 2, no. 2, pp. 164–168, 1944.

[16] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal of the society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963.

[17] P. J. Huber, "Robust estimation of a location parameter," in *Breakthroughs in statistics*. Springer, 1992, pp. 492–518.

[18] V. Larsson, K. Åström, and M. Oskarsson, "Efficient solvers for minimal problems by syzygy-based reduction," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2383–2392.

[19] D. Cox, J. Little, and D. O'Shea, *Using Algebraic Geometry*. Springer Verlag, 1998.

[20] K. Batstone, M. Oskarsson, and K. Åström, "Robust time-of-arrival self calibration with missing data and outliers," in *2016 24th European Signal Processing Conference (EUSIPCO)*, Aug 2016, pp. 2370–2374.

[21] C. Engels, H. Stewénius, and D. Nistér, "Bundle adjustment rules," *Photogrammetric computer vision*, vol. 2, no. 32, 2006.

[22] G. Klein and D. Murray, "Parallel tracking and mapping for small AR workspaces," in *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*. IEEE, 2007, pp. 225–234.

Paper IV

Extension of Time-Difference-of-Arrival Self Calibration Solutions Using Robust Multilateration

Kalle Åström[1], Martin Larsson[1,2], Gabrielle Flood[1], Magnus Oskarsson[1]

[1] *Centre for Mathematical Sciences, Lund University, Lund, Sweden*
[2] *Combain Mobile AB*

Abstract: Recent advances in robust self-calibration have made it possible to estimate microphone positions and at least partial sound source positions using ambient sound. However, there are limits on how well sound source paths can be recovered using state-of-the-art techniques. In this paper we develop and evaluate several techniques to extend partial and incomplete solutions. We present minimal solvers for sound source positioning using non-overlapping pairs of microphone positions and their respective time-difference measurements, and show how these new solvers can be used in a hypothesis and test setting. We also investigate techniques that exploit temporal smoothness of the sound source paths. The different techniques are evaluated on both real and synthetic data, and compared to several state-of-the-art techniques for time-difference-of-arrival multilateration.

Keywords: TDOA, multilateration, minimal problems, RANSAC, self-calibration

This work was partially supported by the strategic research projects ELLIIT and eSSENCE, the Swedish Foundation for Strategic Research project, Semantic Mapping and Visual Navigation for Smart Robots (grant no. RIT15-0038) and Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation. The authors gratefully acknowledge Lund University Humanities Lab.

Reprinted from *2021 29th European Signal Processing Conference (EUSIPCO)*, K. Åström, M. Larsson, G. Flood, and M. Oskarsson, Extension of Time-Difference-of-Arrival Self Calibration Solutions Using Robust Multilateration, 870-874, Copyright 2021, with permission from EURASIP.

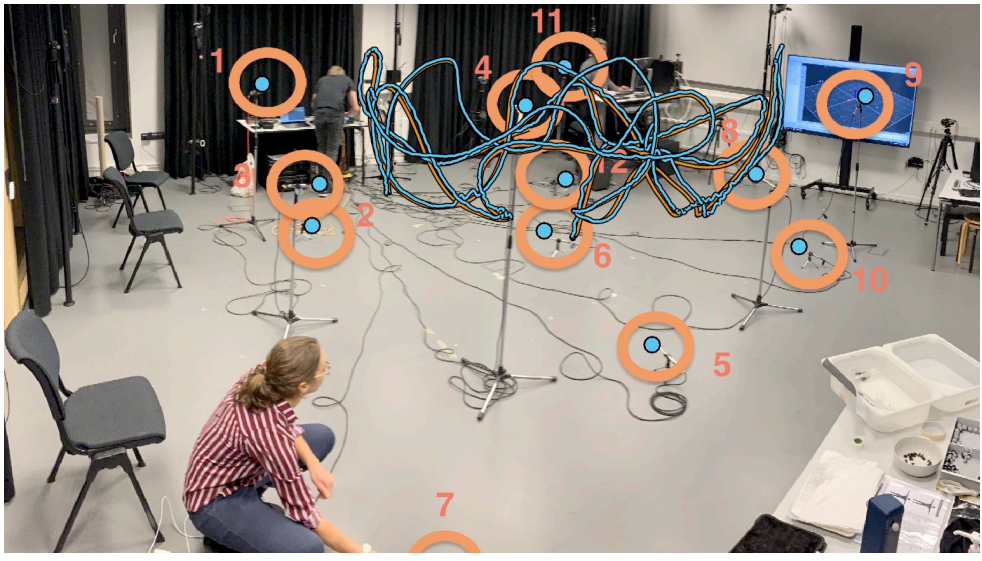


Figure 1: A setup consisting of 12 omni-directional microphones used to collect TDOA measurements from a moving sound source. In the figure is shown 3D reconstructions (light blue) and ground truth (orange) of sound source path.

1 Introduction

Precise localization of sender/receiver node positions using radio or sound signals is a key enabler in numerous applications such as microphone array calibration, speaker diarization, beamforming, radio antenna array calibration, mapping and positioning [1]. In this paper we study the problem of self-calibration of sender/receiver positions using time-difference-of-arrival (TDOA) measurements from a set of fixed and synchronised microphones. The problem is simpler if the sound source has distinct sound events, which are easy to detect [2], or if the sound profile is known [3]. Recent advances in robust parameter estimation has made it possible to solve such problems even for the relatively difficult scenario of unknown ambient sound [4–8]. In many cases it is possible to achieve at least partial estimates of sound source positions and microphone positions. However, these methods typically do not provide good estimates of sound source positions for all time instants, at least not for difficult situations.

In this paper we develop improved robust multilateration methods and show how such methods can improve on sender/receiver node position calibration systems. While focusing on the problem of multilateration, we envision that the proposed method works as a part of a larger self-calibration system, in order to increase robustness. For this reason, our experiments are focused on this scenario.

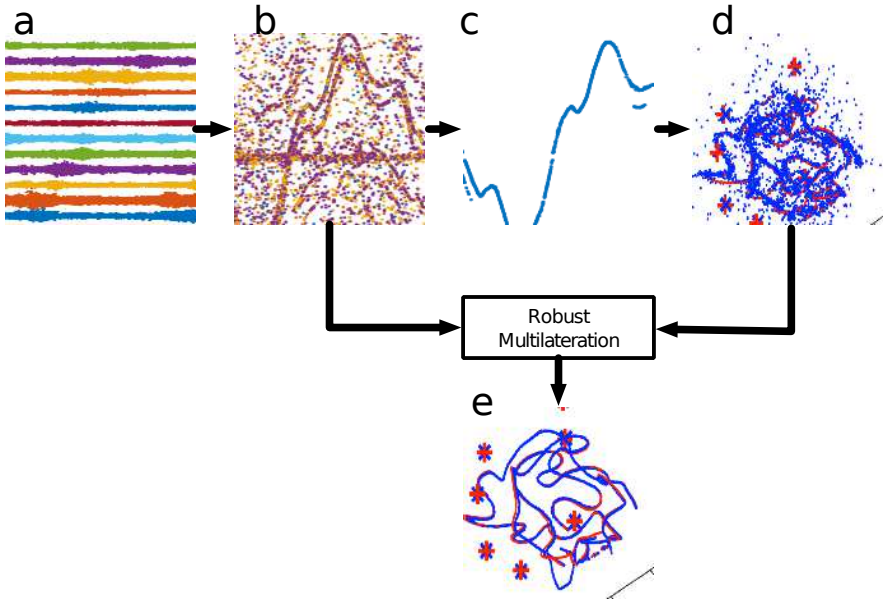


Figure 2: System overview: The input consists of a number of sound recordings (a). Using GCC-PHAT, a number of putative TDOA estimates for each time instant and for each pair of microphones are produced (b). Heuristics are used to prune these matches (c), which then are used to estimate microphone and source positions (d). In this paper we study methods for improved sound source localization using microphone positions from (d) and putative matches from (b). The aim is to achieve robust sound source localization (e).

The contributions of this paper are (i) new datasets for robust TDOA multilateration[1], (ii) a new fast solver for the minimal problem of TDOA multilateration, (iii) new methods for robust TDOA multilateration and (iv) evaluation of state-of-the-art methods for robust TDOA multilateration.

2 Structure from sound pipeline

For the solution of the structure from sound problem, we use the following structure, inspired by [5]. The input to the system is a number m of synchronised sound recordings (Figure 2a). For each pair (i, j) of recordings we use a detector to generate a set of putative time-difference-of-arrival measurements z for a number of time instants $t_k, k = 1, \ldots, n$ (Figure 2b). After a heuristic step for removing outliers (Figure 2c),

[1]We provide both dataset `https://vision.maths.lth.se/sfsdb/` and code `https://github.com/kalleastrom/StructureFromSound`.

the data is used as input to a system for robust structure from sound (Figure 2d). When successful, the system outputs all or a subset of the microphone positions, but often only a subset of the sound source positions. In this paper we study methods for extending an initial solution to additional sound source positions (Figure 2e). The idea is that the set of putative matches (Figure 2b), contains valuable information that could be better exploited using the sound source positions in (Figure 2d).

3 Multilateration Methods

Using sound to measure distances has been exploited for a long time, see for example [9]. Multilateration of sound source positions from a set of known microphones has been utilized, at least since World War I, to locate the source of artillery fire using sound waves [10].

For the multilateration problem, we assume that the microphone positions $(\boldsymbol{r}_1, \ldots, \boldsymbol{r}_m)$ are known. As an example, these microphone posiitons could have been estimated using a self-calibration system, e.g., [8]. At each time instant t_k we estimate time-differences τ_{ij} of the arrival of sound to the two microphones $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$. When multiplied with the speed of sound c, each such time-difference τ_{ij} gives a distance-difference estimate

$$z_{ij} = \tau_{ij} c \approx \|\boldsymbol{r}_i - \boldsymbol{s}\| - \|\boldsymbol{r}_j - \boldsymbol{s}\| + \epsilon, \tag{1}$$

where $\boldsymbol{s}$ is the unknown sound source position and $\|\cdot\|$ denotes the ℓ^2-norm. The noise ϵ is either an *inlier*, assumed to be normally distributed with a relatively small standard deviation, or an *outlier*, assumed to be drawn from a uniform distribution with a significantly larger standard deviation. Henceforth, we will use the term TDOA (time-difference-of-arrival) for the measurements z_{ij}, even though they actually represent distances and not time. Early algorithms were constructed for solving for $\boldsymbol{s}$ in (1), often assuming a planar geometry, and further assuming that the TDOA measurements are outlier free and without missing data. Such algorithms were often iterative and assumed that an initial guess of $\boldsymbol{s}$ was given.

In this paper we assume that we have a pool of hypotheses for the measurements z_{ij}. Each measurement is a collection of tuples $M = \begin{pmatrix} i & j & z_{ij} \end{pmatrix}$. The pool $\mathcal{P}$ consists of all of these putative measurements $\mathcal{P} = \{M_1, \ldots, M_N\}$, where several measurements could be to the same (i, j) combination.

In the experiments the measurements $\mathcal{P}$ were obtained by taking the top $K = 4$ peaks in the GCC-PHAT score [11], for each microphone pair (i, j). In [12], a system was proposed that uses a few top peaks in the GCC-PHAT score and tracks through time

using continuity constraint using the Viterbi algorithm. Another method was proposed in [5], where RANSAC together with continuity constraints was used to track peaks over time. Collecting distance-difference measurements in an $m \times m$ matrix

$$\boldsymbol{Z} = \begin{pmatrix} z_{11} & \cdots & z_{1m} \\ \vdots & \ddots & \vdots \\ z_{m1} & \cdots & z_{mm} \end{pmatrix} \tag{2}$$

we obtain a *TDOA matrix*. In the suggested computational pipeline, we can view the step in Figure 2b as having a TDOA matrix at each time instant. Again, note that for each element of these matrices there are several putative entries.

The true TDOA matrix is at most rank 2, [13], and can be written as

$$\boldsymbol{Z} = \boldsymbol{v}\mathbf{1}^T - \mathbf{1}\boldsymbol{v}^T, \tag{3}$$

where

$$\boldsymbol{v} = \begin{pmatrix} v_1 & \cdots & v_m \end{pmatrix}^T \tag{4}$$

is a vector of distance-differences, which will be called a *TDOA vector*. Adding a constant to $\boldsymbol{v}$ will not change the matrix $\boldsymbol{Z}$. Notice that each column of the TDOA matrix could be used as a TDOA vector, if they are outlier-free and without missing data. In the TDOA vector formulation, the measurement equation is

$$v_i = \|\boldsymbol{r}_i - \boldsymbol{s}\| + o, \tag{5}$$

where the unknown o can be interpreted as the unknown offset of the TDOA vector as discussed above. Alternatively, it can be interpreted as the unknown starting point of the sound. If the vector is obtained by measuring time-differences to a fixed microphone, e.g. $\boldsymbol{r}_1$, then we have $v_i = z_{i1} = \|\boldsymbol{r}_i - \boldsymbol{s}\| + o$ with $o = -\|\boldsymbol{r}_1 - \boldsymbol{s}\|$.

A common trick is to use four or more equations of type (5) to derive three or more equations of the form

$$(v_i - o)^2 - (v_1 - o)^2 = \boldsymbol{r}_i^T\boldsymbol{r}_i - \boldsymbol{r}_1^T\boldsymbol{r}_1 - (\boldsymbol{r}_i - \boldsymbol{r}_1)^T\boldsymbol{s}, \tag{6}$$

where two equations of type (5) are used for microphone i and 1. Note that the square terms $\boldsymbol{s}^T\boldsymbol{s}$ and o^2 disappear, and the constraints become linear in $\boldsymbol{s}$ and o.

In terms of the computational pipeline, we can view the step in Figure 2c as having a TDOA vector at each time instant, although, possibly with missing data and outliers.

Several closed-form solutions exist for multilateration using the TDOA vector formulation and the elimination in (6), e.g., [14–18]. Thus, all of these methods assume that

all time-differences are given to the same microphone. The minimal problem for the 3D case is to use four microphones and it has in general two solutions (counted with complex solutions and multiplicity of solutions). This can be seen as using three linear constraints of type (6) to reduce the four unknowns in $\boldsymbol{s}$ and o. This parameterizes the solution affinely with one parameter. Inserting this into the first equation

$$v_1 = \|\boldsymbol{r}_1 - \boldsymbol{s}\| + o,$$

gives a quadratic constraint, which has at most two solutions.

An initial solution can be refined iteratively by minimizing

$$f_{\boldsymbol{v}}(\boldsymbol{s}, o) = \sum_{i=1}^{m} L(v_i - (\|\boldsymbol{r}_i - \boldsymbol{s}\| + o)), \tag{7}$$

for the TDOA vector formulation or

$$f_{\boldsymbol{Z}}(\boldsymbol{s}) = \sum_{i=1}^{m} \sum_{j=1}^{m} L(z_{ij} - (\|\boldsymbol{r}_i - \boldsymbol{s}\| - \|\boldsymbol{r}_j - \boldsymbol{s}\|)), \tag{8}$$

for the TDOA matrix formulation. Here L is a loss function, e.g., the ℓ^2-loss $L(x) = x^2$. Other common choices are the ℓ^1-loss $L(x) = |x|$ or a robust version such as the Huber loss or truncated versions of ℓ^1or ℓ^2. We will also assume that L removes datapoints that are missing or known to be outliers (e.g. from our proposed bootstrapping in Section 5.1).

Building on previous results [19, 20], Velasco et al. used the redundancy of measurements in the TDOA matrix to perform denoising, detect outliers and fill in missing data [13]. The output from their approach is a TDOA vector, which can be used for trilateration using, e.g., [18]. Unlike the proposed method, [13] does not exploit the known microphone positions when denoising and allows for at most one TDOA measurement for each microphone pair. Additionally, the number of expected outliers is a nuisance parameter that must be specified prior to denoising.

4 Minimal Solvers

The closed-form solution for determining $\boldsymbol{s}$ using TDOA measurements, as presented in previous papers, e.g., [14–18], all assume (for the 3D case) that four elements of the TDOA vector are given, or that four elements of the TDOA matrix from the same row (or column) are given. The trick that is used in (6) does not work for the minimal case of any three measurements of the TDOA matrix. Here we introduce a fast and numerically

stable solver for this minimal case. We derive this for the general N-dimensional case although in practice we most often use it for 2D and 3D problems.

Let N be the dimension of the space, i.e., $\boldsymbol{s} \in \mathbb{R}^N$. Suitably rearranging and squaring (1) twice results in the quadratic constraint

$$\boldsymbol{s}^T \mathbf{A} \boldsymbol{s} + \boldsymbol{b}^T \boldsymbol{s} + c = 0, \tag{9}$$

where

$$\mathbf{A} = 4(\boldsymbol{r}_i - \boldsymbol{r}_j)(\boldsymbol{r}_i - \boldsymbol{r}_j)^T - 4z_{ij}^2 \mathbf{I}, \tag{10}$$

$$\boldsymbol{b} = 4z_{ij}^2(\boldsymbol{r}_i + \boldsymbol{r}_j) - 4(\boldsymbol{r}_i^T \boldsymbol{r}_i - \boldsymbol{r}_j^T \boldsymbol{r}_j)(\boldsymbol{r}_i - \boldsymbol{r}_j), \tag{11}$$

$$c = (\boldsymbol{r}_i^T \boldsymbol{r}_i - \boldsymbol{r}_j^T \boldsymbol{r}_j)^2 - 2(\boldsymbol{r}_i^T \boldsymbol{r}_i + \boldsymbol{r}_j^T \boldsymbol{r}_j) z_{ij}^2 + z_{ij}^4. \tag{12}$$

Constructing N quadratic combinations for different $(\boldsymbol{r}_i, \boldsymbol{r}_j, z_{ij})$ results in a polynomial system in $\boldsymbol{s}$. Using methods from algebraic geometry [21, p. 235] we conclude that there are at most four solutions for $N = 2$ and eight when $N = 3$. Some of the solutions may be complex and some may not satisfy (1) since we have lost the sign of z_{ij} due to the squaring. These solutions are however easily discarded. To produce a solver for the system we use an automatic solver generator [22]. Although there are dependencies between the polynomial coefficients, the problem does not admit smaller elimination template sizes (see [22]) than the case of independent coefficients (6×10 and 26×34 for $N = 2$ and $N = 3$, respectively).

An efficient method for solving three quadratics in three variables, corresponding to $N = 3$, was presented in [23]. There, the problem was reduced to a single univariate polynomial of degree eight whose real solutions were found using Sturm sequences [24]. We implemented their solver but found no clear improvement in execution time or numerical stability over our generated solver.

5 Robust Multilateration Algorithms

5.1 Proposed RANSAC scheme using minimal pairwise solver

We propose to use random sampling consensus (RANSAC) [25]. In the hypothesis and test loop we randomly choose three measurements from the pool of putative matches $\mathcal{P}$. From these three TDOA measurements we use the fast minimal solver to obtain hypotheses for the sound source position $\boldsymbol{s}$ and choose the one whose inlier set is maximal.

Inliers are measurements for which

$$\left| z - \|\boldsymbol{r}_i - \boldsymbol{s}\| - \|\boldsymbol{r}_j - \boldsymbol{s}\| \right| < T, \tag{13}$$

where T is a threshold chosen to distinguish between inliers and outliers. This initial estimate is then improved by optimizing truncated ℓ^2-loss according to (8).

5.2 Using smoothness over time

In the experiments we also consider using smoothness priors on the sound source path. The solution (including the microphone positions) is refined by local minimization of

$$f_{reg}(\boldsymbol{s}, \boldsymbol{r}) = \sum_{i=1}^{m} \sum_{j=1}^{m} \sum_{k=1}^{n} L\big(z_{ijk} - (\|\boldsymbol{r}_i - \boldsymbol{s}_k\| - \|\boldsymbol{r}_j - \boldsymbol{s}_k\|)\big) + \lambda \sum_{k=2}^{n-1} \|\boldsymbol{s}_{k-1} - 2\boldsymbol{s}_k + \boldsymbol{s}_{k+1}\|^2. \tag{14}$$

This requires a good initial estimate of the sound source path.

6 Experimental Validation

6.1 Real data

We collected one dataset consisting of seven recordings with different songs and different sound source motion. The setup consisted of 12 omni-directional microphones (the T-bone MM-1) spanning a volume of 4.0 × 4.6 × 1.5 meters (see Figure 1). Ground truth positions for the microphones and speaker positions were found using a Qualisys motion capture system. The microphones were all internally synchronized, but we assume that the time of sound emission from the speaker is unknown. For each recording a song was played as the speaker was moved around in the room and approximately one minute was recorded using a soundcard with sampling rate 96, 000 Hz. The temperature in the room was measured to be 20.1 °C which indicates a speed of sound of $c = 343$ m/s.

For each pair of microphones, the GCC-PHAT score [11] was calculated. We used a window of 2, 048 samples centered at every 1, 000:th sample points. The search width for the GCC-PHAT score was cropped to ±800 sample points. Thus we are able to find time-difference-of-arrival measurements corresponding to ±2.85 m distance-difference to microphone pairs. For each time instant and each pair of microphones we selected at the four strongest local maxima in the GCC-PHAT score, resulting in a pool of putative measurements. In Figure 2.b these are shown for microphone pair 6 och 8.

Thus, for each recording and for each time window we had time-difference-of-arrival measurements and ground truth microphone and sound source positions. In total there were 46,066 such examples to validate the algorithms on.

6.2 Simulated/real data

The real dataset is quite challenging. It contains outliers, missing data and multiple hypotheses. In order to understand the behaviour of the algorithms we also constructed simulated data. This was done using the ground truth positions of the microphones and the sound source for the seven datasets above. In this way we could make datasets that were similar in geometry, but for which there was less noise, less outliers and/or less missing data. Together with the real recordings, this resulted in the following four datasets:

(a) Simulated TDOA measurements. One hypothesis. Gaussian noise with $\sigma = 2$ sample points. No missing data. No outliers. Ground truth microphone positions.

(b) Simulated TDOA measurements. One hypothesis. Gaussian noise with $\sigma = 2$ sample points. Missing data: 20%. Outliers: 20%. Ground truth microphone positions.

(c) Real TDOA measurements. Four hypotheses. Inlier noise estimated to have $\sigma \approx 5$ sample points. Outliers: $\approx$ 86%. Ground truth microphone positions.

(d) Real TDOA measurements. Four hypotheses. Inlier noise estimated to have $\sigma \approx 5$ sample points. Outliers: $\approx$ 86%. Estimated microphone positions from a state-of-the-art self-calibration system, [8].

6.3 Evaluation of the multilateration methods

We first evaluate methods that only use one individual time instant. We tested two state-of-the-art routines. For both methods we initially, from the pool of putative matches $\mathscr{P}$, generate the TDOA matrix $\mathbf{Z}$ by selecting the measurement for each microphone pair for which the GCC-PHAT score is the strongest. For (i) Chan and Ho [18], we then use one of the microphones (no 6 in our experiment) to calculate the TDOA vector $\boldsymbol{v}$ from $\mathbf{Z}$. The 6th microphone was considered to be best for this purpose since it was in the centre of the room. Finally we estimate the sound source position $\boldsymbol{s}$ using microphone positions and the TDOA vector according to [18]. For the second method (ii) Velasco et

al., we use [13] to robustly estimate the TDOA vector $\boldsymbol{v}$ from $\mathbf{Z}$. Finally, we estimate the sound source position $\boldsymbol{s}$ using microphone positions and the TDOA vector according to [18] as suggested in [13]. We also compared our method to four search based methods. These were (iii) ℓ^2-optimization from a random starting point, (iv) ℓ^1-optimization from a random starting point, (v) truncated ℓ^1-optimization from a random starting point and (vi) truncated ℓ^1-optimization from ten random starting points, choosing the solution with the lowest truncated ℓ^1-loss, as well as the proposed algorithm based on (vii) RANSAC loop to select starting point followed by truncated ℓ^2-optimization.

