Multi-Agent Relative Pose Estimation with Ultra-Wideband Ranging

by

Andrew Fishberg

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Master of Science in Aeronautics and Astronautics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2024

Author .....
Department of Aeronautics and Astronautics
January 12, 2024

Certified by .....
Jonathan P. How
R. C. Maclaurin Professor of Aeronautics and Astronautics
Thesis Supervisor

Accepted by .....
Jonathan P. How
R. C. Maclaurin Professor of Aeronautics and Astronautics
Chair, Graduate Program Committee

Multi-Agent Relative Pose Estimation with Ultra-Wideband Ranging

by

Andrew Fishberg

Submitted to the Department of Aeronautics and Astronautics
on January 12, 2024, in partial fulfillment of the
requirements for the degree of
Master of Science in Aeronautics and Astronautics

Abstract

Inter-agent relative localization is critical for any multi-robot system operating in the absence of external positioning infrastructure or prior environmental knowledge. Motivated by the applications of nuclear non-proliferation, radiological search, and radiological mapping, this thesis explores leveraging multiple ultra-wideband (UWB) ranging sensors to produce frequent inter-agent pose estimates with minimal communication overhead. This work is intended as a component of a larger multi-agent simultaneous localization and mapping (SLAM) system (also known as collaborative SLAM or CSLAM), where persistent UWB-based inter-agent pose estimates provide a valuable alternative source of inter-agent loop closures. By collecting and analyzing real data, we develop improved sensor models, which in turn inform our algorithm design process – thus, this work produces competitive or improved results to state-of-the-art approaches with significantly less overall communication. By comparison, prior work typically supplements noisy UWB range measurements with additional *continuously* transmitted data, such as odometry, leading to potential scaling issues with increased team size and/or decreased communication network capability.

This thesis's main technical contributions are as follows: (1) Exploration of current commercially available off-the-shelf (COTS) UWB devices for use in mobile robotics. By analyzing real data, insights into commonly overlooked sensor quirks are addressed through our improved sensor models. (2) Development and testing of a novel 2D relative pose estimation system based on trilateration, leveraging multiple UWB ranging sensors per agent. (3) Extension of said system to 3D environments. (4) A list of recommendations and continuations for future work.

Thesis Supervisor: Jonathan P. How
Title: R. C. Maclaurin Professor of Aeronautics and Astronautics

Acknowledgments

My deepest gratitude to the countless individuals who made this thesis possible:

To my advisor, Jonathan P. How: Thank you for your mentorship, patience, and irreplaceable guidance in the preparation of this thesis. You are the gold standard of hard work and mastery of your craft – I hope to emulate this within my own career.

To Sertac Karaman & Vivienne Sze: Thank you for launching my MIT graduate school journey. Under your tutelage in LEAN, I was given the freedom to learn and explore – I would not be on this path without you.

To Luca Carlone & John Leonard: Thank you for the fruitful opportunities to collaborate with you and your students. I am eager to see what our continued partnerships will produce.

To Brian Quiter & Emil Rofors: Thank you for hosting me at Lawrence Berkeley National Laboratory, introducing me to the world of nuclear non-proliferation, and providing invaluable assistance in collecting outdoor flight data.

To the Enabling Technologies & Innovation (ETI) consortium[1], especially Anna Erickson, Yuguo Tao, Samuel Kemp, and Mackenzie Duce: Thank you for making the semi-annual ETI meetings rewarding events that I look forward to each semester. ETI would not be the same without administrators and fellow students like you.

To Nicolas Roy, Julie Shah, John Fisher, Michael Short, Justin Solomon, and Micah Altman: Thank you for taking the time to discuss my future career prospects and research paths. Our discussions have had a profound lasting impression on my future trajectory.

To Alan Papalia: Thank you for your mentorship and the opportunity to collaborate on research. Being under your wing has made a big difference – I am excited to grow as a mathematician under your tutelage.

To all my Aerospace Controls Laboratory (ACL) labmates: Thank you for making the lab a memorable, exciting, and productive place to work. Special thanks to Yulun Tian for the indispensable one-on-one mentorship opportunities; to Kaveh Fathian for

[1]The work in this thesis was supported in part by DOE, NNSA, and ALB funding.

his generosity, mentorship, and friendship; to David Rosen, Michael Everett, Miguel Calvo-Fullana, Jesus Tordesillas Torres, Andrea Tagliabue, and Stewart Jamieson for valuable discussions, paper feedback, and overall advice; to Dong-Ki Kim for introducing me to the technical details of reinforcement learning; to Parker Lusk & Kota Kondo for their irreplaceable drone construction and hardware code technical support; to Bryt Bradley for being an administrative wizard; and to Andrew Torgesen for kick-starting the predecessor to this thesis. Thanks to all ACL members I overlapped with: Andrea Tagliabue, Andrew Torgesen, Annika Thomas, Bryt Bradley, Codrin Oneci, Connor Anderson, David Rosen, Dong-Ki Kim, Donggun Lee, Evelyn Ruff, Golnaz Habibi, Hannah Shafferman, Jacqueline Ankenbauer, Jeremy Cai, Jesus Tordesillas Torres, Jimmy Queeney, Kaveh Fathian, Kota Kondo, Lakshay Sharma, Lena Downes, Lucas Jia, Lucy Halperin, Mason Peterson, Michael Everett, Miguel Calvo-Fullana, Naman Aggarwal, Nick Rober, Parker Lusk, Stewart Jamieson, and Yulun Tian.