For each scenario above we calculated the percentage of times the estimated sound source came within 15 cm of the ground truth position. The results are shown in Figure 3. Notice that most methods work well for the outlier free dataset (a), except the truncated ℓ^1-loss optimization with one single random starting point. This shows that finding a good starting point is critical for robust loss optimization. With more outliers (dataset (b)), we see that Chan and Ho and several other methods struggle to find a good solution. For the real data problem with ground truth microphone positions (dataset (c)) the proposed method clearly outperforms the other methods. The final (dataset (d)), is even more challenging, but the overall trend is the same.

6.4 Applying motion priors

We used the result from the different multilateration methods to optimize over the whole sound recording using the motion prior as described in Section 5.2. This optimization needs a fairly good initial estimate in order to converge to the global optimum. The result is also shown in Figure 3. In Table 1 we show the results for dataset (d), with a breakdown to the individual seven recordings in the dataset. As can be seen in the table, there are four songs for which no methods work well. This is clearly a result of having a poor estimate of the microphone positions, since the result from dataset (c) works significantly better, indicating a need for further research. Finally we visualize how the suggested improvements affect the reconstructed 3D path. In Figure 4 we show the improvement to the 3D reconstruction with the proposed system for recording nr 6. It is clear that the proposed improvements (Figure 4-right) reduce the noise and improves the estimation of the sound source path as compared to the current state-of-the-art self-calibration system (Figure 4-left).

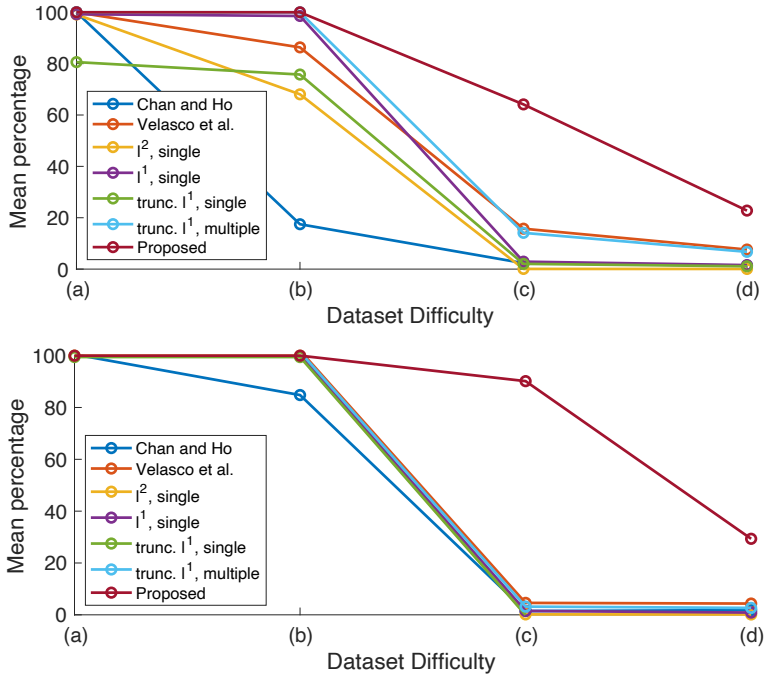


Figure 3: Performance aggregated over all datasets, as a function of varying difficulty in experimental setup, without (left) and with (right) temporal smoothing. The performance is measured by computing how often the source position was estimated within 15 cm from the ground truth position.

Table 1: The results for the seven different recordings from dataset (d) measured as the percentage of times the sound source position was estimated within 15 cm of the ground truth position.

Recording	1	2	3	4	5	6	7
Chan and Ho	0	0	0	1	0	4	4
Velasco et al.	0	0	0	3	0	23	28
ℓ^2, single	0	0	0	0	0	0	0
ℓ^1, single	0	0	0	0	0	4	6
trunc. ℓ^1, single	0	0	0	0	0	3	4
trunc. ℓ^1, multiple	0	0	0	1	0	20	25
Proposed	0	0	0	10	0	70	78

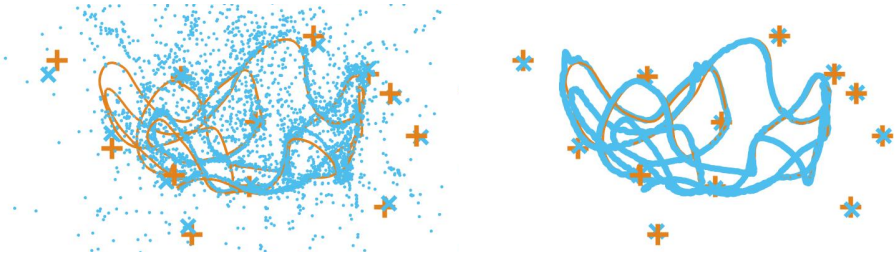


Figure 4: 3D reconstruction (blue) and ground truth (orange) of sound source path from calibration system (left) and from the proposed method (right).

7 Conclusions

In this paper we have made several improvements on robust multilateration using TDOA matrices with multiple hypotheses for each entry. We have developed a fast and efficient solver for the minimal problem of disjoint pairwise TDOA measurements. We have combined this solver with RANSAC algorithms and robust nonlinear estimation and obtained better results than state-of-the-art algorithms for robust multilateration. The resulting system has been tested on both synthetic and real data, producing high quality solutions even in the presence of missing data and high amount of outliers.

References

[1] A. Plinge, F. Jacob, R. Haeb-Umbach, and G. A. Fink, "Acoustic microphone geometry calibration: An overview and experimental evaluation of state-of-the-art algorithms," *IEEE Signal Processing Magazine*, vol. 33, no. 4, pp. 14–29, 2016.

[2] T. Janson, C. Schindelhauer, and J. Wendeberg, "Self-localization based on ambient signals," in *Proceedings of the 6th international conference on Algorithms for Sensor systems, Wireless Adhoc Networks, and Autonomous Mobile Entities*. Springer-Verlag, 2010, pp. 176–188.

[3] K. Batstone, G. Flood, T. Beleyur, V. Larsson, H. R. Goerlitz, M. Oskarsson, and K. Åström, "Robust self-calibration of constant offset time-difference-of-arrival," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 4410–4414.

[4] P. Pertila, M. Mieskolainen, and M. Hamalainen, "Passive self-localization of microphones using ambient sounds," in *Proc. European Signal Processing Conference (EUSIPCO)*. IEEE, 2012.

[5] Z. Simayijiang, S. Segerblom Rex, Y. Kuang, F. Andersson, and K. Åström, "An automatic system for acoustic microphone geometry calibration based on minimal solvers," *Arxiv*, 10 2016.

[6] T. K. Le and N. Ono, "Robust TDOA-based joint source and microphone localization in a reverberant environment using medians of acceptable recovered TOAS," in *2016 International Workshop on Acoustic Signal Enhancement, IWAENC 2016*. Institute of Electrical and Electronics Engineers Inc., 10 2016.

[7] M. Larsson, G. Flood, M. Oskarsson, and K. Åström, "Upgrade methods for stratified sensor network self-calibration," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 4851–4855.

[8] M. Larsson, G. Flood, M. Oskarsson, and K. Åström, "Fast and robust stratified self-calibration using time-difference-of-arrival measurements," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. IEEE, 2021.

[9] J. Meldercreutz, "Om längders mätning genom dåns tilhielp," *Vetenskapsakademiens Handlingar*, vol. 2, pp. 73–77, 1741.

[10] A. P. Bogatskiy, F. S. Kuznetsov, and A. F. Shapovalov, *Slovar' raketnyh i artillerijskih terminov*. Voyennoye, Moscow, 1968.

[11] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 24, no. 4, pp. 320 – 327, aug 1976.

[12] X. Anguera, C. Wooters, and J. Hernando, "Acoustic beamforming for speaker diarization of meetings," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, pp. 2011–2022, 9 2007.

[13] J. Velasco, D. Pizarro, J. Macias-Guarasa, and A. Asaei, "TDOA matrices: Algebraic properties and their application to robust denoising with missing data," *IEEE Transactions on Signal Processing*, vol. 64, pp. 5242–5254, 10 2016.

[14] W. Hahn and S. Tretter, "Optimum processing for delay-vector estimation in passive signal arrays," *IEEE Transactions on Information Theory*, vol. 19, no. 5, pp. 608–614, 1973.

[15] W. R. Hahn, "Optimum signal processing for passive sonar range and bearing estimation," *The Journal of the Acoustical Society of America*, vol. 58, no. 1, pp. 201–207, 1975.

[16] S. Bancroft, "An algebraic solution of the GPS equations," *IEEE transactions on Aerospace and Electronic Systems*, vol. 21, no. 7, pp. 56–59, 1985.

[17] B. T. Fang, "Simple solutions for hyperbolic and related position fixes," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 26, pp. 748–753, 1990.

[18] Y. T. Chan and K. C. Ho, "A simple and efficient estimator for hyperbolic location," *IEEE Transactions on Signal Processing*, vol. 42, pp. 1905–1915, 1994.

[19] H. C. So, Y. T. Chan, and F. K. W. Chan, "Closed-form formulae for time-difference-of-arrival estimation," *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2614–2620, 2008.

[20] R. Schmidt, "Least squares range difference location," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, no. 1, pp. 234–242, 1996.

[21] D. Cox, J. Little, and D. O'Shea, *Ideals, Varieties, and Algorithms*. Springer Verlag, 2007.

[22] V. Larsson, K. Åström, and M. Oskarsson, "Efficient solvers for minimal problems by syzygy-based reduction," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2383–2392.

[23] Z. Kukelova, J. Heller, and A. Fitzgibbon, "Efficient intersection of three quadrics and applications in computer vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1799–1808.

[24] C. F. Sturm, "Résolution des équations algébriques," *Bulletin de Férussac*, vol. 11, pp. 419–425, 1829.

[25] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–95, 1981.

Paper V

Sensor Node Calibration in Presence of a Dominant Reflective Plane

Erik Tegler[I], Martin Larsson[I,2], Magnus Oskarsson[I], Kalle Åström[I]

[1] *Centre for Mathematical Sciences, Lund University, Lund, Sweden*
[2] *Combain Mobile AB*

Abstract: Recent advances in simultaneous estimation of both receiver and sender positions in ad-hoc sensor networks have made it possible to automatically calibrate node positions – a prerequisite for many applications. In man-made environments there are often large planar reflective surfaces that give significant reverberations. In this paper, we study geometric problems of receiver-sender node calibration in the presence of such reflective planes. We establish a rank-1 factorization problem that can be used to simplify the estimation. We also show how to estimate offsets, in the Time difference of arrival case, using only the rank constraint. Finally, we present a new solver for the minimal cases of sender-receiver position estimation. These contributions result in a powerful stratified approach for the node calibration problem, given a reflective plane. The methods are verified with both synthetic and real data.

Keywords: TDOA, TOA, reverberations, minimal problems, self-calibration

This work was partially supported by the Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation, the ADACORSA project with funding from ECSEL JU in the H2020 Framework Programme (H2020/2014-2020) and National Authorities, under GA 876019, and the strategic research project ELLIIT. The authors gratefully acknowledge Lund University Humanities Lab.

Reprinted from *2022 30th European Signal Processing Conference (EUSIPCO)*, E. Tegler, M. Larsson, M. Oskarsson, and K. Åström, Sensor Node Calibration in Presence of a Dominant Reflective Plane, Copyright 2022, with permission from EURASIP.

1 Introduction

Accurate receiver-sender node positions are a key prerequisite for many applications such as microphone array calibration, radio antenna array calibration, mapping and positioning [1]. If all senders and receivers are synchronized, it is possible to obtain absolute distance measurements between senders and receivers. These measurements can be used for self-calibration and such problems (time of arrival problems, TOA) have been studied in a large body of work [2–10]. A variant of the TOA-problem is time difference of arrival (TDOA), where the receivers are synchronized and the senders are unsynchronized [11–13].

Large planar surfaces that act as acoustic or radio mirrors exist in both natural and man-made environments. In such cases, the received signal contains both the part from the direct path as well as parts that have been reflected against surfaces. This has been utilized for GNSS altimetry [14], estimating the shape of a room [15,16], and has the potential to be used in receiver-sender node position calibration [17,18]. In this paper, we study how such reverberations can be exploited. In particular, we study the case of a dominant unknown plane, e.g., the floor plane. In this case, for each receiver there are two detections, the direct and the indirect one reflected in the floor. We assume that these detections are correctly identified, although, in general, finding which surfaces a particular echo has bounced of is a problem in itself known as echo labeling [16,19]. We study how the geometry of this situation can be used, study minimal cases of reconstruction and use the new solvers for robust structure from motion estimation[1] . This leads to a powerful stratified formulation that separates the problem into TDOA offset estimation, height estimation and planar position estimation.

2 System Overview and Contributions

The general problem we address involves m receiver positions $\boldsymbol{R}_i \in \mathbb{R}^3$, $i = 1, \ldots, m$ and n sender positions $\boldsymbol{S}_j \in \mathbb{R}^3$, $j = 1, \ldots, n$. These could for example represent the microphone positions and locations of sound emissions, respectively. The arrival time of a signal sent from sender j to receiver i is denoted t_{ij}, and the time that it is emitted is denoted τ_j. Multiplying the travel time $t_{ij} - \tau_j$ with the speed v of the signal, we obtain the distance between sender and receiver

$$D_{ij} = Z_{ij} - o_j = \|\boldsymbol{R}_i - \boldsymbol{S}_j\|, \tag{1}$$

[1]Code: `https://github.com/Etomer/Reflective-Self-Calibration`

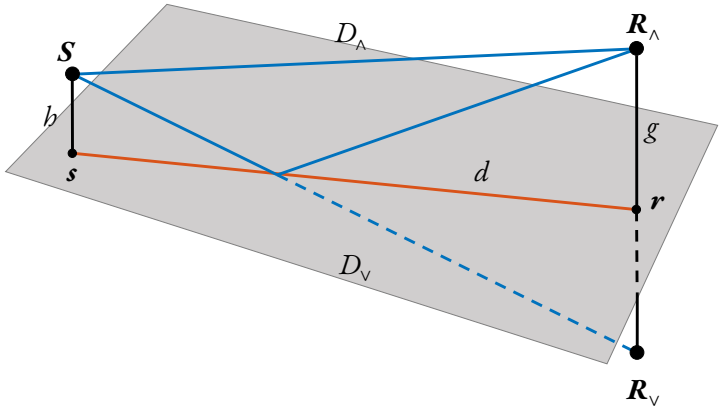


Figure 1: Schematic of the geometry for one sender $\boldsymbol{S}$ and a mirror pair of receivers $(\boldsymbol{R}_\wedge, \boldsymbol{R}_\vee)$. Also shown is the direct distance $D_\wedge$ and the reflected distance $D_\vee$. The sender and the receivers project to $\boldsymbol{s}$ and $\boldsymbol{r}$ respectively in the unknown mirror plane, with corresponding distance d.

where $Z_{ij} = vt_{ij}$, $o_j = v\tau_j$ and $\|.\|$ denotes the ℓ^2-norm. The speed v is throughout the paper assumed to be known and constant. Let Z_{ij} be noisy measurements that typically suffer from small approximately Gaussian noise, outliers with substantially larger errors and missing data. Estimating $\boldsymbol{R}_i$, $\boldsymbol{S}_j$ and o_j from Z_{ij} is known as the TDOA node calibration problem. If the offsets o_j are assumed to be known, we have the corresponding TOA node calibration problem. When building systems to solve such problems robustly, often a number of key system components need to be developed. Some of these are standard components, but some need to be specifically designed if we have special setups of the geometry. In this paper, we address such a specific case, namely when we know that there is a dominant reflective plane present in the scene. This gives a number of system benefits, but also puts a number of constraints on the system. In this case, we assume that we measure both a direct distance $D_\wedge$ and a reflected distance $D_\vee$ (or $Z_\wedge$ and $Z_\vee$ for the TDOA case). We can model the reflections using the true receivers $\boldsymbol{R}_\wedge$, and mirrored receivers $\boldsymbol{R}_\vee$ (see Fig. 1), so that

$$D_{\wedge ij} = Z_{\wedge ij} - o_j = \|\boldsymbol{R}_{\wedge i} - \boldsymbol{S}_j\|, \tag{2}$$

$$D_{\vee ij} = Z_{\vee ij} - o_j = \|\boldsymbol{R}_{\vee i} - \boldsymbol{S}_j\|, \tag{3}$$

defines our sensor node calibration problem.

In Algorithm 1, an overview of our proposed stratified approach for solving this node

calibration problem is shown. The specific components that we have developed, and that also make up the main contribution of our paper, are shown in bold face.

Algorithm 1 Proposed System (main contributions in bold)

Require: TOA or TDOA meaurements between senders and receivers with unknown positions in 3D

1: **Using the assumption of an (unknown) reflective plane, separate the problem into a rank-1 problem and a planar estimation problem (Section 3)**
2: **If we have a TDOA problem, use the rank-1 constraint to solve for the unknown time-offsets (Section 4)**
3: Solve the rank-1 problem in a robust way (allowing for outliers and missing data). We use a RANSAC [20] approach, giving the unknown heights of senders and receivers up to a global unknown parameter. (Section 5)
4: **Solve for the unknown global parameter and the unknown planar positions of the receivers and senders in a robust way using novel minimal solvers. (Section 6)**
5: Use non-linear refinement of all unknowns, e.g., by using gradient descent or Levenberg-Marquardt.

3 Mirror Geometry

The first thing we must consider is that we have a Euclidean ambiguity in our solution. This means that for a given solution, i.e., the position of the dominant reflective plane, senders and receivers, all Euclidean transformations of the solution is also a valid solution to the problem. In order remove this ambiguity, we fix the six degrees of freedoms by specifying our coordinate system. We do this by choosing the reflective plane as the z-plane, placing the first receiver on the z-axis and the second receiver in the yz-plane with positive x-coordinate.

Denoting the z-coordinates (heights) of the receivers g_i and senders h_j, and denoting the horizontal distance between $\boldsymbol{R}_i$ and $\boldsymbol{S}_j$ as d_{ij} (see Fig. 1), we get

$$D^2_{\wedge ij} = d^2_{ij} + (g_i - h_j)^2 = d^2_{ij} + g_i^2 + h_j^2 - 2g_i h_j, \tag{4}$$

$$D^2_{\vee ij} = d^2_{ij} + (g_i + h_j)^2 = d^2_{ij} + g_i^2 + h_j^2 + 2g_i h_j. \tag{5}$$

From these equations we can derive

$$D_{\Delta ij} \equiv \frac{D^2_{\vee ij} - D^2_{\wedge ij}}{4} = g_i h_j, \tag{6}$$

$$D_{\Sigma ij} \equiv \frac{D^2_{\vee ij} + D^2_{\wedge ij}}{2} = d^2_{ij} + g^2_i + h^2_j. \tag{7}$$

The first type of equation, (6), only involves the heights, g_i and h_j, and not the horizontal distance d_{ij}. Grouping together measurements from several sender-receiver pairs we get

$$D_\Delta = \begin{pmatrix} g_1 h_1 & \cdots & g_1 h_n \\ \vdots & \ddots & \vdots \\ g_m h_1 & \cdots & g_m h_n \end{pmatrix} = \begin{pmatrix} g_1 \\ \vdots \\ g_m \end{pmatrix} \begin{pmatrix} h_1 & \cdots & h_n \end{pmatrix}. \tag{8}$$

Estimating the heights g_i and h_j has now turned into a rank-1 matrix factorization problem. We will discuss solution strategies for this problem in Section 5.

The second type of equation (7) can be used to calculate the horisontal distances d_{ij} between the projections on the mirror plane of the receivers and senders,

$$d^2_{ij} = D_{\Sigma ij} - g^2_i - h^2_j. \tag{9}$$

This almost leads to an ordinary TOA-problem in one dimension less, i.e., for the unknown projected receiver and senders positions $\boldsymbol{r}_i$ and $\boldsymbol{s}_j$ in the plane (see Fig. 1), we have

$$d^2_{ij} = \|\boldsymbol{r}_i - \boldsymbol{s}_j\|^2, \tag{10}$$

where d^2_{ij} depends on the estimates of the heights. This dependance leads to a slightly modified TOA-problem, which is discussed in Section 6.

The following sections are concentrated on finding robust initial solutions to the calibration problem. This is typically followed by nonlinear optimization over all inlier data and parameters in a least-squares sense, i.e., we minimize a cost such as

$$\sum_{ij} L(D_{\wedge ij} - ||\boldsymbol{R}_{\wedge i} - \boldsymbol{S}_j||_2) + L(D_{\vee ij} - ||\boldsymbol{R}_{\vee i} - \boldsymbol{S}_j||_2), \tag{11}$$

for a robust loss function L, using some gradient descent method, e.g., Levenberg–Marquardt.

4 Offset Estimation

In this section, we will show how the rank constraint on D_Δ can be used to solve for the offsets o_j present when considering the TDOA-problem. As a reminder, the measurements are given by $Z_{\wedge ij}$ and $Z_{\vee ij}$ and relate to the distances according to

$$D_{\wedge ij} = Z_{\wedge ij} - o_j, \quad D_{\vee ij} = Z_{\vee ij} - o_j. \tag{12}$$

By insertion in (6), we observe that D_Δ is linear in o_j.

$$D_{\Delta ij} = \frac{Z_{\vee ij}^2 - Z_{\wedge ij}^2}{4} - \frac{Z_{\vee ij} - Z_{\wedge ij}}{2} o_j \tag{13}$$

The rank-1 constraint on D_Δ implies that all 2×2-minors vanish. Each minor will be a quadratic polynomial containing the monomials $\{o_{j_1} o_{j_2}, o_{j_1}, o_{j_2}, 1\}$ for some indices $j_1 \neq j_2, j_1, j_2 \in \{1, \dots, n\}$. The polynomial system formed in this way can be written $A\boldsymbol{v} = \boldsymbol{b}$, where A and $\boldsymbol{b}$ only depend on the data $(Z_{\wedge ij}, Z_{\vee ij})$ and $\boldsymbol{v}$ collects all non-constant monomials of the minors. Provided $m \geq 3$ and $n \geq 2$, the linear system is well-defined and $\boldsymbol{v}$ can be solved for. The offsets o_j are then easily extracted as the linear monomials in $\boldsymbol{v}$. Note that this method only utilizes the rank constraint on D_Δ, works independently of the dimension of the space and turns the TDOA problem into a TOA problem. How to solve the TOA problem is the topic of the next two sections.

5 Height Estimation

In Section 3, we saw that the full TOA self-calibration problem, with a mirror plane, decomposes into two separate problems. The problem of estimating the unknown heights turns into a low rank matrix factorization problem (8). Given a solution to this problem, it is clear that the rank-1 factorization of our data will only be determined up to an unknown parameter $\lambda \neq 0$, i.e., $D_\Delta = \hat{\boldsymbol{g}}\hat{\boldsymbol{h}}^T$, where $\boldsymbol{g} = \lambda\hat{\boldsymbol{g}}$ and $\boldsymbol{h} = \frac{1}{\lambda}\hat{\boldsymbol{h}}$.

If we have no missing data and no noise in our measurement, it is easy to find a solution to the factorization problem, simply by choosing $\hat{\boldsymbol{g}}$ as the first column of D_Δ and $\hat{\boldsymbol{h}}$ as the first row of D_Δ divided by $\hat{g}_1$. Consequently, with m receivers and n senders we only use $m + n - 1$ of the mn available equations. There are hence $(m-1)(n-1)$ constraints (invariants) that the noiseless realization should fulfill. In general, we will have noise, gross outliers and missing data in our measurement matrix. It is well known that the least-squares estimate is given by truncating the singular value decomposition of D_Δ to rank one [21]. However, if we have gross outliers this is not the best estimate, and if we

have missing data we cannot even compute the singular value decomposition. There has been much previous work on low-rank matrix factorization [22–25].

In this case, we can solve the factorization in an easier way since D_Δ only has rank one and for most of these problems $n \gg m$. We find the solution by fixing $\hat{g}_1 = 1$ and then solve for $\hat{g}_i$ using a RANSAC-voting scheme with vote j computed as $D_{\Delta ij}/D_{\Delta 1j}$. We then solve for each $\hat{h}_j$ by using a RANSAC-voting scheme, where vote i is given by $D_{\Delta ij}/\hat{g}_i$. We then decide which entries in D_Δ are inliers by checking which entries in $|\hat{\boldsymbol{g}}\hat{\boldsymbol{h}}^T - D_\Delta|$ are less than some chosen tolerance.

6 Planar Position Estimation

We will now turn our attention to the problem of estimating projected planar positions of the receivers and senders in the mirror plane, given that we have estimates of the heights. In the previous section, we saw that there were $(m-1)(n-1)$ invariants in the data, that are always fulfilled for noiseless data. This means that the number of excess constraints $\mathscr{E}$ is

$$\mathscr{E} = 2mn - (3m + 3n - 3) - (m-1)(n-1) \tag{14}$$

$$= mn - 2n - 2m + 2. \tag{15}$$

Setting $\mathscr{E} = 0$ gives the two minimal cases $(m, n) = (3, 4)$ and $(m, n) = (4, 3)$, which are the minimal amount of data that is required to solve the full TOA-problem. Note that, in these cases, the heights are slightly overdetermined when there is noise in the measurements. From (9) and (10) we get

$$\|\boldsymbol{r}_i - \boldsymbol{s}_j\|^2 = D_{\Sigma ij} - \lambda^2 \hat{g}_i^2 - \frac{1}{\lambda^2}\hat{h}_j^2, \tag{16}$$

where $\boldsymbol{r}_i$, $\boldsymbol{s}_j$ and λ are the unknown parameters. The scale λ is what makes (16) different from a standard planar TOA self-calibration problem, for which the minimal case is $(m, n) = (3, 3)$) [6, 7]. Using algebraic tools, it could be possible to eliminate the receiver and sender positions from (16), resulting in equations in only λ. This would enable a complete separation of the height estimation in the previous section and the planar position estimation treated here. However, we have found this elimination to be intractable[2]. Instead, we will eliminate only the senders and produce a solver for the receivers in conjunction with λ.

[2]The approach is nevertheless possible for the 2D equivalent of the 3D mirroring problem considered here. Then $(m, n) = (2, 2)$ and the constraint becomes a single quartic polynomial in λ^2.

Our approach for solving the planar TOA-problem together with λ is to formulate the problem as a polynomial equation system and then use an existing automatic solver generator [26]. The generated solver consists of a linear system (the so-called elimination template), and an eigendecomposition of the same size as the number of solutions to the problem.

To start, let $(m, n) = (3, 4)$, and fix the coordinate system as described in Section 3, i.e., let $\boldsymbol{r}_1 = \mathbf{0}$. We can then construct the linear systems $A\boldsymbol{s}_j = \boldsymbol{b}_j$, where

$$A = \begin{bmatrix} -2\boldsymbol{r}_2^T \\ -2\boldsymbol{r}_3^T \end{bmatrix} \quad \text{and} \quad \boldsymbol{b}_j = \begin{bmatrix} d_{2j}^2 - d_{1j}^2 - \boldsymbol{r}_2^T\boldsymbol{r}_2 \\ d_{3j}^2 - d_{1j}^2 - \boldsymbol{r}_3^T\boldsymbol{r}_3 \end{bmatrix}. \tag{17}$$

Since $d_{1j}^2 = \boldsymbol{s}_j^T\boldsymbol{s}_j$, we can eliminate the senders and form the equation system

$$d_{1j}^2 = \boldsymbol{b}_j^T(AA^T)^{-1}\boldsymbol{b}_j \quad \text{for} \quad j = 1, \dots, 4, \tag{18}$$

provided that A is invertible. Here, we will perform a change of variables, and instead of parameterizing the receivers in the coordinates $\boldsymbol{r}_i$, we use the squared inter-receiver distances c_{12}^2, c_{13}^2 and c_{23}^2, where $c_{ik} = \|\boldsymbol{r}_i - \boldsymbol{r}_k\|$, as we have observed this to produce more stable solvers.

For (18) to become polynomial it has to be multiplied with $\lambda^2 \det(AA^T)$, resulting in

$$\det(AA^T)(\lambda^2 D_{\Sigma 1j} - \lambda^4\hat{g}_1^2 - \hat{h}_j^2) = \lambda^2\boldsymbol{b}_j^T \operatorname{adj}(AA^T)\boldsymbol{b}_j \tag{19}$$

for $j = 1, \dots, 4$, where

$$AA^T = 2\begin{bmatrix} 2c_{12}^2 & c_{12}^2 + c_{13}^2 - c_{23}^2 \\ c_{12}^2 + c_{13}^2 - c_{23}^2 & 2c_{13}^2 \end{bmatrix}, \tag{20}$$

$$\boldsymbol{b}_j = \begin{bmatrix} D_{\Sigma 2j} - \lambda^2\hat{g}_2^2 - D_{\Sigma 1j} + \lambda^2\hat{g}_1^2 - c_{12}^2 \\ D_{\Sigma 3j} - \lambda^2\hat{g}_3^2 - D_{\Sigma 1j} + \lambda^2\hat{g}_1^2 - c_{13}^2 \end{bmatrix}. \tag{21}$$

However, this introduces spurious solutions causing the ideal generated by the polynomial system to not be zero-dimensional. For example, if $\lambda = 0$ the system reduces to the single equation $\det(AA^T) = 0$ which has infinitely many solutions. These spurious solutions can be removed by saturating with the unknowns $\{c_{12}^2, c_{13}^2, c_{23}^2, \lambda^2\}$ when generating the solver [9]. The produced solver has 14 solutions and an elimination template (see [26]) of size 192×206. The template can be reduced to 88×102 by saturating with $\{c_{12}^2, c_{13}^2, c_{23}^2\}$ algebraically before generating the solver. From the solutions, $\boldsymbol{r}_i$ and d_{ij} are easily found, after which $\boldsymbol{s}_j$ can be found by solving the linear systems in (17).

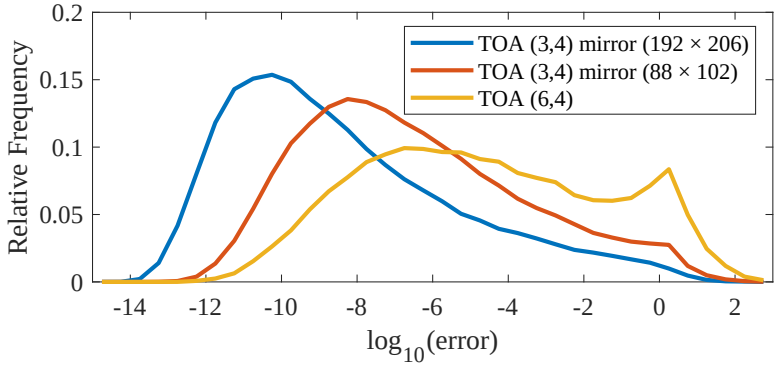


Figure 2: Distribution of distance errors from multiple trials when providing solvers with noiseless data.

Observe that by relaxing the mirroring constraints on $\boldsymbol{R}_{\wedge}$ and $\boldsymbol{R}_{\vee}$, we get the minimal TOA problem $(m, n) = (6, 4)$ for which solvers already exist [7]. However, those solvers are slower and unnecessarily big in the sense that they have 38 solutions and a template size of 493×531 [27]. Furthermore, together with the offset estimation in Section 4, we have constructed minimal solvers also for the TDOA case. Without the presence of a reflective plane, these problems are significantly more difficult [28].

7 Experiments

To evaluate the stability of our solvers, we generated synthetic TOA data consisting of receiver and sender coordinates drawn from $\mathcal{N}(0, 1)$. $D_{\wedge}$ and $D_{\vee}$ were calculated accordingly without added noise, and the heights were estimated as in Section 5 up to the scaling factor λ. Figure 2 shows the norm of the distance errors resulting from the estimated node positions. As can be seen, the proposed $(3, 4)$ solvers produce smaller errors than the existing $(6, 4)$ TOA solver. They are also significantly faster with execution times of 1.6 ms (192×206) and 0.7 ms (88×102), compared to the 18 ms of the $(6, 4)$ TOA solver.

In order to test our methods in a real setting, we constructed a controlled TOA-experiment. We used a number of synchronized microphones and a moving loudspeaker playing a musical piece. The experiment was done in an environment which also featured an independent motion capture system, in order to evaluate the results. Distance estimates were found using GCC-PHAT [29] between the microphones, and in order to have a controlled experiment we used the ground truth to estimate the time offset between the

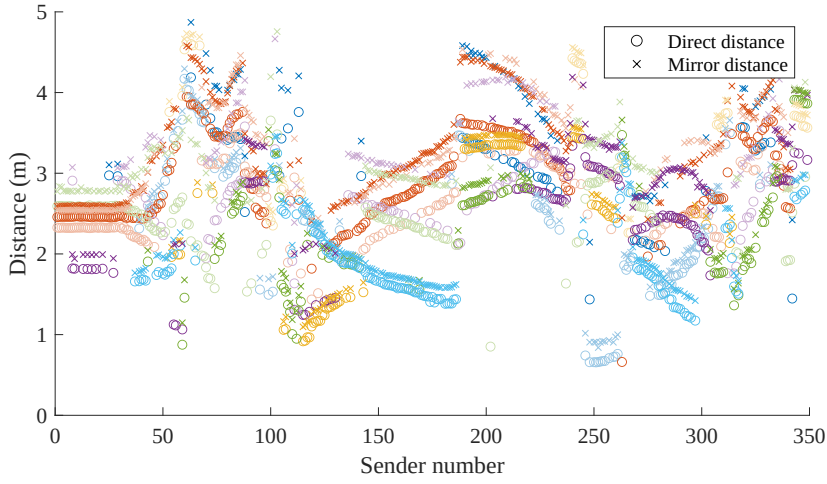


Figure 3: Distance measurements for the real sound experiment, with different colors for different microphones. Both the estimated direct path and the estimated mirror path are shown. Note that there is a very large amount of missing data.

speaker and the microphones. The resulting measurements for 11 microphones and 349 speaker positions are shown in Fig. 3. The dataset contains very little outliers, but very large amounts of missing data and noise in the measurements. We then proceeded to estimate both sender and receiver positions, using our stratified approach. The heights were found as described in Section 5, after which the minimal solver (192×206) described in Section 6 was used to estimate initial solutions for $\boldsymbol{r}_i$, $\boldsymbol{s}_j$ and λ. We used the solver in a RANSAC-voting scheme by letting the solutions vote for the correct height scaling factor. This gives an initial solution for the planar positions for three receiver and four sender positions, as well as an estimate of the global height scale. This solution was then extended using trilateration, with subsequent non-linear refinement. The results for the heights and the planar reconstruction are shown in Fig. 4, where also the ground truth is shown. The resulting mean errors in 3D-positions were in this case 8.7 cm for the receivers and 13 cm for the senders. Note that for a majority sender positions we have only three distance measurements.

8 Conclusion

In this paper, we have described how dominant reflective planes can be used to give powerful constraints on TOA and TDOA node calibration problems. We have devel-

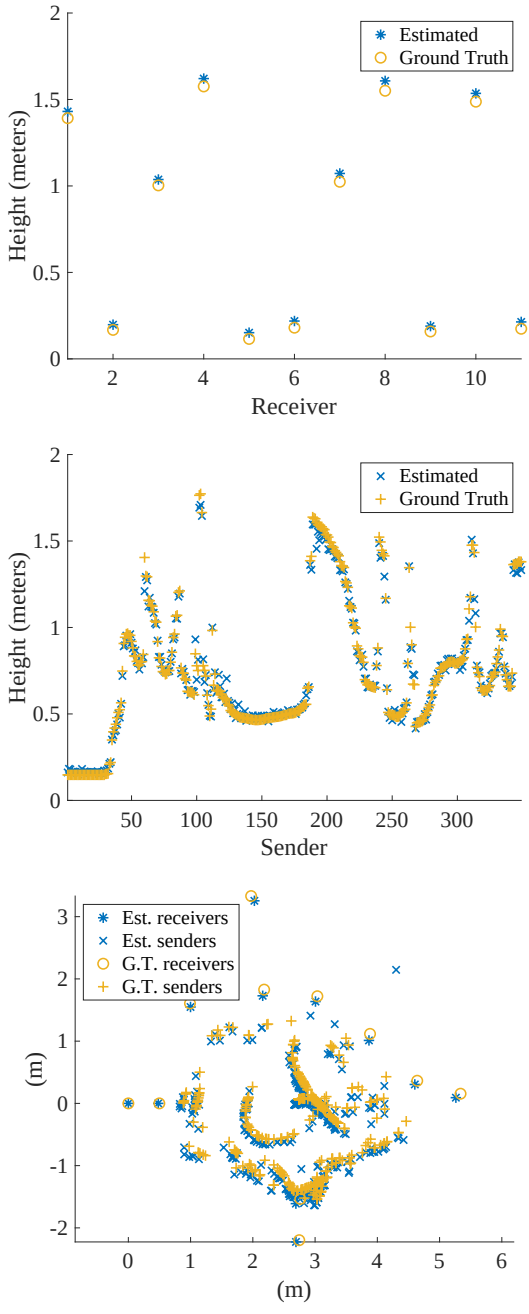


Figure 4: Results on the real sound experiment. Top and center show the estimated heights (in blue) for the receivers and senders, respectively, compared to the ground truth (in yellow). At the bottom, the estimated planar positions (in blue) for the receivers and senders are shown. The reconstruction has been rigidly registered to the ground truth (in yellow).

oped tractable methods, that in a stratified way, solves for time offsets, node heights and planar positions of nodes, using minimal solvers that can be efficiently applied in bootstrapping algorithms. We have further applied these methods to both synthetic and real data, with promising result.

References

[1] A. Plinge, F. Jacob, R. Haeb-Umbach, and G. A. Fink, "Acoustic microphone geometry calibration: An overview and experimental evaluation of state-of-the-art algorithms," *IEEE Signal Processing Magazine*, vol. 33, no. 4, pp. 14–29, 2016.

[2] S. T. Birchfield and A. Subramanya, "Microphone array position calibration by basis-point classical multidimensional scaling," *IEEE transactions on Speech and Audio Processing*, vol. 13, no. 5, 2005.

[3] M. Crocco, A. Del Bue, M. Bustreo, and V. Murino, "A closed form solution to the microphone position self-calibration problem," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2012.

[4] J. C. Chen, R. E. Hudson, and K. Yao, "Maximum likelihood source localization and unknown sensor location estimation for wideband signals in the near-field," *IEEE transactions on Signal Processing*, vol. 50, 2002.

[5] P. Pertila, M. Hamalainen, and M. Mieskolainen, "Passive temporal offset estimation of multichannel recordings of an ad-hoc microphone array," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 21, no. 11, pp. 2393–2402, Nov. 2013.

[6] H. Stewénius, "Gröbner basis methods for minimal problems in computer vision," Ph.D. dissertation, Lund University, 2005.

[7] Y. Kuang, S. Burgess, A. Torstensson, and K. Åström, "A complete characterization and solution to the microphone position self-calibration problem," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2013.

[8] K. Batstone, M. Oskarsson, and K. Åström, "Robust time-of-arrival self calibration with missing data and outliers," in *European Signal Processing Conference (EUSIPCO)*, 2016.

[9] V. Larsson, K. Åström, and M. Oskarsson, "Polynomial solvers for saturated ideals," in *International Conference on Computer Vision (ICCV)*. IEEE, 2017.

[10] M. Kreković, I. Dokmanić, and M. Vetterli, "Echoslam: Simultaneous localization and mapping with acoustic echoes," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 11–15.

[11] Y. Kuang and K. Åström, "Stratified sensor network self-calibration from tdoa measurements," in *European Signal Processing Conference (EUSIPCO)*, 2013.

[12] L. Wang, T. Hon, J. D. Reiss, and A. Cavallaro, "Self-localization of ad-hoc arrays using time difference of arrivals," *IEEE Transactions on Signal Processing*, vol. 64, no. 4, pp. 1018–1033, Feb 2016.

[13] M. Larsson, G. Flood, M. Oskarsson, and K. Åström, "Fast and robust stratified self-calibration using time-difference-of-arrival measurements," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 4640–4644.

[14] R. N. Treuhaft, S. T. Lowe, C. Zuffada, and Y. Chao, "2-cm GPS altimetry over Crater Lake," *Geophysical Research Letters*, vol. 28, no. 23, pp. 4343–4346, Dec. 2001. [Online]. Available: http://doi.wiley.com/10.1029/2001GL013815

[15] Y. E. Baba, A. Walther, and E. A. P. Habets, "3D Room Geometry Inference Based on Room Impulse Response Stacks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 5, pp. 857–872, May 2018, conference Name: IEEE/ACM Transactions on Audio, Speech, and Language Processing.

[16] I. Dokmanić, R. Parhizkar, A. Walther, Y. M. Lu, and M. Vetterli, "Acoustic echoes reveal room shape," *Proceedings of the National Academy of Sciences*, vol. 110, no. 30, pp. 12 186–12 191, Jul. 2013. [Online]. Available: https://pnas.org/doi/full/10.1073/pnas.1221464110

[17] I. Dokmanić, L. Daudet, and M. Vetterli, "How to localize ten microphones in one finger snap," in *2014 22nd European Signal Processing Conference (EUSIPCO)*. IEEE, 2014, pp. 2275–2279.

[18] F. Ribeiro, D. Ba, C. Zhang, and D. Florêncio, "Turning enemies into friends: Using reflections to improve sound source localization," in *2010 IEEE International Conference on Multimedia and Expo*, Jul. 2010, pp. 731–736, iSSN: 1945-788X.

[19] J. Scheuing and B. Yang, "Disambiguation of TDOA Estimation for Multiple Sources in Reverberant Environments," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 8, pp. 1479–1489, Nov. 2008, conference Name: IEEE Transactions on Audio, Speech, and Language Processing.

[20] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[21] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, no. 3, pp. 211–218, 1936. [Online]. Available: http://dx.doi.org/10.1007/BF02288367

[22] M. Oskarsson, K. Batstone, and K. Åström, "Trust no one: Low rank matrix factorization using hierarchical ransac," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5820–5829.

[23] V. Larsson and C. Olsson, "Convex low rank approximation," *International Journal of Computer Vision*, vol. 120, no. 2, pp. 194–214, 2016.

[24] A. Eriksson and A. Hengel, "Efficient computation of robust weighted low-rank matrix approximations using the L_1 norm," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2012.

[25] X. Guo and Z. Lin, "Low-rank matrix recovery via robust outlier estimation," *IEEE Transactions on Image Processing*, vol. 27, no. 11, pp. 5316–5327, 2018.

[26] V. Larsson, K. Åström, and M. Oskarsson, "Efficient solvers for minimal problems by syzygy-based reduction," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2383–2392.

[27] M. Larsson, G. Flood, M. Oskarsson, and K. Åström, "Upgrade Methods for Stratified Sensor Network Self-Calibration," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2020, pp. 4851–4855.

[28] L. Ferranti, K. Astrom, M. Oskarsson, J. Boutellier, and J. Kannala, "Sensor Networks TDOA Self-Calibration: 2D Complexity Analysis and Solutions," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Toronto, ON, Canada: IEEE, Jun. 2021, pp. 4635–4639. [Online]. Available: https://ieeexplore.ieee.org/document/9414634/

[29] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *IEEE transactions on acoustics, speech, and signal processing*, vol. 24, no. 4, pp. 320–327, 1976.

Paper VI

Optimal Trilateration is an Eigenvalue Problem

Martin Larsson[1,3], Viktor Larsson[2], Kalle Åström[1], Magnus Oskarsson[1]

[1] *Centre for Mathematical Sciences, Lund University, Lund, Sweden*
[2] *Department of Computer Science, ETH Zurich, Switzerland*
[3] *Combain Mobile AB*

Abstract: The problem of estimating receiver or sender node positions from measured receiver-sender distances is a key issue in different applications such as microphone array calibration, radio antenna array calibration, mapping and positioning using UWB or using round-trip-time measurements between mobile phones and WiFi-units. In this paper we address the problem of optimally estimating a receiver position given a number of distance measurements to known sender positions, so called *trilateration*. We show that this problem can be rephrased as an eigenvalue problem. We also address different error models and the multilateration setting where an additional offset is also unknown, and show that these problems can be modeled using the same framework.

Keywords: Trilateration, Calibration, Optimal estimation, Multilateration.

This work was partially supported by the strategic research projects ELLIIT and eSSENCE, the Swedish Foundation for Strategic Research project, Semantic Mapping and Visual Navigation for Smart Robots (grant no. RIT15-0038), and Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

1 Introduction

Sound localization has been a topic of interest in a wide range of applications for centuries, and is well known to be a difficult problem, especially in a reverberating room environment (see e.g. [1–8], and the references therein). Measuring the time it takes for the signal to reach each sensor, the position of the source can be estimated. In the literature, this is referred to as either time of arrival (TOA) estimation, if the time of signal emission is known, or otherwise time difference of arrival (TDOA) estimation, where only the relative time delays are used. Common techniques for delay estimation include different variations on cross-correlation or canonical correlation analysis (CCA), which then allows the sources to be located in a second step using tri- and multi-lateration (see e.g. [9, 10]). Other examples of sensor from which we can get distance measurements include Ultra-Wideband, WiFi signal strength and Narrowband Radio Signals [11–14].

1.1 Trilateration and Related Work

To formalize our problem, we want to recover the position of an unknown receiver $\mathbf{x} \in \mathbb{R}^n$, given the positions of N anchors $\mathbf{s}_j$ and distance measurements d_j to these anchors. Typically if the Euclidean distances are measured we aim at having $\|\mathbf{x} - \mathbf{s}_j\| \approx d_j$ for $j = 1, 2, \ldots, N$. For Gaussian noise the Maximum Likelihood (ML) estimate is given by the following optimization problem,

Problem 1 $\mathbf{x}^\star = \operatorname{argmin}_{\mathbf{x}\in\mathbb{R}^n} \sum_{j=1}^{N} \left(\|\mathbf{x} - \mathbf{s}_j\| - d_j \right)^2$.

This is a non-linear and non-convex optimization problem which can have several local minima.

Previously several methods have been proposed for solving Problem 1. In [15] the authors used an SDP relaxation approach. The authors solve a convex relaxation of the problem, and there is no guarantee that the solution will be optimal in the original cost. Recently [16] presented a fixed point iteration for solving Problem 1.

A standard way to derive a linear solver is to consider the equations $\|\mathbf{x} - \mathbf{s}_j\|^2 = d_j^2$. Forming differences between pairs of these equations the quadratic terms in $\mathbf{x}$ cancel, leaving only linear equations in $\mathbf{x}$,

$$2\left(\mathbf{s}_i - \mathbf{s}_j\right)^T \mathbf{x} = d_j^2 - d_i^2 + \mathbf{s}_j^T \mathbf{s}_j - \mathbf{s}_i^T \mathbf{s}_i \tag{1}$$

which can be solved in a least squares sense. However, this does not minimize any meaningful cost (see Figure 1). Variants of this method can be constructed by e.g. by

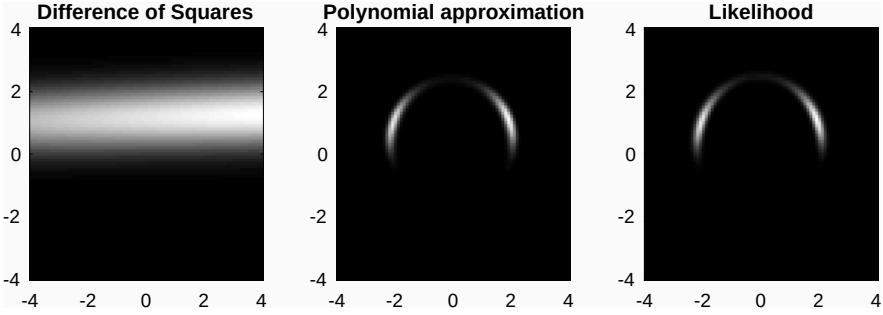


Figure 1: Comparison of cost functions for a synthetic instance with two local minima. Left to right: Difference of squares as in (1), Weighted approximation (6), True likelihood.

taking the difference of all points to one reference point or by taking the difference with the mean of all points, see [17].

In [15] the authors also present a globally optimal method for minimizing the following surrogate function,

$$h(\boldsymbol{x}) = \sum_j \left(\|\boldsymbol{x} - \mathbf{s}_j\|^2 - d_j^2 \right)^2 . \tag{2}$$

They consider the equivalent problem of minimizing

$$h(\boldsymbol{x}, \alpha) = \sum_j \left(\alpha - 2\boldsymbol{x}^T \mathbf{s}_j + \mathbf{s}_j^T \mathbf{s}_j - d_j^2 \right)^2 , \tag{3}$$

under the quadratic constraint $\alpha = \boldsymbol{x}^T \boldsymbol{x}$. They then set up the Lagrangian and after some manipulation end up with a single equation that only depends on the multiplier. In [15] this equation is then solved using bisection.

In [18] Zhou presented another method for the minimizing the squared distances loss. To solve the problem [18] introduce the artificial constraint $\sum_j \|\mathbf{x} - \mathbf{s}_j\|^2 = \sum_j d_j^2$, which is not satisfied in general, resulting in sub-optimal solutions.

In [19] the authors propose to minimize the maximum likelihood cost in Problem 1 by solving a sequence of weighted versions of (2), in an IRLS-like fashion [20].

1.2 Non-Gaussian Error Models

When estimating the distances using the signal strength (see e.g. [21]) the noise often becomes Gaussian when considering the measured power

$$P_j = b + k \log \|\mathbf{x} - \mathbf{s}_j\| + \epsilon, \quad \epsilon \in \mathcal{N}(0, \sigma) \tag{4}$$

or equivalently $\log\|\mathbf{x}-\mathbf{s}_j\|^2 = 2\frac{P_j-b}{k} + \epsilon'$ where $\epsilon' \in \mathcal{N}(0, 2\sigma/k)$. Here b and k are model parameters that we assume to be known. This leads to the following ML estimator (where $m_j = 2\frac{P_j-b}{k}$),

Problem 2 $\mathbf{x}^\star = \operatorname{argmin}_{\mathbf{x}\in\mathbb{R}^n} \sum_j \left(\log\|\mathbf{x}-\mathbf{s}_j\|^2 - m_j\right)^2$.

1.3 Multilateration

In the multilateration setting we have measurements to a number of known sender positions. These measurements contain a common unknown offset which also has to be estimated. In this case the ML estimate (under the assumption of Gaussian noise) is given as the solution to

Problem 3 $\mathbf{x}^\star = \operatorname{argmin}_{\mathbf{x}\in\mathbb{R}^n} \sum_{j=1}^{N} \left(\|\mathbf{x}-\mathbf{s}_j\| - (d_j + o)\right)^2$.

1.4 Paper Contributions

The cost functions in Problem 1 and 2 are both on the form

$$h(\mathbf{x}) = \sum_j \left(\Psi(\|\mathbf{x}-\mathbf{s}_j\|^2) - m_j\right)^2 \tag{5}$$

with $\Psi(x) = \sqrt{x}$ and $\Psi(x) = \log x$ respectively. We can get an approximation by replacing Ψ with its first order Taylor expansion. The linearization point is chosen as the point x_j which satisfies $\Psi(x_j) = m_j$. Note that this does not depend on the receiver position. By differentiating $\Psi(x)$ and inserting the linearization points, the approximations of the cost functions in Problem 1 and 2 become

$$h_1(\boldsymbol{x}) = \sum_j \frac{1}{4d_j^2}\left(\|\mathbf{x}-\mathbf{s}_j\|^2 - d_j^2\right)^2, \tag{6}$$

$$h_2(\boldsymbol{x}) = \sum_j e^{-2m_j}\left(\|\mathbf{x}-\mathbf{s}_j\|^2 - e^{m_j}\right)^2, \tag{7}$$

and for the multilateration cost function in Problem 3 we get

$$h_3(\boldsymbol{x}) = \sum_j \frac{1}{4(d_j+o)^2}\left(\|\mathbf{x}-\mathbf{s}_j\|^2 - (d_j+o)^2\right)^2. \tag{8}$$

Note that these approximate functions have very similar structure.

In this paper we present fast closed form solutions to the two non-convex optimization problems

$$\min_{\mathbf{x}} \sum_j w_j \left(\|\mathbf{x} - \mathbf{s}_j\|^2 - d_j^2 \right)^2, \tag{w-TOA}$$

$$\min_{\mathbf{x},\, o} \sum_j w_j \left(\|\mathbf{x} - \mathbf{s}_j\|^2 - (d_j + o)^2 \right)^2, \tag{w-TDOA}$$

by deriving equivalent eigenvalue problems. In contrast to previous approaches we can enumerate all stationary points and guarantee global optimality. Our method works for arbitrary dimension. Since we enumerate all stationary points we are also able to identify situations where there are multiple good competing hypotheses. Additionally we further explore the IRLS-like scheme used in [19] to minimize different cost functions corresponding to ML estimates for different noise distributions, for example as in Problem 2.

2 Optimal Trilateration

In this section we will show that the problem (w-TOA) is equivalent to an eigenvalue problem. The cost function in (w-TOA) can be written as

$$h(\boldsymbol{x}) = \sum_j w_j \left(\boldsymbol{x}^T \boldsymbol{x} - 2\boldsymbol{x}^T \boldsymbol{s}_j + \boldsymbol{s}_j^T \boldsymbol{s}_j - d_j^2 \right)^2. \tag{9}$$

Since the cost is differentiable everywhere, the globally optimal solution must lie at a stationary point of $h(\boldsymbol{x})$. The first order optimality conditions are $\nabla h(\boldsymbol{x}) =$

$$4 \sum_j w_j \left(\boldsymbol{x}^T \boldsymbol{x} - 2\boldsymbol{x}^T \boldsymbol{s}_j + \boldsymbol{s}_j^T \boldsymbol{s}_j - d_j^2 \right) (\boldsymbol{x} - \boldsymbol{s}_j) = 0. \tag{10}$$

This gives us n polynomial equation system of degree 3 in n unknowns. Naive application of the Bezout bound [22] for this system yields that there are at most 3^n solutions (where n is dimension of the ambient space). However, we will show that due to the specific structure of the equations, there are in general only $2n + 1$ stationary points.

2.1 Simplifying the Equations

Collecting the terms in (10) by degree we get

$$\frac{1}{4}\nabla h(\boldsymbol{x}) = (\textstyle\sum_j w_j)(\boldsymbol{x}^T\boldsymbol{x})\boldsymbol{x} \tag{11}$$

$$- \left(\boldsymbol{x}^T\boldsymbol{x}I + 2\boldsymbol{x}\boldsymbol{x}^T\right)\left(\textstyle\sum w_j\boldsymbol{s}_j\right) \tag{12}$$

$$+ \left(\sum_j w_j\left((\boldsymbol{s}_j^T\boldsymbol{s}_j - d_j^2)I + 2\boldsymbol{s}_j\boldsymbol{s}_j^T\right)\right)\boldsymbol{x} \tag{13}$$

$$+ \textstyle\sum_j w_j\left(d_j^2 - \boldsymbol{s}_j^T\boldsymbol{s}_j\right)\boldsymbol{s}_j = 0. \tag{14}$$

Since the global coordinate system is arbitrary we can choose this to simplify the equations. Similarly to the approach in [18] we start by translating the senders $\boldsymbol{s}_j$ with

$$\boldsymbol{t} = -(\textstyle\sum_j w_j\boldsymbol{s}_j)/(\sum_j w_j). \tag{15}$$

This ensures that $\sum_j w_j\boldsymbol{s}_j = 0$ which cancels all the quadratic terms in (12). Similarly, since the cost function is homogeneous in the weights, we can w.l.o.g. assume that $\sum_j w_j = 1$, which make the coefficients for all third degree terms one. The equations are now of the form

$$(\boldsymbol{x}^T\boldsymbol{x})\boldsymbol{x} + A\boldsymbol{x} + \boldsymbol{b} = 0. \tag{16}$$

The matrix A is symmetric, and thus we can perform an orthogonal eigenvalue decomposition $A = UDU^T$. Performing the change of variables $\boldsymbol{x} \to U\boldsymbol{x}$ and $\boldsymbol{b} \to U\boldsymbol{b}$, the equations separate and we get (since $\boldsymbol{x}^T\boldsymbol{x} = (U\boldsymbol{x})^T U\boldsymbol{x}$)

$$(\boldsymbol{x}^T\boldsymbol{x})x_i + D_{ii}x_i + b_i = 0, \quad i = 1, 2, \ldots, n. \tag{17}$$

2.2 Deriving the Eigenvalue Problem

Multiplying each equation in (17) with x_i we get

$$(\boldsymbol{x}^T\boldsymbol{x})x_i^2 + D_{ii}x_i^2 + b_i x_i = 0, \quad i = 1, 2, \ldots, n. \tag{18}$$

With some abuse of notation we use $\boldsymbol{x}^2$ to denote the vector of pure squares, i.e. $\boldsymbol{x}^2 = (x_1^2, x_2^2, \ldots, x_n^2)^T$. The equations in (17) and (18) can then be written as

$$(\boldsymbol{x}^T\boldsymbol{x})\begin{pmatrix}\boldsymbol{x}^2\\ \boldsymbol{x}\\ 1\end{pmatrix} = \underbrace{\begin{bmatrix}-D & -\mathrm{diag}(\boldsymbol{b}) & \boldsymbol{0}\\ \boldsymbol{0} & -D & -\boldsymbol{b}\\ \mathbb{1}^T & \boldsymbol{0}^T & 0\end{bmatrix}}_{=:M}\begin{pmatrix}\boldsymbol{x}^2\\ \boldsymbol{x}\\ 1\end{pmatrix}. \tag{19}$$

where the last row is the trivial equation $(\boldsymbol{x}^T\boldsymbol{x}) = \mathbb{1}^T\boldsymbol{x}^2$. Note that the matrix in (19) is a constant square matrix and does not depend on the unknowns $\boldsymbol{x}$. Additionally the matrix M can easily be computed for a given problem instance.

Any solution to the original problem (10) must also satisfy (19) (after appropriate change of variables). This means that for any solution $\boldsymbol{x}$ we have that the vector $(\boldsymbol{x}^2, \boldsymbol{x}, 1)^T$ is an eigenvector to M with eigenvalue $\boldsymbol{x}^T\boldsymbol{x}$. So to enumerate all solutions to the original system we do the following:

1. Compute matrix M as described above.
2. Compute all eigenvectors.
3. Normalize eigenvectors such that the last element is one and extract $\boldsymbol{x}$ from the corresponding elements.
4. Evaluate original cost at each stationary point candidate and choose the one with smallest cost.

While it is possible that there are eigenvectors which do not correspond to a stationary point, all stationary points are among the eigenvectors, so by enumerating all of them we are guaranteed to find the global optimum.

2.3 Stationary Points and Degenerate Configurations

From the results in the previous section, it follows that there are at most $2n+1$ stationary points with different values for $\boldsymbol{x}^T\boldsymbol{x}$. In general each eigenvector will corresponds to a solution of the original system. However, it is possible to have eigenvalues with higher multiplicity. This happens for example in degenerate situations (e.g. all senders lie on a line). Note that in this case (19) still holds, and we still recover the correct eigenvalues (i.e. values of $\boldsymbol{x}^T\boldsymbol{x}$). However, since the eigenspaces are not necessarily one-dimensional extra care must be taken to recover the solutions.

2.4 Optimal Multilateration

In this section we consider the multilateration case where we also need to estimate an unknown offset between the senders and the receiver. Similarly to the trilateration problem the cost function can be expanded as $h(\boldsymbol{x}, o) =$

$$\sum_j w_j \left(\boldsymbol{x}^T\boldsymbol{x} - 2\boldsymbol{x}^T\boldsymbol{s}_j + \boldsymbol{s}_j^T\boldsymbol{s}_j - d_j^2 - 2d_j o - o^2\right)^2 . \tag{20}$$

The first order optimality conditions are[1] $0 = \frac{1}{4}\nabla b(\boldsymbol{x}, o) =$

$$\sum_j w_j \left(\boldsymbol{x}^T\boldsymbol{x} - o^2 - 2\left(\boldsymbol{x}^T, o\right)\begin{pmatrix}\boldsymbol{s}_j \\ d_j\end{pmatrix} + \boldsymbol{s}_j^T\boldsymbol{s}_j - d_j^2 \right)\begin{pmatrix}\boldsymbol{x} - \boldsymbol{s}_j \\ o + d_j\end{pmatrix}.$$

Again, shifting the coordinate systems (this time also in o) we can cancel the quadratic terms, and the equations become

$$(\boldsymbol{x}^T\boldsymbol{x} - o^2)\left(\boldsymbol{x}, o\right)^T + \mathcal{A}\left(\boldsymbol{x}, o\right)^T + \boldsymbol{b} = 0. \tag{21}$$

Now the goal is to transform these equations into an eigenvalue problem. Unfortunately it is not possible to use the diagonalization trick here. The last $n+1$ rows are already given by (21) and we only need to determine the top n rows, i.e.

$$(\boldsymbol{x}^T\boldsymbol{x} - o^2)\begin{pmatrix}\boldsymbol{x}^2 \\ o^2 \\ \boldsymbol{x} \\ o \\ 1\end{pmatrix} = \begin{bmatrix} ? & ? & ? \\ \boldsymbol{0}^T & -\mathcal{A} & -b \\ (\mathbb{1}^T, -1) & \boldsymbol{0}^T & 0 \end{bmatrix}\begin{pmatrix}\boldsymbol{x}^2 \\ o^2 \\ \boldsymbol{x} \\ o \\ 1\end{pmatrix}, \tag{22}$$

Let $\lambda = \boldsymbol{x}^T\boldsymbol{x} - o^2$. Multiplying (21) with $\mathrm{diag}(\boldsymbol{x}, o)$ we get

$$\lambda\left(\boldsymbol{x}^2, o^2\right)^T = -\mathrm{diag}(\boldsymbol{x}, o)\left(\mathcal{A}\left(\boldsymbol{x}, o\right)^T + \boldsymbol{b}\right). \tag{23}$$

This almost yields the missing rows, except for a few quadratic mixed terms (e.g. x_1x_2) appearing in the RHS. To eliminate the mixed quadratic terms, the goal is to the express them in monomials appearing in the eigenvector, i.e. $(\boldsymbol{x}^2, o^2, \boldsymbol{x}, o, 1)$.

Multiplying the first equation in (21) with x_2 and subtracting the second equation multiplied with x_1, any terms containing λ cancel, and we are left with an equation containing only mixed quadratic terms and monomials which appear in the eigenvector in (22). Doing this for all pairs of equations in (21) yields a set of $\binom{n+1}{2}$ equations, on the form

$$C_0\boldsymbol{m} + C_1\left(\boldsymbol{x}^2, o^2, \boldsymbol{x}, o, 1\right)^T = 0 \tag{24}$$

where $\boldsymbol{m}$ is the vector of monomials containing the mixed quadratic terms. Inserting $\boldsymbol{m} = -C_0^{-1}C_1\left(\boldsymbol{x}^2, o^2, \boldsymbol{x}, o, 1\right)^T$ into (23) we can eliminate all mixed terms and recover the missing rows in the eigenvalue problem (22).

[1] To simplify calculations the last equation has changed sign.

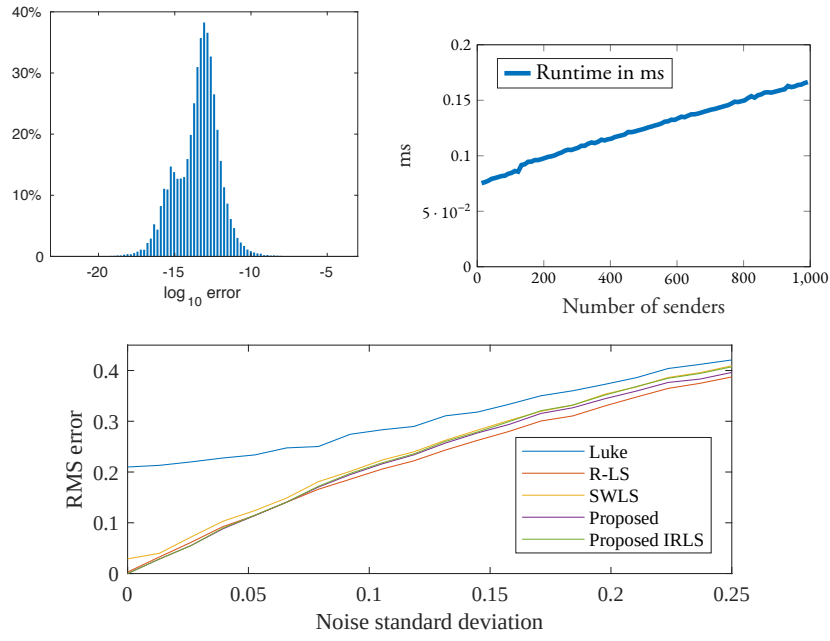


Figure 2: *Top left:* Distribution of errors for synthetic noise-less instances. *Top right:* Runtime (in ms) for different number of senders. Bottom: RMS error in receiver position for various amounts of noise.

3 Experiments and Applications

3.1 Numerical Stability and Computational Cost

We evaluated our method on synthetically generated data. Figure 2 shows the equation residuals (left) and the runtime plotted against the number of senders (right).

3.2 Maximum Likelihood Estimation using IRLS

Since we can solve the weighted problem (w-TOA) optimally, we can use this to iteratively minimize the true ML cost functions. We iterate the following two steps until convergence:

1. $\boldsymbol{x}^t = \arg\min_{\mathbf{x}} \sum_j w_j^t \left(\|\mathbf{x} - \mathbf{s}_j\|^2 - d_j^2 \right)^2$
2. $w_j^{t+1} = (2\|\mathbf{x}^t - \mathbf{s}_j\| \left(\|\mathbf{x}^t - \mathbf{s}_j\| + d_j \right))^{-1}.$

These weights are chosen such that the gradients of the ML and the weighted cost align in each iteration, i.e.

$$\nabla_{\mathbf{x}} \, w_j \left(\|\mathbf{x} - \mathbf{s}_j\|^2 - d_j^2 \right)^2 = \nabla_{\mathbf{x}} \, \left(\|\mathbf{x} - \mathbf{s}_j\| - d_j \right)^2 \tag{25}$$

This ensures that any limit point is a stationary point of the original cost. Note that this weighting is different from the weighting used in [19].

Similarly, to solve Problem 2 we update the weights as

$$w_j^t = \frac{\log(\|\mathbf{x}^t - \mathbf{s}_j\|^2) - m_j}{\log(\|\mathbf{x}^t - \mathbf{s}_j\|^2)(\|\mathbf{x}^t - \mathbf{s}_j\|^2 - e^{m_j})}. \tag{26}$$

Figure 2 shows the RMS error in receiver positions for different amounts of noise in a setup of six senders and one receiver, all sampled uniformly from a unit cube. We compare the proposed method and its IRLS application as just presented with Luke [16], the R-LS solver from [15] and the SWLS solver from [19]. All methods perform similarly except for Luke which occasionally convergences to the wrong stationary point.

3.3 Real Data Experiment

We evaluate our method using TOA datasets gathered with an ultra-wideband (UWB) setup. Six senders were kept stationary as a single receiver was moved through the setup. Ground truth for sender and receiver positions was determined using an optical motion capture system. We compare the proposed method with Zhou [18], Luke [16], the SR-LS solver from [15] and the R-LS solver from [15] (see Table 1). For reference we include the ML estimate found by solving Problem 1 using standard iterative optimization methods initialized at the ground truth positions. All algorithms were implemented in MATLAB. In many cases the proposed solver without IRLS performs best. This is likely due to that the errors are not completely Gaussian.

4 Conclusion

In this paper we have introduced two eigenvalue solvers that give closed-form-solutions to two different non-linear weighted least squares problems. We have also shown how these solvers can be used to do optimal trilateration and multilateration.

Table 1: RMS errors (meters) and total execution time (seconds) when running eight real UWB datasets.

Data-set	Zhou [18]	Luke [16]	SR-LS [15]	R-LS [15]	ML	Prop.	Prop. IRLS
1	0.66	0.34	0.40	0.34	0.34	**0.32**	0.34
2	0.65	**0.52**	0.54	**0.52**	**0.52**	**0.52**	**0.52**
3	11.05	0.64	1.18	0.64	0.64	**0.42**	0.64
4	0.54	0.31	0.44	0.32	0.31	**0.30**	0.31
5	0.64	0.34	0.41	0.34	0.34	**0.33**	0.34
6	0.47	0.31	0.37	0.31	0.31	**0.28**	0.31
7	0.53	0.32	0.35	0.32	0.32	**0.31**	0.32
8	0.74	0.39	0.48	0.39	0.39	**0.36**	0.39
Time	**0.43**	6.49	3.63	93.00	33.41	0.50	2.13

References

[1] J. Meldercreutz, "Om Längders Mätning Genom Dåns Tilhielp," *Vetenskapsakademiens Handlingar*, vol. 2, pp. 73–77, 1741.

[2] B. Champagne, S. Bedard, and A. Stephenne, "Performance of time-delay estimation in the presence of room reverberation," *IEEE Trans. Speech Audio Process.*, vol. 4, no. 2, pp. 148–152, Mar 1996.

[3] J. H. DiBiase, H. F. Silverman, and M. S. Brandstein, "Robust localization in reverberent rooms," in *Microphone Arrays: Techniques and Applications*, M. Brandstein and D. Ward, Eds. New York: Springer-Verlag, 2001, pp. 157–180.

[4] T. Gustafsson, B. D. Rao, and M. Trivedi, "Source localization in reverberant environments: modeling and statistical analysis," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 6, pp. 791–803, Nov 2003.

[5] E. Kidron, Y. Y. Schechner, and M. Elad, "Cross-modal localization via sparsity," *IEEE Trans. Signal Process.*, vol. 55, no. 4, pp. 1390–1404, April 2007.

[6] M. D. Gillette and H. F. Silverman, "A linear closed-form algorithm for source localization from time-differences of arrival," *IEEE Signal Processing Letters*, vol. 15, pp. 1–4, 2008.

[7] K. C. Ho and M. Sun, "Passive source localization using time differences of arrival and gain ratios of arrival," *IEEE Trans. Signal Process.*, vol. 56, no. 2, pp. 464–477, Feb 2008.

[8] X. Alameda-Pineda and R. Horaud, "A geometric approach to sound source localization from time-delay estimates," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 6, pp. 1082–1095, June 2014.

[9] H. F. Silverman and S. E. Kirtman, "A two-stage algorithm for determining talker location from linear microphone array data," *Computer Speech & Language*, vol. 6, no. 2, pp. 129–152, 1992. [Online]. Available: http://www.sciencedirect.com/science/article/pii/088523089290023W

[10] G. Shen, R. Zetik, and R. Thoma, "Performance comparison of toa and tdoa based location estimation algorithms in los environment," in *Positioning, Navigation and Communication, 2008. WPNC 2008. 5th Workshop on*, March 2008, pp. 71–78.

[11] C. Rizos, A. G. Dempster, B. Li, and J. Salter, "Indoor positioning techniques based on wireless lan," 2007.

[12] A. A. Adebomehin and S. D. Walker, "Enhanced ultrawideband methods for 5g los sufficient positioning and mitigation," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2016 IEEE 17th International Symposium on A*. IEEE, 2016, pp. 1–4.

[13] X. Li, K. Batstone, K. Åström, M. Oskarsson, C. Gustafson, and F. Tufvesson, "Robust phase-based positioning using massive mimo with limited bandwidth," in *28th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC 2017.* IEEE–Institute of Electrical and Electronics Engineers Inc., 2 2018.

[14] K. Batstone, M. Oskarsson, and K. Åström, "Towards real-time time-of-arrival self-calibration using ultra-wideband anchors," in *Indoor Positioning and Indoor Navigation (IPIN), 2017 International Conference on*. IEEE, 2017, pp. 1–8.

[15] A. Beck, P. Stoica, and J. Li, "Exact and approximate solutions of source localization problems," *IEEE Transactions on signal processing*, vol. 56, no. 5, pp. 1770–1778, 2008.

[16] D. R. Luke, S. Sabach, M. Teboulle, and K. Zatlawey, "A simple globally convergent algorithm for the nonsmooth nonconvex single source localization problem," *Journal of Global Optimization*, vol. 69, no. 4, pp. 889–909, 2017.

[17] W. Navidi, W. S. Murphy Jr, and W. Hereman, "Statistical methods in surveying by trilateration," *Computational statistics and data analysis*, vol. 27, no. 2, pp. 209–228, 1998.

[18] Y. Zhou, "A closed-form algorithm for the least-squares trilateration problem," *Robotica*, vol. 29, no. 3, pp. 375–389, 2011.

[19] A. Beck, M. Teboulle, and Z. Chikishev, "Iterative minimization schemes for solving the single source localization problem," *SIAM Journal on Optimization*, vol. 19, no. 3, pp. 1397–1416, 2008.

[20] K. Aftab and R. Hartley, "Convergence of iteratively re-weighted least squares to robust m-estimators," in *Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on*. IEEE, 2015, pp. 480–487.

[21] T. S. Rappaport *et al.*, *Wireless communications: principles and practice*. prentice hall PTR New Jersey, 1996, vol. 2.

[22] D. A. Cox, J. Little, and D. O'Shea, *Using algebraic geometry*, ser. Graduate Texts in Mathematics. Springer-Verlag New York, 2005, vol. 185.

Paper VII

Trilateration Using Motion Models

Martin Larsson[1,2], Erik Tegler[1], Kalle Åström[1], Magnus Oskarsson[1]

[1] *Centre for Mathematical Sciences, Lund University, Lund, Sweden*
[2] *Combain Mobile AB*

Abstract: In this paper, we present a framework for doing localization from distance measurements, given an estimate of the local motion. We show how we can register the local motion of a receiver, to a global coordinate system, using trilateration of given distance measurements from the receivers to senders in known positions. We describe how many different motion models can be formulated within the same type of registration framework, by only changing the transformation group. The registration is based on a test and hypothesis framework, such as RANSAC, and we present novel and fast minimal solvers that can be used to bootstrap such methods. The system is tested on both synthetic and real data with promising results.

Keywords: Trilateration, time-of-arrival, motion model, odometry, IMU, minimal solvers, RANSAC.

This work was partially supported by the Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation, and the strategic research projects ELLIIT and eSSENCE.

Reprinted from *2022 25th International Conference on Information Fusion (FUSION)*, M. Larsson, E. Tegler, K. Åström, and M. Oskarsson, Trilateration Using Motion Models, Copyright 2022, with permission from International Society of Information Fusion (ISIF).

1 Introduction

Where am I? This is a question that is fundamental to most living organisms in the world. Both the methods and sensor inputs used for answering this question vary wildly. In this paper, we will specifically look into how one can combine a local motion model with distance measurements to a global coordinate system in order to do positioning.

Many animals—from desert ants [1] to mammals [2]—use dead reckoning or path integration based on locomotion estimates to navigate, and such techniques have also been used by humans throughout history for navigating both at sea and land. Typically one gets quite accurate results from dead reckoning based on, e.g., estimated headings and speed, but these results invariably suffer from drift.

Using (direct or indirect) distance measurements has also been used for a long time, e.g., using sound waves from as early as the eighteenth century [3]. Given a small set of distance measurements (in the plane at least two and in space at least three) to known locations, one can estimate the position using trilateration. These distances are often acquired using approaches based on Time Of Arrival (TOA), Received Signal Strength (RSS), or Time Difference Of Arrival (TDOA). In such cases a signal, e.g, radio, sound, or light, is typically emitted from the senders and received at the sought position, or vice versa. For TOA and TDOA the speed of the signal in the medium is most often assumed to be known, which then directly gives distance measurements. For RSS, a signal propagation model can be used to translate power measurements to distances. One problem with trilateration techniques is that one often has gross outliers in the data, arising from errors when correlating signals, from the geometry of the problem, or multipath effects.

Our contribution in this paper is a way of robustly handling outliers, and making use of distance measurements, at the same time as we use any available motion model or odometry data to locally constrain the motion, and by this way aggregating distance measurements. Specifically, we will formulate our trilateration problem as registering a local receiver coordinate system to a global sender coordinate system, so that the distance measurements are realized. We propose to do this robustly in a RANSAC framework [4–6], so that we can handle outliers in the distance measurements. We will focus on the most tractable case, when the receivers are in a plane, i.e., the two-dimensional case. In order to do this, we need to develop a number of minimal trilateration registration solvers, that are used as components in bootstrapping the solutions. In Section 2 we describe the geometry of our setup. Then in Section 3 we describe the proposed solvers, and how they were developed. We also test our approach using the proposed solvers, both on synthetic and real data, in Section 4.

The Matlab-Mex implementations are publicly available[1].

1.1 Related Work

The problem considered presently can be described as the simultaneous trilateration and registration of points. Regarding trilateration, also known as single source localization, there has been a large body of previous work. The minimal problem, where the number of distance measurements equals the spatial dimension, has a closed-form solution [7–9]. However, when the problem is over-determined, some form of maximum likelihood (ML) estimator must be constructed. Finding the ML estimate given Gaussian noise in the measurements is a nonlinear, non-smooth and non-convex problem, and unlike the minimal case, it lacks a closed-form solution. Multiple iterative methods, with various convergence guaranties, have nevertheless been proposed [10–13]. There has also been several contributions where the ML problem is relaxed, and the error in the squared distance measurements are minimized [14–17]. Various forms of linear solvers have also been proposed, see [18] and references therein. Common for the works listed above is that they do not treat the scenario when gross outliers are present. These outliers can result from non-line-of-sight (NLOS) measurements, in which case they are necessarily longer than the line-of-sight (LOS) distance. These measurements can either be identified and removed [19, 20] or incorporated into the model [21].

The positioning resulting from trilateration can be improved by incorporating a motion model. The most common approach here is using filtering, to fuse the data from different sources, e.g., using a Kalman filter [22–24] or hidden Markov models [25]. This can give significantly better result, especially in the case of noisy RSS distance measurements. However, by the real-time nature of these filters, outlier detection might not be trivial, and such approaches also rely on a reasonable initialization. The most similar approach, to our proposed framework, is perhaps within robotics and autonomous vehicles, using angle measurements combined with odometry [26]. Although, the underlying governing measurement equations are completely different.

When it comes to registration of corresponding point sets, it has been long known that the globally optimal solution is found by singular value decomposition (SVD) [27–29]. Robust methods that allows for missing data and outliers have since then been proposed, see [30–32] and references therein. However, the problem considered here is slightly different, in that the corresponding points are required to be a specific distance from each other rather than being coincident. This fundamentally changes the problem and

[1]The code for all presented solvers is publicly available at *https://github.com/hamburgerlady/motion-model-trilateration*

requires a more involved approach. Closer similarities can be found within the study of forward kinematics of parallel manipulators. In particular, forward kinematics of the well known Stewart platform [33,34] and the planar 3-RPR manipulator [35] is identical to the minimal rigid registration problem considered here, in 3D and 2D, respectively. However, to the best of our knowledge, this type of problem has never before been applied in the context of robust trilateration with motion priors. Furthermore, when the scale of the registration transform is unknown, the problem becomes different from the forward kinematics one.

2 Problem Formulation

We will now describe our approach in more detail and give motivation and use cases where it is suitable. We assume that we have measured the distances d_{ij} from a number of receivers $\mathbf{r}_i \in \mathbb{R}^N$ and senders $\mathbf{s}_j \in \mathbb{R}^N$,

$$d_{ij}^2 = |T(\mathbf{r}_i) - \mathbf{s}_j|^2, \tag{1}$$

where the receivers are given in a local coordinate system and T is a transformation from the receiver coordinate system to the global sender coordinate system. We will now consider the problem when $\mathbf{r}_i$ and $\mathbf{s}_j$ are known but the transformation T is unknown. Note that the problem is agnostic in what we label as receivers and senders. We will henceforth, for notation purposes, assume that the receivers are in the local coordinate system. Furthermore we can w.l.o.g. assume that we enumerate our senders and receivers so that $i = j$, (with the possible need of duplicating receiver or sender positions), so that

$$d_i^2 = |T(\mathbf{r}_i) - \mathbf{s}_i|^2. \tag{2}$$

Depending on the degrees of freedom of T, we would need differently many distance measurements d_i in order to minimally estimate T. We will now give a number of examples when the above formulation is an appropriate model for localizing a number of unknown receivers.

We will in this paper assume that the receivers are in the plane, i.e., $N = 2$. One can do the same analysis in 3D, but the specific problems become much harder. In many cases, we have quite accurate information on the local motion of a moving receiver. This could for instance be measurements from odometry systems or Inertial Measurement Unit (IMU) measurements, but it could also be a constrained motion model. Consider the following hierarchy of knowledge about the motion of a receiver, where we assume that motion is:

1. in a straight line with constant speed.
2. in a straight line and with known relative speed.
3. in a straight line and with known absolute speed.
4. with known direction and with constant speed.
5. with known direction and with known relative speed.
6. with known direction and with known absolute speed.

Relative speeds could be realized using, e.g., a step counter with unknown step length, absolute speeds with a calibrated step counter or odometry, and directions using an IMU. All these cases can be parameterized using a local coordinate systems for the receiver positions, and can be registered to the sender coordinate using either a similarity transform (Cases 1, 2, 4 and 5) or a Euclidean transformation (Cases 3 and 6). Since a similarity transformation in the plane has four degrees of freedom, we would need at least four distance measurements to solve (2). For the rigid Euclidean transformation, we would need at least three measurements.

3 Solvers

In this section, we will describe our proposed minimal solvers for the 2D cases. Since these solvers typically are used in a RANSAC framework [4–6], the most important aspect of them is that they are fast. We will use an approach based on the action matrix method [36] and use an automatic solver generator [37, 38]. The process is based on formulating problem instances—the system of multivariate polynomial equations that we want to solve—with random coefficient from a finite field, where the analysis can be done in exact arithmetic using a computer algebra system such as Macaulay2 [39]. From these calculations the solver can be constructed. The resulting solver contains two major steps, a linear system—the so-called elimination template—to construct the action matrix, and then the solution is extracted using an eigenvalue solver of the action matrix. Please see [37] for details on the process. Even though the solver generation is automatic, the properties of the resulting solver (in terms of speed, accuracy and robustness) will heavily rely on the parametrization used.

3.1 Minimal Euclidean Three-Point Solver

We assume that we have three receivers, $\mathbf{r}_i \in \mathbb{R}^2$ in a local coordinate system, that we want to register to the corresponding senders, $\mathbf{s}_j \in \mathbb{R}^2$, in the global coordinate system, so that,

$$d_i^2 = |R\mathbf{r}_i + \mathbf{t} - \mathbf{s}_i|^2, \quad i = 1, 2, 3, \tag{3}$$

where R is the unknown two-dimensional rotation and $\mathbf{t} \in \mathbb{R}^2$ is the unknown translation. Since $R^T R = I$ we can simplify (3) to

$$d_i^2 = \mathbf{r}_i^T \mathbf{r}_i + 2\mathbf{r}_i^T R^T (\mathbf{t} - \mathbf{s}_i) + \mathbf{t}^T \mathbf{t} - 2\mathbf{s}_i^T \mathbf{t} + \mathbf{s}_i^T \mathbf{s}_i, \tag{4}$$

for $i = 1, 2, 3$. We can see that these equations are linear in R, but there is a cross-term between R and $\mathbf{t}$ and also a quadratic term in $\mathbf{t}$. We will parameterize the rotation using two parameters (a, b), so that

$$R = \begin{bmatrix} a & -b \\ b & a \end{bmatrix}, \quad a^2 + b^2 = 1. \tag{5}$$

With the non-linear constraint on (a, b) and (4) we have four equations in four unknowns. If we use this system and construct a solver using the automatic solver from [37], we get in general six (possibly complex) solutions, and an elimination template of size 45×51. However, we can get a faster solver by reducing the problem to a single univariate sextic polynomial. This was done in [35] using Cayley-Menger determinants, but our approach is as follows. If we take differences between two equations in (4), the quadratic terms $\mathbf{t}^T \mathbf{t}$ cancel out. This means that from two such differences, we get linear expressions in $\mathbf{t}$ from which we can solve directly for the translation (as a function of the rotation R). Inserting this expression for $\mathbf{t}$ in (4) gives an equation in only (a, b), with terms up to total degree five in (a, b). We simplify this expression further, by substituting all powers of b^2 and higher using $b^2 = 1 - a^2$. This gives us a new equation in a and b,

$$c_1 a^3 + c_2 a^2 b + c_3 a^2 + c_4 ab + c_5 a + c_6 b + c_7 = 0, \tag{6}$$

where $c_k, k = 1, \ldots, 7$, are coefficients only depending on the measured input data. Note that it is now linear in b. Multiplying this equation with b and again substituting all powers of b^2 using $b^2 = 1 - a^2$ gives another equation that is linear in b. We now have two equations on the form

$$\underbrace{\begin{bmatrix} f_{11}(a) & f_{12}(a) \\ f_{21}(a) & f_{22}(a) \end{bmatrix}}_{M} \begin{bmatrix} b \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \tag{7}$$

where $f_{kl}(a)$ are polynomials in a with coefficients that only depend on the input data. Since (7) should have a non-trivial solution, we get our final equation in a by

$$\det M = e_6 a^6 + e_5 a^5 + e_4 a^4 + e_3 a^3 + e_2 a^2 + e_1 a + e_0 = 0. \tag{8}$$

This equation can be solved very fast and efficiently using the companion matrix or by Sturm sequences [40, 41].

3.2 Minimal Similarity Four-Point Solver

For the similarity transformation case we get the same type of equations as for the Euclidean case. The difference is that we use four equations on the form,

$$d_i^2 = |S\mathbf{r}_i + \mathbf{t} - \mathbf{s}_i|^2, \quad i = 1, \ldots, 4, \tag{9}$$

where S is a scaled rotation matrix. We can parametrize the problem in more or less the same way as for the Euclidean case, using four parameters, but now without the scale constraint on (a, b), so that

$$S = \begin{bmatrix} a & -b \\ b & a \end{bmatrix}. \tag{10}$$

We could solve for the translation also in this case, using the same approach as for the Euclidean case. However, this will for the similarity case introduce spurious solutions (arising from the denominator of the solution in $\mathbf{t}$). One could use saturation to automatically eliminate the spurious solutions using the technique describe in [42], but this will in this case lead to a much slower solver. Instead, we keep the initial formulation (9) and use this as input to the automatic solver generator. The system of equations has in general up to six solutions, and the generated solver has an elimination template of size 53×59.

4 Experiments

In this section, we will evaluate our solvers and test them as system components for robust trilateration. We will do a number of systematic tests on synthetic data, and show a proof-of-concept evaluation on a dataset with real Round-Trip Time (RTT) measurements. We have implemented our two minimal solvers in MATLAB. A number of the time-consuming steps were implemented as compiled Mex-C++ subroutines. On a standard laptop (Macbook pro 2.5 GHz Dual-Core Intel Core i7 running MATLAB 2020b), the average execution times are 25 µs for the Euclidean solver and 57 µs for the similarity solver.

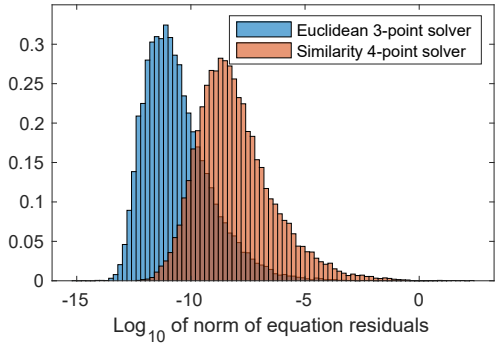


Figure 1: Distribution of the norm of equations residuals for 10 000 instances of running our minimal solvers. Note that the errors are on a logarithmic scale.

4.1 Solver Tests on Synthetic Data

In order to test the numerical stability of our solvers, without noise, we generated synthetic minimal problem instances (three sender-receiver-distance measurements for the Euclidean solver and four sender-receiver-distance measurements for the similarity solver). We then ran our minimal solvers and evaluated the equation residuals. We repeated this 10 000 times in order to generate error statistics. In Figure 1 the error distributions for the two minimal solvers are shown. One can see that we get small errors in general. The errors for the similarity solver are slightly larger than for the Euclidean solver. This is to be expected since the elimination template is larger. In order to test our solvers in the presence of noise we ran our solvers in a RANSAC framework. We randomly placed ten receivers and ten senders in an area of size 50×50. We then calculated the distances between the receivers and senders. In order to simulate a local coordinate system, we randomly chose a transformation (Euclidean respectively similarity) and applied this to the receiver positions. Finally, we added Gaussian noise to all positions, with a standard deviation of σ.

The unknown transformation between the local receiver coordinate system and the global sender coordinate system was then estimated using RANSAC and our minimal respective solvers. The estimated transformation was compared to the ground truth transformation in terms of rotation angle difference and relative distance between translations. For the similarity case we also compared the scale. In the left of Figure 2, the resulting errors are shown as functions of σ for the added noise (on a logarithmic scale). One can see that the errors degrade gracefully. This first test was done without outliers in the data. For a second test we repeated the experiment, but we also corrupted a certain percentage of the data grossly, in order to test the sensitivity to outliers. The right of Figure 2 shows

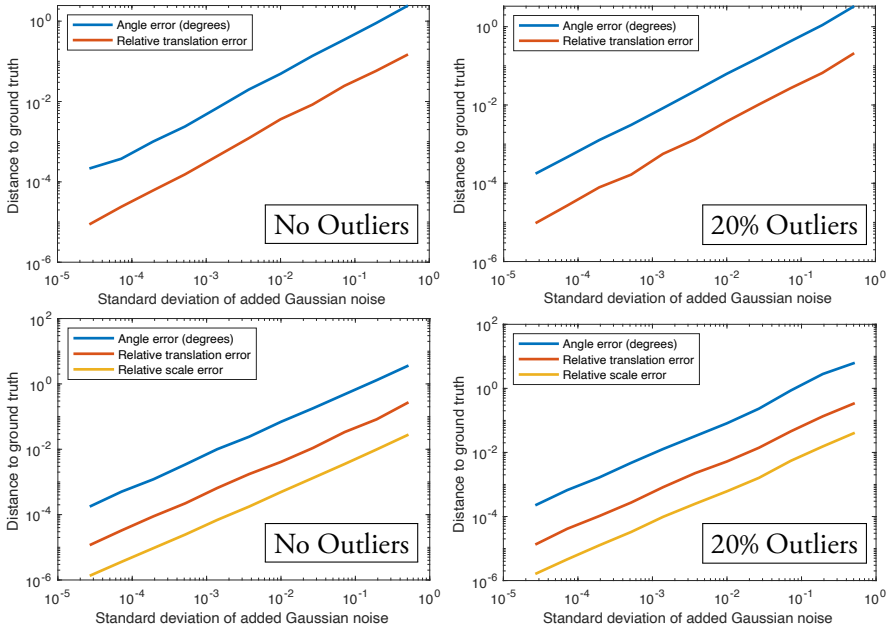


Figure 2: Results on synthetic data for varying degrees of noise and outliers (on a log-log scale). Top left shows comparison to ground truth after RANSAC using the proposed three-point Euclidean solver, as a function of the added noise. Here there are no outliers in the data. Top right shows the same plot for 20 % outliers in the input data. Bottom row shows the result after RANSAC using the proposed four-point similarity solver. Bottom left is without outliers and bottom right with 20 % outliers. One can see that the errors degrade gracefully for all cases.

Table 1: The error characteristics used in the synthetic system experiment

	Standard deviation	Outlier ratio
Step Counter	0.5 steps	-
Heading measurement	0.005 rad	-
Distance measurement	0.1 m	10 %

the errors in this case, with 20 % outliers. Here we get more or less the same error characteristic as without outliers. Since we only have ten measurements in this case, and we need three or four for an estimate, more than around 30 % outliers will lead to a large degradation of the results.

4.2 System Tests on Synthetic Data

In order to test our proposed approach in a more realistic scenario, we simulated an indoor positioning scenario, with a person walking with a step counter, heading estimation and distance measurements to a set of known sender positions. There are of course a large number of parameters that one can set in such a scenario, but we have tried to make a simple test, with typical error characteristics, based on, e.g., sound measurements. We generated 20 random positioned senders in a 50 m × 50 m area. We then simulated a path with 50 receiver positions. At each position we measure the distance to three randomly chosen senders. We also assume that we have measured the number of steps and heading between each receiver position. We add Gaussian noise to all measurements, and also corrupt a certain ratio of the distance measurements in order to simulate, e.g., NLOS measurements. The used error statistics for the experiment are summarized in Table 1. In order to test our method, we divided the set of 50 receiver positions into ten segments of five positions in each segment. In each segment we parameterized a local coordinate system using the measured number of steps and the measured heading. For the Euclidean case, we also assume that the step length is known (0.75 m), but for the similarity case this parameter is estimated. For each segment we have 15 distance measurements, since we have three distance measurements for each receiver position. For each segment we estimate the transformation to the global coordinate system using RANSAC with our minimal solvers. We repeated the experiment 100 times. For a simple comparison, we also report the results using optimal trilateration [14] for the individual receiver positions (based on three measurements, without the use of odometry data) and dead reckoning based on the odometry data without the use of distance measurements. The results are shown in Table 2. One example reconstruction is shown in Figure 3. One

Table 2: Comparison to ground truth positions on synthetic system

Method	Optimal trilateration [14]	Dead reckoning*	Proposed (Euclidean)	Proposed (similarity)
RMS (m)	16.2	2.00	**0.179**	0.498

*Assumes ground truth initialization of first position.

can see that the receiver positions are generally reconstructed well. Also shown are the reconstructions using dead reckoning and optimal trilateration. These reconstructions suffer from drift and outliers respectively.

4.3 System Test on Semi-Synthetic Data

We have done some preliminary tests also based on semi-synthetic data. Using a phone, we have recorded real RTT measurements in an office environment. A person walked around the office while collecting measurements to Wi-Fi hotspots. At a number of positions the user marked the location manually on a map of the office. This corresponds to the selective ground truth for the experiment. In addition, we also know the location of all beacons. Here, we do not have access to any reliable motion estimates, so we have used synthetic motion data (heading and step counter) based on the ground truth positions and assuming a linear motion between the ground truth positions. We added Gaussian noise to the motion data with standard deviations of 0.5 steps and 0.01 rad for the step counter and heading, respectively. Using the real distance measurements (with standard deviation on the order of meters) we ran our system and compared the result with the sparse ground truth. In this case we got an average RMS of 2.6 m compared to 3.9 m using optimal trilateration on the individual ground truth receivers. One of the reconstructions is shown in Figure 4. This experiment should be seen as a proof-of-concept, since the results are based on synthetically generated motion measurements, and also depend on a number of parameter settings in the system. However, it shows promising results of being able to aggregate distance measurements over several receiver positions and eliminate outliers, to get more robust and accurate results.

5 Conclusion

In this paper, we have presented a framework for performing robust trilateration using motion measurements or a constrained motion model. We have shown that a number of different motion priors can be formulated as registration of the local motion coordinate

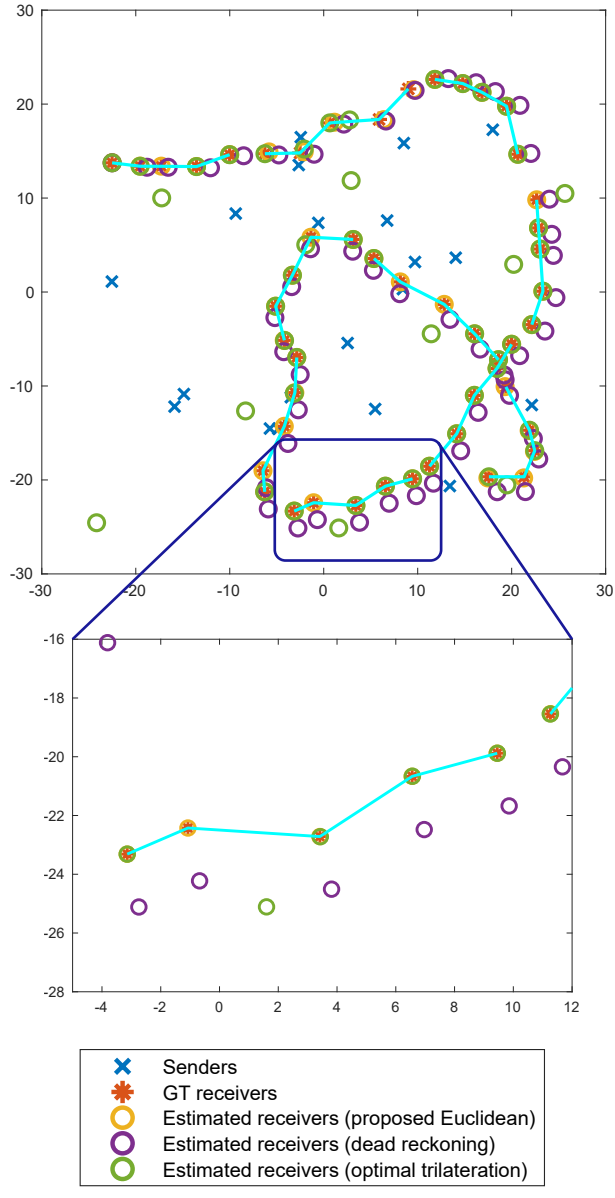


Figure 3: Reconstruction and comparison with ground truth for the synthetic system experiment. Left shows the complete reconstruction. The proposed solution is very close to the ground truth positions in this case. On the right, a magnification of one segment is shown. The solution based on only odometry data (in purple) suffers from drift. The solution based on optimal trilateration (in green) is very accurate for many receiver positions, but breaks down for cases where there are outliers in the distance measurements.

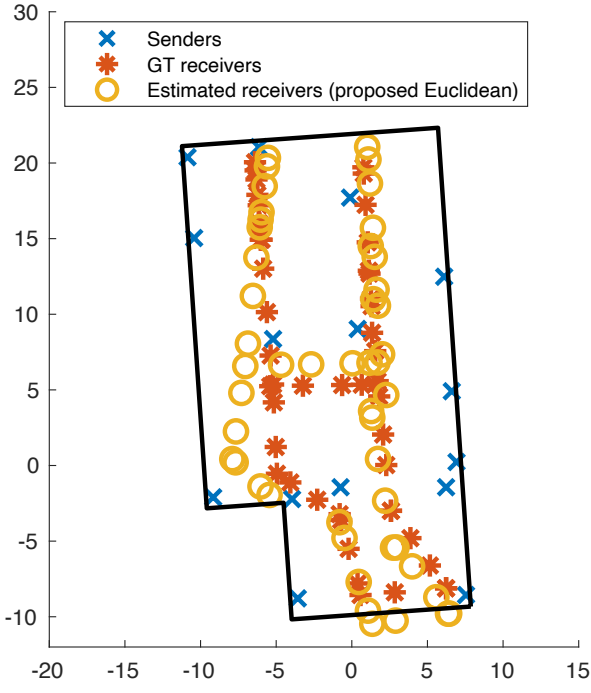


Figure 4: Reconstruction and comparison with ground truth for the semi-synthetic system experiment. Real Wi-Fi RTT distance measurements were used together with synthetic motion data in the proposed approach.

system to a global coordinate system, so that the estimated distance measurements are realized. Depending on the motion model, we use either a similarity transform or a rigid Euclidean transform. We have presented fast minimal solvers for the two-dimensional versions of these problem, that can be used to bootstrap efficient and robust RANSAC-type hypothesize and test methods. We have tested the solvers on synthetic data, and also shown preliminary tests, based on real RTT measurements, with promising results. Future work includes developing the systems, so that they are robust to different scenarios. We would also like to tackle the more challenging three-dimensional case.

References

[1] M. Wittlinger, R. Wehner, and H. Wolf, "The ant odometer: stepping on stilts and stumps," *science*, vol. 312, no. 5782, pp. 1965–1967, 2006.

[2] A. S. Etienne and K. J. Jeffery, "Path integration in mammals," *Hippocampus*, vol. 14, no. 2, pp. 180–192, 2004.

[3] J. Meldercreutz, "Om Längders Mätning Genom Dåns Tilhielp," *Vetenskapsakademiens Handlingar*, vol. 2, pp. 73–77, 1741.

[4] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography," *Commun. Assoc. Comp. Mach.*, 1981.

[5] O. Chum, T. Werner, and J. Matas, "Two-view geometry estimation unaffected by a dominant plane," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1. IEEE, 2005, pp. 772–779.

[6] D. Barath, J. Noskova, M. Ivashechkin, and J. Matas, "Magsac++, a fast, reliable and accurate robust estimator," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 1304–1312.

[7] F. Thomas and L. Ros, "Revisiting trilateration for robot localization," *IEEE Transactions on Robotics*, vol. 21, no. 1, pp. 93–101, Feb. 2005, conference Name: IEEE Transactions on Robotics.

[8] D. E. Manolakis, "Efficient solution and performance analysis of 3-D position estimation by trilateration," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, no. 4, pp. 1239–1248, Oct. 1996, conference Name: IEEE Transactions on Aerospace and Electronic Systems.

[9] I. D. Coope, "Reliable computation of the points of intersection of n spheres in R^n," *ANZIAM Journal*, vol. 42, pp. C461–C477, Dec. 2000. [Online]. Available: https://journal.austms.org.au/ojs/index.php/ANZIAMJ/article/view/608

[10] D. R. Luke, S. Sabach, M. Teboulle, and K. Zatlawey, "A simple globally convergent algorithm for the nonsmooth nonconvex single source localization problem," *Journal of Global Optimization*, vol. 69, no. 4, pp. 889–909, Dec. 2017. [Online]. Available: https://doi.org/10.1007/s10898-017-0545-6

[11] A. Beck, M. Teboulle, and Z. Chikishev, "Iterative Minimization Schemes for Solving the Single Source Localization Problem," *SIAM Journal on Optimization*, vol. 19, no. 3, pp. 1397–1416, Jan. 2008. [Online]. Available: http://epubs.siam.org/doi/10.1137/070698014

[12] R. Jyothi and P. Babu, "SOLVIT: A Reference-Free Source Localization Technique Using Majorization Minimization," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2661–2673, 2020, conference Name: IEEE/ACM Transactions on Audio, Speech, and Language Processing.

[13] N. Sirola, "Closed-form algorithms in mobile positioning: Myths and misconceptions," in *Navigation and Communication 2010 7th Workshop on Positioning*, Mar. 2010, pp. 38–44.

[14] M. Larsson, V. Larsson, K. Åström, and M. Oskarsson, "Optimal Trilateration Is an Eigenvalue Problem," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2019, pp. 5586–5590, iSSN: 2379-190X.

[15] Y. Zhou, "A closed-form algorithm for the least-squares trilateration problem," *Robotica*, vol. 29, no. 3, pp. 375–389, May 2011, publisher: Cambridge University Press. [Online]. Available: https://www.cambridge.org/core/journals/robotica/article/closedform-algorithm-for-the-leastsquares-trilateration-problem/FC7B1E4BAADD781FEE559DD304A31409

[16] A. Beck, P. Stoica, and J. Li, "Exact and Approximate Solutions of Source Localization Problems," *IEEE Transactions on Signal Processing*, vol. 56, no. 5, pp. 1770–1778, May 2008, conference Name: IEEE Transactions on Signal Processing.

[17] K. W. Cheung, H. C. So, W. Ma, and Y. T. Chan, "Least squares algorithms for time-of-arrival-based mobile location," *IEEE Transactions on Signal Processing*, vol. 52, no. 4, pp. 1121–1130, Apr. 2004, conference Name: IEEE Transactions on Signal Processing.

[18] P. Stoica and J. Li, "Lecture Notes - Source Localization from Range-Difference Measurements," *IEEE Signal Processing Magazine*, vol. 23, no. 6, pp. 63–66, Nov. 2006, conference Name: IEEE Signal Processing Magazine.

[19] H. J. Jo and S. Kim, "Indoor smartphone localization based on los and nlos identification," *Sensors*, vol. 18, no. 11, p. 3987, 2018.

[20] Yiu-Tong Chan, Wing-Yue Tsui, Hing-Cheung So, and Pak-chung Ching, "Time-of-arrival based localization under NLOS conditions," *IEEE Transactions on Vehicular Technology*, vol. 55, no. 1, pp. 17–24, Jan. 2006, conference Name: IEEE Transactions on Vehicular Technology.

[21] Xin Wang, Zongxin Wang, and B. O'Dea, "A TOA-based location algorithm reducing the errors due to non-line-of-sight (NLOS) propagation," *IEEE Transactions on Vehicular Technology*, vol. 52, no. 1, pp. 112–116, Jan. 2003, conference Name: IEEE Transactions on Vehicular Technology.

[22] A. Poulose, O. S. Eyobu, and D. S. Han, "A combined pdr and wi-fi trilateration algorithm for indoor localization," in *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*. IEEE, 2019, pp. 072–077.

[23] G. Chen, X. Meng, Y. Wang, Y. Zhang, P. Tian, and H. Yang, "Integrated wifi/pdr/smartphone using an unscented kalman filter algorithm for 3d indoor localization," *Sensors*, vol. 15, no. 9, pp. 24 595–24 614, 2015.

[24] N. Yu, X. Zhan, S. Zhao, Y. Wu, and R. Feng, "A precise dead reckoning algorithm based on bluetooth and multiple sensors," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 336–351, 2017.

[25] J. Liu, R. Chen, L. Pei, R. Guinness, and H. Kuusniemi, "A hybrid smartphone indoor positioning solution for mobile lbs," *Sensors*, vol. 12, no. 12, pp. 17 208–17 233, 2012.

[26] M. Oskarsson and K. Åström, "Accurate and automatic surveying of beacon positions for a laser guided vehicle," *Proc. European Consortium for Mathematics in Industry, Gothenburg, Sweden*, 1998.

[27] B. K. P. Horn, "Closed-form solution of absolute orientation using unit quaternions," *Journal of the Optical Society of America A*, vol. 4, 1987.

[28] B. Horn, H. M. Hilden, and S. Negahdaripour, "Closed-form solution of absolute orientation using ortonormal matrices," *Journal of the Optical Society of America A*, 1988.

[29] W. Kabsch, "A solution for the best rotation to relate two sets of vectors," *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, vol. 32, no. 5, pp. 922–923, 1976.

[30] X. Bai, Z. Luo, L. Zhou, H. Chen, L. Li, Z. Hu, H. Fu, and C.-L. Tai, "Pointdsc: Robust point cloud registration using deep spatial consistency," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 859–15 869.

[31] F. Crosilla, A. Beinat, A. Fusiello, E. Maset, and D. Visintini, "Orthogonal procrustes analysis," in *Advanced Procrustes Analysis Models in Photogrammetric Computer Vision*. Springer, 2019, pp. 7–28.

[32] F. Pomerleau, F. Colas, and R. Siegwart, "A review of point cloud registration algorithms for mobile robotics," *Foundations and Trends in Robotics*, vol. 4, no. 1, pp. 1–104, 2015.

[33] B. Dasgupta and T. Mruthyunjaya, "The Stewart platform manipulator: a review," *Mechanism and Machine Theory*, vol. 35, no. 1, pp. 15–40, Jan. 2000. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0094114X99000063

[34] M. Furqan, M. Suhaib, and N. Ahmad, "Studies on Stewart platform manipulator: A review," *Journal of Mechanical Science and Technology*, vol. 31, no. 9, pp. 4459–4470, Sep. 2017. [Online]. Available: http://link.springer.com/10.1007/s12206-017-0846-1

[35] N. Rojas and F. Thomas, "The forward kinematics of 3-r<formula formulatype="inline"><tex notation="tex">$\underline{P}$</tex></formula>r planar robots: A review and a distance-based formulation," *IEEE Transactions on Robotics*, vol. 27, no. 1, pp. 143–150, 2011.

[36] D. Cox, J. Little, and D. O'shea, *Using algebraic geometry*. Springer, 2005.

[37] V. Larsson, K. Åström, and M. Oskarsson, "Efficient solvers for minimal problems by syzygy-based reduction," in *CVPR*, 2017.

[38] V. Larsson, M. Oskarsson, K. Åström, A. Wallis, Z. Kukelova, and T. Pajdla, "Beyond gröbner bases: Basis selection for minimal solvers," in *CVPR*, 2018.

[39] D. R. Grayson and M. E. Stillman, "Macaulay2, a software system for research in algebraic geometry," Available at http://www.math.uiuc.edu/Macaulay2/.

[40] D. Hook and P. McAree, "Using sturm sequences to bracket real roots of polynomial equations," in *Graphics gems*. Academic Press Professional, Inc., 1990, pp. 416–422.

[41] C. F. Sturm, "Résolution des équations algébriques," *Bulletin de Férussac*, vol. 11, pp. 419–425, 1829.

[42] V. Larsson, K. Åström, and M. Oskarsson, "Polynomial solvers for saturated ideals," in *ICCV*, 2017.

Paper VIII

Single Source Localization As an Eigenvalue Problem

Martin Larsson[I,2], Viktor Larsson[I], Kalle Åström[I], Magnus Oskarsson[I]

[1] *Centre for Mathematical Sciences, Lund University, Lund, Sweden*
[2] *Combain Mobile AB*

Abstract: In this paper, a novel method for solving the single source localization problem is presented, enabling efficient positioning of a source using time-of-arrival measurements. The proposed method formulates the problem as an eigenvalue problem, enabling the usage of existing eigensolvers, and thus producing a fast, numerically stable and easy to implement solver. Unlike previous works, degenerate cases are treated, where multiple and possibly infinitely many solutions exist. Furthermore, by the introduction of suitable weightings in the cost function, multiple noise distributions in the measurements can be accommodated. The proposed method is validated against a range of state-of-the-art methods using synthetic and real data, where it is shown to be among the fastest and most numerically stable, while also being able to handle degenerate cases.

Keywords: source localization, trilateration, global optimization, generalized trust-region subproblem.

This work was partially supported by the Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation. The authors gratefully acknowledge Lund University Humanities Lab.

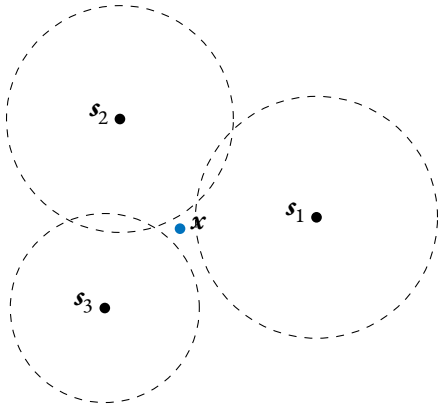


Figure 1: Three senders s_1, s_2, s_3 and the maximum likelihood solution x to the trilateration problem.

1 Introduction

The single source localization problem has received a considerable amount of attention due to its broad application within areas such as GNSS positioning, wireless networks, speaker diarization, robotics, and indoor positioning. The problem consists of finding an unknown receiver position using some form of distance measurements to known sender locations. These distances are often acquired using approaches based on Time Of Arrival (TOA), Received Signal Strength (RSS), or Time Difference Of Arrival (TDOA). In all cases some signal, e.g, radio, sound, or light, is emitted from the senders and received at the sought position, or vice versa.

In the case of TOA, the one-way propagation time of the signal is measured, and knowing the propagation speed in the medium, the distance traveled can be calculated. RSS instead utilizes the fact that signals get attenuated in the medium, and by modeling this attenuation, a distance measurement can be derived from the received signal strength. Solving the single source localization problem for the cases of TOA and RSS is called *trilateration*. TDOA is similar to TOA, but the former features an additional unknown offset in the distance measurement that needs to be estimated. This fundamentally changes the single source localization problem which is then referred to as *multilateration*.

The problem of trilateration can be formalized as having m senders $\boldsymbol{s}_j \in \mathbb{R}^n$, $j = 1, \ldots, m$, of known position with distance measurements $d_j \in \mathbb{R}$ to an unknown receiver position $\boldsymbol{x} \in \mathbb{R}^n$. This can be modeled as

$$d_j = \|\boldsymbol{x} - \boldsymbol{s}_j\| + \epsilon_j, \tag{1}$$

where ϵ_j represents additive noise in the measurements. Provided i.i.d. Gaussian noise the Maximum Likelihood (ML) estimator for $\boldsymbol{x}$ is given by

$$\underset{\boldsymbol{x}}{\text{minimize}} \sum_{j=1}^{m} (\|\boldsymbol{x} - \boldsymbol{s}_j\| - d_j)^2. \tag{2}$$

This is a nonlinear, nonsmooth and nonconvex optimization problem which can have multiple local minima. As a result, proposed approaches have so far been limited to iterative methods [1–5] or have relied on some form of relaxation of the problem. Common for the iterative methods is that, while they under some circumstances guarantee convergence to a stationary point of (2), they do not guarantee convergence to a global minimum and are thus dependent on a reasonable initialization. A non-iterative approach was proposed in [6], where a convex relaxation of the problem was solved using Semidefinite Programming (SDP). However, due to the relaxation, there is no guarantee that the solution is optimal in the original cost function.

Another relaxation is given by minimizing the error in the squared distances

$$\underset{\boldsymbol{x}}{\text{minimize}} \sum_{j=1}^{m} (\|\boldsymbol{x} - \boldsymbol{s}_j\|^2 - d_j^2)^2. \tag{3}$$

This formulation has the benefit of being polynomial and smooth, although, it lacks the statistical interpretation that (2) has and gives higher influence to larger errors and distances. A solution was provided in [6] where they considered the equivalent problem

$$\underset{\boldsymbol{x},\alpha}{\text{minimize}} \sum_{j=1}^{m} (\alpha - 2\boldsymbol{x}^T\boldsymbol{s}_j + \boldsymbol{s}_j^T\boldsymbol{s}_j - d_j^2)^2, \tag{4}$$

under the constraint $\alpha = \boldsymbol{x}^T\boldsymbol{x}$. This resulted in a Generalized Trust Region Subproblem (GTRS) which was reduced to a single equation in one variable and subsequently solved using bisection. In an earlier work [7], a similar approach was used, where also a weighting of the terms was introduced, approximating (2) for small errors. A closed-form approximate solution to (3) was proposed in [8], where they introduced the artificial constraint $\sum_j \|\boldsymbol{x} - \boldsymbol{s}_j\|^2 = \sum_j d_j^2$, which is not satisfied in general, resulting in sub-optimal solutions. In the minimal case, i.e., when $m = n$, an exact solution can be found and several closed-form methods have been proposed [9–11]. Various linear methods have also been proposed, of which several are equivalent to the unconstrained problem in (4), see [12] and references therein.

1.1 Contribution

In this paper we present a novel method based on eigendecomposition for solving the optimization problem

$$\underset{x}{\text{minimize}} \sum_{j=1}^{m} w_j (\|\boldsymbol{x} - \boldsymbol{s}_j\|^2 - d_j^2)^2, \tag{5}$$

where $w_j \geq 0$ is a weighting. We show how to derive suitable weightings depending on the error distribution in the measurements, and, in particular, how to approximate (2). We then analyze the eigendecomposition problem and show that the global minimizer of (5) corresponds to the largest real eigenvalue. Using this fact and other observations, we construct an efficient and numerically stable algorithm for performing trilateration. Finally, we validate the proposed method against a range of state-of-the-art methods on both real and synthetic data. The material presented here is partially based on the conference paper [13].

2 The Weighted Cost Function

In the presence of noise, minimizing the cost function in (5) is not the same as minimizing the one in (2). However, with a suitable weighting, the former provides an accurate approximation of the latter, while also accommodating other noise distributions.

To derive the weightings, we start by introducing the residual functions

$$r_j(\boldsymbol{x}) = \Psi_j(\|\boldsymbol{x} - \boldsymbol{s}_j\|^2) - \Psi_j(d_j^2), \tag{6}$$

where $\Psi_j(z)$ is a normalization transformation that is differentiable at $z = d_j^2$. These are the residuals we wish to minimize in the least squares sense. Assuming $\Psi_j(d_j^2)$ contains Gaussian noise with covariance matrix $\mathbf{P}^{-1}$, the ML-estimate for $\boldsymbol{x}$ is given by minimizing

$$h_0(\boldsymbol{x}) = \boldsymbol{r}(\boldsymbol{x})^T \mathbf{P} \boldsymbol{r}(\boldsymbol{x}), \tag{7}$$

where $\boldsymbol{r}(\boldsymbol{x}) \in \mathbb{R}^m$ is the residual vector with elements $r_j(\boldsymbol{x})$. Note that if we choose $\Psi_j(z) = \sqrt{z}$ and $\Psi_j(z) = z$, we get the optimization problems in (2) and (3), respectively.

Depending on $\Psi_j(z)$, minimizing $h_0(\boldsymbol{x})$ can be a difficult problem. We will therefore, in each residual $r_j(\boldsymbol{x})$, replace $\Psi_j(z)$ with its first order Taylor approximation at d_j^2.

$$\Psi_j(z) \approx \Psi_j(d_j^2) + \Psi_j'(d_j^2)(z - d_j^2). \tag{8}$$

Insertion into (6) yields the approximate residuals

$$\tilde{r}_j(\boldsymbol{x}) = \Psi_j'(d_j^2)\left(\|\boldsymbol{x} - \boldsymbol{s}_j\|^2 - d_j^2\right). \tag{9}$$

Let $\mathbf{W}$ be a positive definite weighting matrix with elements $w_{ij} = \Psi_i'(d_i^2)P_{ij}\Psi_j'(d_j^2)$. Note that $\mathbf{W}$ is constant and does not depend on $\boldsymbol{x}$. The cost function approximating $h_0(\boldsymbol{x})$ can now be written as

$$h(\boldsymbol{x}) = \frac{1}{4}\sum_{i=1}^{m}\sum_{j=1}^{m} w_{ij}(\|\boldsymbol{x} - \boldsymbol{s}_i\|^2 - d_i^2)(\|\boldsymbol{x} - \boldsymbol{s}_j\|^2 - d_j^2), \tag{10}$$

which is a general form of (5). This is the cost function for which the proposed method finds the global minimum.

2.1 Different Noise Distributions

Accommodating different noise distributions in the measurements is achieved by finding suitable normalization transformations $\Psi_j(z)$ such that $\Psi_j(d_j^2)$ is Gaussian. For example, when working with TOA measurements, the distances d_j are often modeled as containing Gaussian additive noise. In this case, letting $\Psi_j(z) = \sqrt{z}$ will cause $\Psi_j(d_j^2)$ to be Gaussian assuming $d_j > 0$. The weighting in the approximate residuals (9) are then given by

$$\Psi_j'(d_j^2) = \frac{1}{2d_j}. \tag{11}$$

In particular, if the noise is i.i.d., then $\mathbf{P} = \mathbf{I}$ and $\mathbf{W}$ is diagonal with elements $w_{jj} = 1/4d_j^2$. The same weighting was derived in [7].

A different example is when RSS measurements are used. Then the signal strength can be modeled using the log-distance path loss model [14, Chapter 8], also known as the one-slope model [15, Chapter 4.7],

$$C_j = (C_0)_j - 10\eta_j \log_{10}(d_j) + \epsilon_j, \tag{12}$$

where $(C_0)_j$ and η_j are known parameters and ϵ_j is Gaussian noise. Given a signal measurement C_j the corresponding distance d_j can be calculated. Letting $\Psi_j(z) = 5\eta_j \log_{10}(z)$, we get that $\Psi_j(d_j^2)$ is Gaussian and the weighting in the approximate residuals (9) are given by

$$\Psi_j'(d_j^2) = \frac{5\eta_j}{d_j^2 \log 10}. \tag{13}$$

Note that these weightings are undefined when $d_j = 0$. In practice, this is not a problem as we can simply clamp d_j to some suitable small number, e.g., $d_j \leftarrow \max(d_j, 10^{-3})$ for $j = 1, \ldots, m$.

2.2 Trilateration With Partially Known Receiver Position

In some scenarios, the receiver position might be partially known, e.g., the receiver might be confined to a plane while the senders are free in 3D space. It turns our that, modifying the cost function in (10) to allow for this, simply reduces to a trilateration problem in a lower dimension.

Assume that k coordinates of the receiver position are known. We can then partition $\boldsymbol{x}$ into $\boldsymbol{x}' \in \mathbb{R}^{n-k}$ and $\boldsymbol{x}'' \in \mathbb{R}^k$, representing the unknown and known coordinates of $\boldsymbol{x}$, respectively. Correspondingly, we partition the sender positions $\boldsymbol{s}_j$ into $\boldsymbol{s}_j' \in \mathbb{R}^{n-k}$ and $\boldsymbol{s}_j'' \in \mathbb{R}^k$. Given that $\|\boldsymbol{x} - \boldsymbol{s}_j\|^2 = \|\boldsymbol{x}' - \boldsymbol{s}_j'\|^2 + \|\boldsymbol{x}'' - \boldsymbol{s}_j''\|^2$, we can rewrite the approximate residuals in (9) as

$$\tilde{r}_j(\boldsymbol{x}) = \Psi_j'(d_j^2)\left(\|\boldsymbol{x}' - \boldsymbol{s}_j'\|^2 - (d_j')^2\right). \tag{14}$$

where $(d_j')^2 = d_j^2 - \|\boldsymbol{x}'' - \boldsymbol{s}_j''\|^2$. These residuals are on the same form as (9) but over a lower dimension and can thus be solved using the proposed method.

3 Eigenvalue Formulation

In this section we will derive a method for minimizing $h(\boldsymbol{x})$ in (10) by transforming the first order optimality conditions into an eigenvalue problem. The global minimizer can then be extracted from the largest real eigenvalue. We will also show how to handle degenerate cases, where there are more than one global minimizer.

Differentiating $h(\boldsymbol{x})$ and collecting the terms by degree yields

$$\nabla h(\boldsymbol{x}) = \sum_{i=1}^{m}\sum_{j=1}^{m} w_{ij}(\|\boldsymbol{x}-\boldsymbol{s}_i\|^2 - d_i^2)(\boldsymbol{x}-\boldsymbol{s}_j) \tag{15}$$

$$= \left(\sum_{i=1}^{m}\sum_{j=1}^{m} w_{ij}\right)(\boldsymbol{x}^T\boldsymbol{x})\boldsymbol{x} \tag{16}$$

$$- (\boldsymbol{x}^T\boldsymbol{x}I + 2\boldsymbol{x}\boldsymbol{x}^T)\left(\sum_{ij} w_{ij}\boldsymbol{s}_i\right) \tag{17}$$

$$+ \left(\sum_{i=1}^{m}\sum_{j=1}^{m} w_{ij}\left(2\boldsymbol{s}_j\boldsymbol{s}_i^T + (\boldsymbol{s}_i^T\boldsymbol{s}_i - d_i^2)\mathbf{I}\right)\right)\boldsymbol{x} \tag{18}$$

$$- \sum_{i=1}^{m}\sum_{j=1}^{m} w_{ij}(\boldsymbol{s}_i^T\boldsymbol{s}_i - d_i^2)\boldsymbol{s}_j, \tag{19}$$

where in (15) we exploited the symmetry of $\mathbf{W}$.

To solve $\nabla h(\boldsymbol{x}) = 0$ we will simplify the gradient expression in three ways. First, note that the cost function is homogeneous in w_{ij}, and we can thus w.l.o.g. assume that $\sum_{ij} w_{ij} = 1$. This removes the coefficient of the third order term. Second, similar to [16], by applying the translation $t = \sum_{ij} w_{ij}\boldsymbol{s}_i$ to the senders, i.e., $\boldsymbol{s}_j \leftarrow \boldsymbol{s}_j - \boldsymbol{t}$, we ensure that $\sum_{ij} w_{ij}\boldsymbol{s}_j = 0$, canceling the second order term. Note that this does not change the problem as long as any solution $\boldsymbol{x}$ is translated back accordingly. The gradient can now be written as

$$\nabla h(\boldsymbol{x}) = (\boldsymbol{x}^T\boldsymbol{x})\boldsymbol{x} - \mathbf{A}\boldsymbol{x} + \boldsymbol{g}. \tag{20}$$

Third, note that $\mathbf{A}$ is real and symmetric and can thus be diagonalized by an orthogonal matrix $\mathbf{Q}$. Letting $\boldsymbol{y} = \mathbf{Q}^T\boldsymbol{x}$, we construct the new cost function $f(\boldsymbol{y}) = h(\mathbf{Q}\boldsymbol{y})$. Note that, minimizing $f(\boldsymbol{y})$ is equivalent to minimizing $h(\boldsymbol{x})$ where the senders have been rotated using $\mathbf{Q}$. The gradient of $f(\boldsymbol{y})$ now becomes

$$\nabla f(\boldsymbol{y}) = (\boldsymbol{y}^T\boldsymbol{y})\boldsymbol{y} - \mathbf{D}\boldsymbol{y} + \boldsymbol{b}, \tag{21}$$

where $\mathbf{D} = \mathbf{Q}^T\mathbf{A}\mathbf{Q}$ and $\boldsymbol{b} = \mathbf{Q}^T\boldsymbol{g}$. Furthermore, let the elements of $\mathbf{D}$ be sorted such that $D_{11} \geq D_{22} \geq \cdots \geq D_{nn}$.

Solving for the stationary points is equivalent to solving the n equations

$$(\boldsymbol{y}^T\boldsymbol{y})y_k - D_{kk}y_k + b_k = 0. \tag{22}$$

Multiplying the kth equation in (22) with y_k we get

$$(\boldsymbol{y}^T\boldsymbol{y})y_k^2 - D_{kk}y_k^2 + b_k y_k = 0, \tag{23}$$

which, naturally, are also satisfied at a stationary point. Letting $\boldsymbol{y}^2 = (y_1^2, y_2^2, \dots, y_n^2)^T$ denote the vector of squared coordinates, we use (23), (22) and the trivial equation $\boldsymbol{y}^T\boldsymbol{y} = \mathbf{1}^T\boldsymbol{y}^2$ to form the eigendecomposition

$$(\boldsymbol{y}^T\boldsymbol{y})\begin{pmatrix}\boldsymbol{y}^2\\ \boldsymbol{y}\\ 1\end{pmatrix} = \underbrace{\begin{pmatrix}\mathbf{D} & -\operatorname{diag}(\boldsymbol{b}) & \mathbf{0}\\ \mathbf{O} & \mathbf{D} & -\boldsymbol{b}\\ \mathbf{1}^T & \mathbf{0}^T & 0\end{pmatrix}}_{\triangleq \mathbf{M}}\begin{pmatrix}\boldsymbol{y}^2\\ \boldsymbol{y}\\ 1\end{pmatrix}. \tag{24}$$

Note that the matrix $\mathbf{M}$ does not depend on $\boldsymbol{y}$, and any stationary point of $f(\boldsymbol{y})$ corresponds to an eigenpair of $\mathbf{M}$ on the form in (24). In the following section, we will show that, under a small assumption, the converse is also true, i.e., every eigenvalue of $\mathbf{M}$ maps to a set of stationary points of $f(\boldsymbol{y})$.

3.1 Finding All Stationary Points

It is not necessarily the case that any eigenpair of $\mathbf{M}$ is on the form in (24). However, to find all stationary points it is sufficient to, for each eigenvalue, find all corresponding eigenvectors on the form in (24). The approach for doing this can be divided into two cases, depending on whether $\lambda\mathbf{I} - \mathbf{D}$ is singular or not.

Proposition 3.1. *If λ is an eigenvalue of $\mathbf{M}$ and $\lambda\mathbf{I}-\mathbf{D}$ has full rank, then $\boldsymbol{y} = -(\lambda I - \mathbf{D})^{-1}\boldsymbol{b}$ is the unique stationary point of $f(\boldsymbol{y})$ satisfying $\lambda = \boldsymbol{y}^T\boldsymbol{y}$.*

Proof. Let $\boldsymbol{v} = (\boldsymbol{v}_1^T, \boldsymbol{v}_2^T, v_3)^T$ be an eigenvector corresponding to λ. If $v_3 = 0$, then $(\lambda\mathbf{I} - \mathbf{D})\boldsymbol{v}_2 = 0 \implies \boldsymbol{v}_2 = 0 \implies \boldsymbol{v}_1 = 0 \implies \boldsymbol{v} = 0$, a contradiction. Consequently, we can w.l.o.g. assume $v_3 = 1$. Then $\boldsymbol{v}_2 = -(\lambda I - \mathbf{D})^{-1}\boldsymbol{b}$ and $\boldsymbol{v}_1 = -(\lambda\mathbf{I} - \mathbf{D})^{-1}\operatorname{diag}(\boldsymbol{b})\boldsymbol{v}_2 = \boldsymbol{v}_2^2$. From the last row of $\mathbf{M}$ we get $\lambda = \boldsymbol{v}_2^T\boldsymbol{v}_2$, and the eigenpair is on the form in (24). By the second row of $\mathbf{M}$, $\boldsymbol{v}_2$ is a stationary point. It is clear that $\boldsymbol{v}$ is the only eigenvector associated with λ and $\boldsymbol{v}_2$ is unique. □

The case when $\lambda\mathbf{I} - \mathbf{D}$ is singular we denote as a *degenerate case*. As we will see, this corresponds to when the trilateration problem is under-defined and has multiple, possibly infinitely many, solutions. Let $\mathbf{A}^+$ denote the Moore-Penrose pseudo inverse of $\mathbf{A}$.

Proposition 3.2. *If λ is an eigenvalue of $\mathbf{M}$, $\lambda\mathbf{I} - \mathbf{D}$ is singular, and $(\lambda\mathbf{I} - \mathbf{D})\boldsymbol{y} = -\boldsymbol{b}$ has a solution, then $\boldsymbol{y} = \boldsymbol{y}_p + \boldsymbol{y}_h$ is a stationary point, where $\boldsymbol{y}_p = -(\lambda\mathbf{I} - \mathbf{D})^+\boldsymbol{b}$ and $\boldsymbol{y}_h \in \ker(\lambda\mathbf{I} - \mathbf{D})$ such that $\boldsymbol{y}_h^T\boldsymbol{y}_h = \lambda - \boldsymbol{y}_p^T\boldsymbol{y}_p$.*

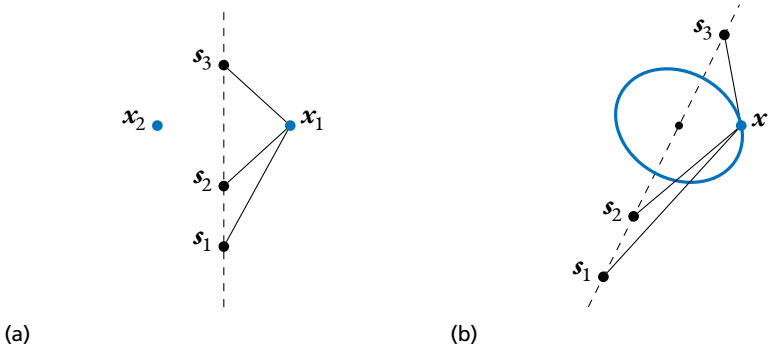


Figure 2: Two degenerate sender configurations. (a) When the senders are collinear in 2D, there are two possible solutions, $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$. (b) When the senders are collinear in 3D, there are infinitely many solutions, all located on a circle.

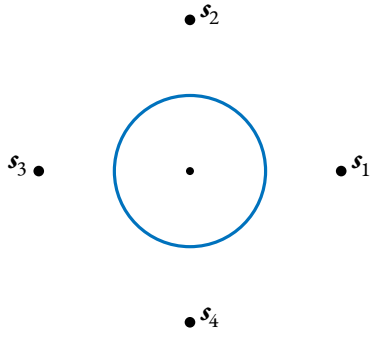


Figure 3: Degenerate case in the plane occurring when the senders are evenly distributed on the unit circle and the distance measurements are $d_i = \sqrt{5/2}$ for $i = 1, \ldots, 4$. The solutions consist of the blue circle.

Proof. The proposition is an obvious observation. If there is a $\boldsymbol{y}$ satisfying $(\lambda \mathbf{I} - \mathbf{D})\boldsymbol{y} = -\boldsymbol{b}$ and $\lambda = \boldsymbol{y}^T\boldsymbol{y}$, then it clearly satisfies $\nabla f(\boldsymbol{y}) = 0$ and is a stationary point. Due to the properties of the pseudo inverse, $\boldsymbol{y}_p^T\boldsymbol{y}_h = 0$ and $\lambda = \boldsymbol{y}^T\boldsymbol{y} = \boldsymbol{y}_p^T\boldsymbol{y}_p + \boldsymbol{y}_h^T\boldsymbol{y}_h$. □

The geometric interpretation of Proposition 3.2 is that, provided λ is real and $\lambda - \boldsymbol{y}_p^T\boldsymbol{y}_p \geq 0$, the stationary points satisfying $\lambda = \boldsymbol{y}^T\boldsymbol{y}$ consists of a hypersphere centered at $\boldsymbol{y}_p$ with radius $\sqrt{\lambda - \boldsymbol{y}_p^T\boldsymbol{y}_p}$. In particular, when $\lambda\mathbf{I} - \mathbf{D}$ has rank $n-1$ there is still a finite number of solutions (two). Figure 2 shows two degenerate cases resulting from the senders not spanning the space. A less intuitive example of a degenerate case is shown in Figure 3, where the senders indeed do span the plane.

3.2 Finding the Global Minimizer

The (real) global minimizer of $f(\boldsymbol{y})$ can be found by enumerating all stationary points and evaluating the cost. However, this is unnecessary as the following two propositions show that the global minimum corresponds to the largest real eigenvalue of $\mathbf{M}$.

Proposition 3.3. *The point $\boldsymbol{y}$ is a global minimizer of $f(\boldsymbol{y})$ if and only if $\nabla f(\boldsymbol{y}) = 0$ and $\lambda = \boldsymbol{y}^T\boldsymbol{y} \geq D_{11}$. If $\lambda > D_{11}$ then $\boldsymbol{y}$ is the unique global minimizer of $f(\boldsymbol{y})$.*

Proof. Minimizing $f(\boldsymbol{y})$ is equivalent to the generalized trust region subproblem

$$\begin{aligned} \underset{\boldsymbol{y},\lambda}{\text{minimize}} \quad & \frac{1}{4}\lambda^2 - \frac{1}{2}\boldsymbol{y}^T\mathbf{D}\boldsymbol{y} + \boldsymbol{b}^T\boldsymbol{y}, \\ \text{subject to} \quad & \frac{1}{2}(\boldsymbol{y}^T\boldsymbol{y} - \lambda) = 0, \end{aligned} \tag{25}$$

where the Laplacian is given by

$$\mathscr{L}(\boldsymbol{y}, \lambda, \nu) = \frac{1}{4}\lambda^2 - \frac{1}{2}\boldsymbol{y}^T\mathbf{D}\boldsymbol{y} + \boldsymbol{b}^T\boldsymbol{y} + \frac{1}{2}\nu(\boldsymbol{y}^T\boldsymbol{y} - \lambda), \tag{26}$$

and ν is the Laplacian multiplier. By [17, Theorem 3.2], $(\boldsymbol{y}, \lambda)$ is a global minimizer if and only if it for some ν satisfies

$$\nabla_{\boldsymbol{y}}\mathscr{L}(\boldsymbol{y}, \lambda, \nu) = -\mathbf{D}\boldsymbol{y} + \boldsymbol{b} + \nu\boldsymbol{y} = 0, \tag{27}$$

$$\nabla_{\lambda}\mathscr{L}(\boldsymbol{y}, \lambda, \nu) = \frac{1}{2}(\lambda - \nu) = 0, \tag{28}$$

$$\nabla^2_{\boldsymbol{y},\lambda}\mathscr{L}(\boldsymbol{y}, \lambda, \nu) \succeq 0 \quad \Leftrightarrow \quad \nu\mathbf{I} - \mathbf{D} \succeq 0, \tag{29}$$

and $\lambda = \boldsymbol{y}^T\boldsymbol{y}$, which is equivalent to $\nabla f(\boldsymbol{y}) = 0$ and $\lambda\mathbf{I} - \mathbf{D} \succeq 0$. Uniqueness of the global minimizer is given by [17, Theorem 4.1] when $\lambda\mathbf{I} - \mathbf{D} \succ 0$. □

Proposition 3.4. *For any global minimizer $\boldsymbol{y}$ of $f(\boldsymbol{y})$, we have $\lambda_{max} = \boldsymbol{y}^T\boldsymbol{y}$, where λ_{max} is the largest real eigenvalue of $\mathbf{M}$.*

Proof. From the definition of $h(\boldsymbol{x})$, a global minimizer must exist, and, consequently, $\lambda_{\max} \geq D_{11}$ by Proposition 3.3.

If $\lambda_{\max} > D_{11}$, then $\boldsymbol{y} = -(\lambda_{\max}\mathbf{I} - \mathbf{D})^{-1}\boldsymbol{b}$ is a stationary point satisfying $\lambda_{\max} = \boldsymbol{y}^T\boldsymbol{y}$ by Proposition 3.1 and a unique global minimizer by Proposition 3.3.

If $\lambda_{\max} = D_{11}$, then clearly any global minimizer $\boldsymbol{y}$ satisfies $\lambda_{\max} = \boldsymbol{y}^T\boldsymbol{y}$, or there would exist a real eigenvalue larger than $\lambda_{\max}$. □

When solving trilateration problems we are often working in 2D or 3D space, and $\mathbf{M}$ will have size 5×5 or 7×7, respectively. For such small problems it is not expensive to simply calculate all eigenvalues, although Proposition 3.4 has shown that we are only interested in the largest real eigenvalue. Nevertheless, there are efficient algorithms for finding the eigenvalue with the largest real part, i.e., the rightmost eigenvalue [18, 19], but for these to be applicable, we need to first show that the largest real eigenvalue also is the rightmost one.

Proposition 3.5 (cf. [20][Theorem 3.4]). *The rightmost eigenvalue of* $\mathbf{M}$ *is real.*

Proof. Assume for contradiction that $\lambda = \alpha + \beta i$ is the rightmost eigenvalue of $\mathbf{M}$, where $\alpha, \beta \in \mathbb{R}$, $\alpha \geq D_{11}$, $\beta \neq 0$ and i is the imaginary unit. Since $\lambda = \alpha - \beta i$ also is an eigenvalue, we can assume $\beta > 0$. Then $\lambda I - \mathbf{D}$ has full rank and by Proposition 3.1

$$\lambda - \sum_{k=1}^{n} \frac{b_k^2}{(\lambda - D_{kk})^2} = \lambda - \boldsymbol{y}^T \boldsymbol{y} = 0. \tag{30}$$

However, $\mathrm{Im}(b_k^2/(\lambda - D_{kk})^2) \leq 0$ for $k = 1, \ldots, n$, implying $\mathrm{Im}(\lambda - \boldsymbol{y}^T \boldsymbol{y}) > 0$. This is a contradiction, and λ cannot be an eigenvalue of $\mathbf{M}$. □

3.3 Alternative Eigendecomposition

In the previous sections, we have worked with the matrix $\mathbf{D}$ found by diagonalizing $\mathbf{A}$. This diagonalization step can be mitigated by instead of $\mathbf{M}$ considering the eigendecomposition of

$$\mathbf{M}_A = \begin{pmatrix} \mathbf{A} & \mathbf{I} & \mathbf{0} \\ \mathbf{O} & \mathbf{A} & -\boldsymbol{g} \\ -\boldsymbol{g}^T & \mathbf{0}^T & 0 \end{pmatrix}. \tag{31}$$

This matrix is similar to $\mathbf{M}$ with the change of basis matrix

$$\mathbf{P} = \begin{pmatrix} -\operatorname{diag}(\boldsymbol{b})\mathbf{Q}^T & & \\ & \mathbf{Q}^T & \\ & & 1 \end{pmatrix}, \tag{32}$$

i.e., $\mathbf{M}_A = \mathbf{P}^{-1}\mathbf{M}\mathbf{P}$. Consequently, $\mathbf{M}$ and $\mathbf{M}_A$ have the same eigenvalues whenever $\operatorname{diag}(\boldsymbol{b})$ has full rank. Assuming this is the case also when $\operatorname{diag}(\boldsymbol{b})$ is singular, the propositions from the previous sections naturally transfer to this new formulation. In particular, if $\lambda_{\max}$ is the largest real eigenvalue of $\mathbf{M}_A$ and $\lambda_{\max}\mathbf{I} - \mathbf{A}$ has full rank, the global minimizer of $h(\boldsymbol{x})$ is given by $\boldsymbol{x} = -(\lambda_{\max}\mathbf{I} - \mathbf{A})^{-1}\boldsymbol{g}$.

Algorithm 1 Simple Trilateration

Input: Sender positions $\boldsymbol{s}_j$, distances d_j, weights w_{ij}
Output: Receiver position $\boldsymbol{x}$

1: Normalize weights: $w_{ij} \leftarrow w_{ij} / \sum_{ij} w_{ij}$.
2: Translate senders: $\boldsymbol{s}_j \leftarrow \boldsymbol{s}_j - \boldsymbol{t}$, where $\boldsymbol{t} = \sum_{ij} w_{ij} \boldsymbol{s}_i$.
3: Find the largest real eigenvalue $\lambda_{\max}$ of $\mathbf{M}_A$ (31).
4: Find receiver position as $\boldsymbol{x} = -(\lambda_{\max}\mathbf{I} - \mathbf{A})^{-1}\boldsymbol{g}$.
5: Undo translation: $\boldsymbol{x} \leftarrow \boldsymbol{x} + \boldsymbol{t}$.

It is worth noting that $\mathbf{M}_A$ closely resembles the matrix constructed in [20] for solving the Trust-Region Subproblem (TRS). However, the problem considered here does not belong to that class of problems, and as such, their method is not directly applicable. In a later work [21], the method proposed in [20] was extended to the GTRS. However, applying their method would require the introduction of an additional unknown corresponding to α in (4) (cnf. the SR-LS approach in [22]) and result in a $2n + 3 \times 2n + 3$ generalized eigenvalue problem, as appose to the here proposed $2n + 1 \times 2n + 1$ ordinary eigenvalue problem.

3.4 The Proposed Algorithm

The results from the previous sections now admit an algorithm for finding the global minimizer of $h(\boldsymbol{x})$. The simplest version of the proposed method is shown in Algorithm 1, but we will offer a few improvements resulting in the proposed method in Algorithm 2.

While the formulation using $\mathbf{M}_A$ in Section 3.3 avoids the eigendecomposition of $\mathbf{A}$, explicitly calculating $\mathbf{D}$ simplifies calculations related to the degenerate cases, e.g., it is trivial to find the rank, kernel and Moore-Penrose pseudo inverse of $\lambda\mathbf{I}-\mathbf{D}$. Furthermore, in our implementation, we found that the eigendecomposition of $\mathbf{M}$ actually is faster to calculate than that of $\mathbf{M}_A$, resulting in an overall faster solver.

Another issue with Algorithm 1 is that it does not handle the degenerate cases. When $\operatorname{rank}(\lambda\mathbf{I} - \mathbf{D}) < n - 1$, there are an infinite number of solutions, and the problem is ill-defined. However, when $\operatorname{rank}(\lambda\mathbf{I} - \mathbf{D}) = n - 1$, there are two solutions given by (see

Algorithm 2 Trilateration

Input: Sender positions $\boldsymbol{s}_j$, distances d_j, weights w_{ij}
Output: Receiver position $\boldsymbol{x}$

1: Normalize weights: $w_{ij} \leftarrow w_{ij} / \sum_{ij} w_{ij}$.
2: Translate senders: $\boldsymbol{s}_j \leftarrow \boldsymbol{s}_j - \boldsymbol{t}$, where $\boldsymbol{t} = \sum_{ij} w_{ij} \boldsymbol{s}_i$.
3: Diagonalize $\mathbf{A} = \mathbf{Q}\mathbf{D}\mathbf{Q}^T$ in (20).
4: Find the largest real eigenvalue $\lambda_{\max}$ of $\mathbf{M}$ (24).
5: **if** $\operatorname{rank}(\lambda_{\max}\mathbf{I} - \mathbf{D}) = n$ **then**
6: Solve for $\boldsymbol{y}$ using (33)-(34) and choose the sign in (34) such that $\operatorname{sgn}(y_1) = -\operatorname{sgn}(b_1)$.
7: **else if** $\operatorname{rank}(\lambda_{\max}\mathbf{I} - \mathbf{D}) = n - 1$ **then**
8: Solve for the two solutions $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ using (33)-(34).
9: **else**
10: The problem is ill-defined. Return nothing.
11: **end if**
12: Undo rotation: $\boldsymbol{x} = \mathbf{Q}\boldsymbol{y}$.
13: Undo translation: $\boldsymbol{x} \leftarrow \boldsymbol{x} + \boldsymbol{t}$.

Proposition 3.2)

$$y_k = -\frac{b_k}{\lambda - D_{kk}} \quad \text{for} \quad k = 2, \ldots, n, \tag{33}$$

$$y_1 = \pm\sqrt{\lambda - \sum_{k=2}^{n} y_k^2}. \tag{34}$$

If $\lambda\mathbf{I} - \mathbf{D}$ has full rank, a single solution is given by Proposition 3.1. However, when approaching a degenerate case the numerical stability of $\boldsymbol{y} = -(\lambda\mathbf{I}-\mathbf{D})^{-1}\boldsymbol{b}$ gets worse. To avoid this, we always treat the problem as if the rank is $n-1$ and use (33)-(34) also in the nondegenerate case. There is still only one correct solution though, so we choose the sign in (34) such that $\operatorname{sgn}(y_1) = -\operatorname{sgn}(b_1)$. This approach yields better numerical stability when transitioning to and from degenerate cases. The improved proposed method is summarized in Algorithm 2. Note that the rank check on $\lambda\mathbf{I}-\mathbf{D}$ is only used to determine whether one or two solutions should be returned.

Table 1: Execution speed of a number of trilateration methods from the literature.

Method	Execution time		
	$m = 4$	$m = 10$	$m = 100$
Zhou [8]	11 µs	11 µs	13 µs
Linear	1.3 µs	1.4 µs	4.7 µs
Beck SDR [6]	4.8 ms	14 ms	46 s
Adachi [21]	45 µs	45 µs	47 µs
Beck SFP [2]	490 µs	300 µs	2000 µs
Luke [1]	480 µs	250 µs	610 µs
Beck SR-LS [6]	31 µs	36 µs	43 µs
Proposed (Alg. 1)	26 µs	27 µs	30 µs
Proposed (Alg. 2)	18 µs	18 µs	21 µs

4 Experiments with Synthetic Data

In this section, we compare the proposed methods in Algorithms 1 and 2 with a number of other methods from the literature [1, 8, 22]. In addition to these, we include a linear method which is the unconstrained version of (4), and one method solving (4) using the generalized eigenvalue decomposition presented in [21]. To solve the SDP in [22] we used Hypatia [23]. All methods have been implemented in Julia [24]. It should be noted that some of these methods minimize the ML cost (2) and some minimize $h(\boldsymbol{x})$ with the weights $w_{ii} = 1/4d_i^2$, $w_{ij} = 0$ for $i \neq j$, as proposed in Section 2.1. However, Zhou [8] does not (trivially) allow for a similar weighting and is consequently minimizing (3).

4.1 Execution Speed

First, we compare the execution time of the different methods. This is not trivial as several of the methods are iterative with various termination criteria. Nevertheless, we attempted to match these where possible. Synthetic data was generated using random receiver and sender positions in 3D space; $\boldsymbol{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\boldsymbol{s}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ for $i = 1, \ldots, m$. The distance measurements were calculated without any added noise. A benchmark was setup where each solver was run 10 000 times or for a total maximum of 10 seconds, each time with different synthetic data. The benchmark was run on a AMD Ryzen Threadripper 3990X, and the resulting median run times are shown in Table 1 for $m = 4, 10, 100$.

As can be seen, the proposed method is competitive with regards to execution time. Note

also that, as mentioned before, Algorithm 1 is not faster than Algorithm 2 even though it avoids the eigendecomposition of $\mathbf{A}$. Luke [1] and Beck SFP [2] are slower when $m = 4$ compared to $m = 10$. A possible explanation is that these iterative methods converge slower for near-degenerate cases, which are more likely to occur when $m = 4$. They also become slower when $m = 100$, which is expected as they use all sender positions in their inner iterations. In contrast to this, the proposed method quickly reduces the data to $\mathbf{A} \in \mathbb{R}^{n \times n}$ and $\boldsymbol{g} \in \mathbb{R}^n$, where the sizes only depend on the spatial dimension n and not the number of senders m. Consequently, the proposed method and some of the others maintain an almost constant execution time over the number of senders. The slowest method is Beck SDR [6] by several orders of magnitude. This solver consists of a semidefinite relaxation where the number of unknowns scales quadratically with the number of senders.

4.2 Gaussian Noise

To evaluate the proposed method over various amounts of noise, a large set of synthetic datasets was constructed. The positions for $m = 10$ senders and a single receiver in 3D space were sampled from the standard normal distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$, after which distances measurements were calculated and subsequently perturbed by Gaussian noise with standard deviation σ. A total of 1000 datasets were constructed.

Figure 4 shows the relative errors in the estimated receiver positions for a range of trilateration methods. The maximum likelihood (ML) estimate, found using local optimization of (2) initialized at the ground truth receiver position, is included for reference. As can be seen, most of the methods performed fairly similarly. However, Zhou [8] and the linear method perform worse. Two possible reasons for Zhou performing worse is: (i) it does not use the weighting proposed in Section 2.1, and (ii) it assumes $\sum_j \|\boldsymbol{x} - \boldsymbol{s}_j\|^2 = \sum_j d_j^2$ which is generally not true in the presence of noise.

4.3 Degenerate Configurations

Some of the methods in the literature do not handle degenerate cases and become numerically unstable as we approach such a scenario. Furthermore, in situations similar to that in Figure 2a, there are two possible solutions to the trilateration problem. Both Algorithm 2 and Zhou [8] return two solutions in this case, while the remaining methods, if successful, only return one solution.

To investigate the numerical stability of the methods close to a degenerate case, we generate $m = 6$ senders $\boldsymbol{s}_j \in \mathbb{R}^n$ for $j = 1, \ldots, 6$ and a single receiver $\boldsymbol{x} \in \mathbb{R}^n$ with coor-

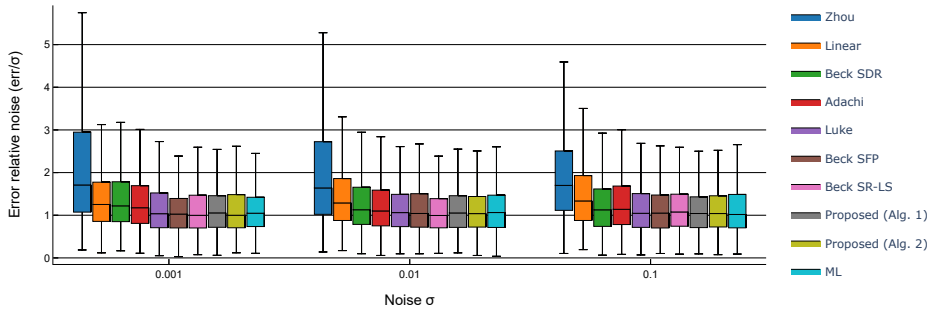


Figure 4: Relative errors in estimated receiver position for a selection of trilateration methods over various amounts of noise. The whiskers in the boxplot indicate the 1.5 IQR value and outliers are not shown.

dinates sampled from $\mathcal{N}(0, 1)$. The distance measurements are then calculated without any added noise. Finally, the x-coordinate of each sender position is multiplied with a scaling factor. As this scaling factor approaches zero the senders become coplanar and a degenerate case occurs.

Figure 5 shows the median error the in estimated receiver position over 1000 trials and a range of scaling factors. If a method returns two solutions, the solution with the smallest error is used. As can be seen, Algorithm 2 yields errors close to machine precision over the whole range of scaling factors. Due to the use of (33) and (34) also in the nondegenerate case, there is no obvious transition where the rank of $\lambda_{\max}\mathbf{I} - \mathbf{D}$ changes.

All of the other methods perform notably worse than Algorithm 2. In particular, this experiment highlights a flaw in Zhou [8]. In general, the method would actually perform similarly to Algorithm 2 for this type of scenario. Unfortunately, it is sensitive specifically to the scaling in the x-coodinates done here.

5 Experiments with Real Data

The aim of this section is to evaluate the proposed method using real data gathered from an Ultra-Wideband (UWB) setup. Six senders were kept stationary in a 2 m × 2 m × 1 m volume, while a single receiver was moved through the setup. TOA distance measurements were recorded using UWB, while ground-truth positions for the senders and receiver were recorded using an optical motion capture system. In total, eight datasets were

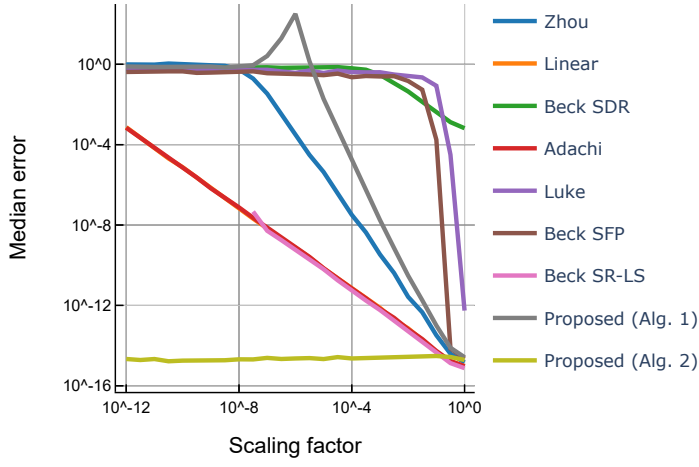


Figure 5: Median error in estimated receiver position over 1000 trails and a range of scaling factors. The scaling factor is multiplied with the x-coordinate of each sender causing them to become coplanar as the factor approaches zero. The linear method overlaps with Adachi and Beck SR-LS.

gathered, each containing between 358 and 934 receiver positions which were estimated using the proposed Algorithm 2, Zhou [16], Luke [25], the SR-LS solver from [22], and the SDR solver from [22]. For comparison, the ML estimate was found by performing local optimization on (2), initialized at the ground truth.

Table 2 shows RMS errors in the estimated receiver positions for the eight datasets and the compared trilateration methods, where the minimum error for each dataset is marked in boldface. As can be seen, the proposed method achieves the minimum error for all datasets. The fact that the ML estimate does not produce the lowest errors indicates that the assumption of Gaussian noise in the measurements is not completely justified for the data used here. As it happens, the approximated cost function in the proposed method provides a better model for the noise distribution and yields lower RMS errors.

6 Conclusion

In this paper, we have presented a novel method for performing trilateration by formulating the problem as an eigendecomposition problem. This enables the usage of

Table 2: RMS errors (meters) in receiver position for eight datasets based on real UWB data.

Data-set	Zhou [16]	Luke [25]	SR-LS [22]	SDR [22]	ML	Proposed Alg. 2
1	0.66	0.34	0.40	0.34	0.34	**0.32**
2	0.65	**0.52**	0.54	**0.52**	**0.52**	**0.52**
3	11.05	0.64	1.18	0.64	0.64	**0.42**
4	0.54	0.31	0.44	0.32	0.31	**0.30**
5	0.64	0.34	0.41	0.34	0.34	**0.33**
6	0.47	0.31	0.37	0.31	0.31	**0.28**
7	0.53	0.32	0.35	0.32	0.32	**0.31**
8	0.74	0.39	0.48	0.39	0.39	**0.36**

existing well-developed eigensolvers, resulting in algorithms that are both fast and easy to implement. We have also made theoretical contributions by showing that the global minimum of the cost function corresponds to the largest real eigenvalue. Furthermore, unlike previous works, we have treated the degenerate and near-degenerate cases, resulting in a more robust solution. This can be especially useful when the number of senders are few or measurements are sparse.

Through experiments on synthetic data, we have verified that the weighted cost function used closely approximates the ML estimator in (2). We have also demonstrated that the proposed approach remains numerically stable when approaching degenerate cases. Finally, the practical application of the proposed method has been demonstrated using real data from UWB.

In summary, the proposed method improves on existing ones in speed, robustness and ease of implementation, making it a competitive approach for performing trilateration.

References

[1] D. R. Luke, S. Sabach, M. Teboulle, and K. Zatlawey, "A simple globally convergent algorithm for the nonsmooth nonconvex single source localization problem," *Journal of Global Optimization*, vol. 69, no. 4, pp. 889–909, Dec. 2017. [Online]. Available: https://doi.org/10.1007/s10898-017-0545-6

[2] A. Beck, M. Teboulle, and Z. Chikishev, "Iterative Minimization Schemes for Solving the Single Source Localization Problem," *SIAM Journal on Op-*

timization, vol. 19, no. 3, pp. 1397–1416, Jan. 2008. [Online]. Available: http://epubs.siam.org/doi/10.1137/070698014

[3] R. Jyothi and P. Babu, "SOLVIT: A Reference-Free Source Localization Technique Using Majorization Minimization," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2661–2673, 2020, conference Name: IEEE/ACM Transactions on Audio, Speech, and Language Processing.

[4] N. Sirola, "Closed-form algorithms in mobile positioning: Myths and misconceptions," in *Navigation and Communication 2010 7th Workshop on Positioning*, Mar. 2010, pp. 38–44.

[5] Y.-T. Chan, H. Yau Chin Hang, and P.-c. Ching, "Exact and approximate maximum likelihood localization algorithms," *IEEE Transactions on Vehicular Technology*, vol. 55, no. 1, pp. 10–16, Jan. 2006, conference Name: IEEE Transactions on Vehicular Technology.

[6] A. Beck, P. Stoica, and J. Li, "Exact and Approximate Solutions of Source Localization Problems," *IEEE Transactions on Signal Processing*, vol. 56, no. 5, pp. 1770–1778, May 2008, conference Name: IEEE Transactions on Signal Processing.

[7] K. W. Cheung, H. C. So, W. Ma, and Y. T. Chan, "Least squares algorithms for time-of-arrival-based mobile location," *IEEE Transactions on Signal Processing*, vol. 52, no. 4, pp. 1121–1130, Apr. 2004, conference Name: IEEE Transactions on Signal Processing.

[8] Y. Zhou, "A closed-form algorithm for the least-squares trilateration problem," *Robotica*, vol. 29, no. 3, pp. 375–389, May 2011, publisher: Cambridge University Press.

[9] F. Thomas and L. Ros, "Revisiting trilateration for robot localization," *IEEE Transactions on Robotics*, vol. 21, no. 1, pp. 93–101, Feb. 2005, conference Name: IEEE Transactions on Robotics.

[10] D. E. Manolakis, "Efficient solution and performance analysis of 3-D position estimation by trilateration," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, no. 4, pp. 1239–1248, Oct. 1996, conference Name: IEEE Transactions on Aerospace and Electronic Systems.

[11] I. D. Coope, "Reliable computation of the points of intersection of n spheres in R^n," *ANZIAM Journal*, vol. 42, pp. C461–C477, Dec. 2000. [Online]. Available: https://journal.austms.org.au/ojs/index.php/ANZIAMJ/article/view/608

[12] P. Stoica and J. Li, "Lecture Notes - Source Localization from Range-Difference Measurements," *IEEE Signal Processing Magazine*, vol. 23, no. 6, pp. 63–66, Nov. 2006, conference Name: IEEE Signal Processing Magazine.

[13] M. Larsson, V. Larsson, K. Åström, and M. Oskarsson, "Optimal Trilateration Is an Eigenvalue Problem," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2019, pp. 5586–5590, iSSN: 2379-190X.

[14] I. Sharp and K. Yu, *Wireless Positioning: Principles and Practice*, ser. Navigation: Science and Technology. Singapore: Springer, 2019. [Online]. Available: http://link.springer.com/10.1007/978-981-10-8791-2

[15] E. Commission, D.-G. for the Information Society, and Media, *COST Action 231 : Digital mobile radio towards future generation systems: Final Report*. Publications Office, 1999.

[16] Y. Zhou, "A closed-form algorithm for the least-squares trilateration problem," *Robotica*, vol. 29, no. 3, pp. 375–389, 2011.

[17] J. J. More, "Generalizations of the trust region problem," *Optimization Methods and Software*, vol. 2, no. 3-4, pp. 189–209, Jan. 1993. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/10556789308805542

[18] W. E. Arnoldi, "The principle of minimized iterations in the solution of the matrix eigenvalue problem," *Quarterly of Applied Mathematics*, vol. 9, no. 1, pp. 17–29, 1951. [Online]. Available: https://www.ams.org/qam/1951-09-01/S0033-569X-1951-42792-9/

[19] R. B. Lehoucq, D. C. Sorensen, and C. Yang, *ARPACK Users' Guide*. Society for Industrial and Applied Mathematics, 1998, _eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9780898719628. [Online]. Available: https://epubs.siam.org/doi/abs/10.1137/1.9780898719628

[20] S. Adachi, S. Iwata, Y. Nakatsukasa, and A. Takeda, "Solving the Trust-Region Subproblem By a Generalized Eigenvalue Problem," *SIAM Journal on Optimization*, vol. 27, no. 1, pp. 269–291, Jan. 2017. [Online]. Available: http://epubs.siam.org/doi/10.1137/16M1058200

[21] S. Adachi and Y. Nakatsukasa, "Eigenvalue-based algorithm and analysis for nonconvex QCQP with one constraint," *Mathematical Programming*, vol. 173, no. 1, pp. 79–116, Jan. 2019. [Online]. Available: https://doi.org/10.1007/s10107-017-1206-8

[22] A. Beck, P. Stoica, and J. Li, "Exact and approximate solutions of source localization problems," *IEEE Transactions on signal processing*, vol. 56, no. 5, pp. 1770–1778, 2008.

[23] C. Coey, L. Kapelevich, and J. P. Vielma, "Solving natural conic formulations with hypatia.jl," 2021.

[24] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, "Julia: A fresh approach to numerical computing," *SIAM Review*, vol. 59, no. 1, pp. 65–98, 2017. [Online]. Available: https://doi.org/10.1137/141000671

[25] D. R. Luke, S. Sabach, M. Teboulle, and K. Zatlawey, "A simple globally convergent algorithm for the nonsmooth nonconvex single source localization problem," *Journal of Global Optimization*, vol. 69, no. 4, pp. 889–909, 2017.

This page was intentionally left blank solely for this thesis to reach exactly 200 pages.

Printed by Media-Tryck, Lund 2022 NORDIC SWAN ECOLABEL 3041 0903

Doctoral Theses in Mathematical Sciences 2022:9

ISBN 978-91-8039-377-5
ISSN 1404-0034