To Varun Murali, Igor Spasojevic, Peter Li, Soumya Sudhakar, and Samir Wadhwania: Thank you for your friendship and helpful advice.

To my colleagues at The Educational Justice Institute (TEJI), Lee Perlman, Carole Cafferty, Marisa Gaetz, Martin Nisser, Mary Ellen McGorry, Joshua Long, and Jane Abbott: Thank you for providing opportunities to teach within the American prison system. These eye-opening experiences and interactions have changed my life.

To my Teaching and Learning Lab (TLL) mentors, Ruthann Thomas, Raechel Soicher, and Benjamin Hansberry: Thank you for fostering my growth as a teacher and challenging me to find new and innovative ways to share my passions.

To Brian Mernoff, Jacqueline Goldstein, and the entire AeroAstro Communication Lab: Thank you for providing the opportunity to help others while growing as a communicator and educator.

To my UROPs: Thank you for all your hard work and fun times. I can only hope you have learned half as much from me as I have learned from mentoring all of you.

To my professional mentors at MIT Lincoln Laboratory: Thank you for providing a starry-eyed straight-out-of-undergrad programmer the opportunity to dive right

into world-class state-of-the-art robotics development. You sparked my research interest in multi-agent robotics and allowed me to flourish in an unparalleled nurturing environment. Special thanks to: Bob Shin, Scott Van Broekhoven, Shourov Chatterji, James Ward, Mike Boulet, Brian Tipton, Andrew Silberfarb, Ross Allen, Ryan Eubank, Sam Stambler, Aaron Plotnik, Andrew Kopeikin, Kenta Hood, Will Bartlett, and Mike Klinker.

To my Beaver Works colleagues, Joel Grimm, Lisa Kelley, Sabina Chen, Eyassu Shimelism, Chris Lai, and HMC Clinic Teams: Thank you for your stalwart work in creating a truly transformational educational experience. I will always fondly remember our experiences with the various MIT RACECARs.

To Brian Plancher: Thank you for always making time to be a mentor, friend, and all-around great person. Your dedication goes well above any obligation, and I appreciate it.

To Brian Teague & Alyssa Pierson: Thank you for all your advice and being a model post-Mudd couple for Sarah and me.

To Sean Huver: Thank you for providing me with mentorship, advice, and numerous career opportunities over the years. I really appreciate how we consistently reconnect throughout our dynamic paths.

To my Harvey Mudd College mentors, Zachary Dodds, Ran Libeskind-Hadas, Jessica Wu, and Colleen Lewis: Thank you for your unyielding passion and commitment to education that has molded me into the person I am today. Special thanks to Zachary Dodds for enabling my endless shenanigans over the years and being a once-in-a-generation professor and role model.

To James Gill, Carol Valosin, Rhonda Solomon, and Robert M. Kelly: Thank you for being among my first computer science teachers and mentors all the way back in high school. None of us may have realized it at the time, but you inspired me to go down this path.

To my personal friends: Thank you for all the fun and memorable times. Life is richer because of friends like you. In Boston: Andy Russell, Sarah Anthony, Max Comstock, Karishma Chadha, Nicole Francisco, Anjali Kayal, Devon Stork, Theia

Henderson, Govind Ramnarayan, and Taha Aziz. From Harvey Mudd College: Wai Sing Wong, Jennifer Rogers Brennan, Ross Mawhorter, Luis Viornery, Emma Meersman, Ben Lowenstein, Dani Bork, Hannah Young, Sam Posner, Chris Miro, and Justin Bai. From Freehold High School's Computer Science Academy: Alan Kahn, Bethann Polinksy, Cole Capobianco, Matt Siegel, Nolan Lum, Sangini Shah, Tad Cordle, Tae-Min Kim, and Will Marter.

To our family friends, Bill & Jane Jemas, AJ Jemas, and Jake Jemas: Thank you for being a consistent part of my entire life. You are *de facto* family.

To my parents, David & Carol Fishberg: Thank you for being the most dedicated and caring parents anyone could ever hope for. As I continue to venture further from the worlds of dentistry and taxes, your endless love and support is something I can always depend on.

To my sister and her partner, Erica Fishberg & Abraham Perez: Thank you for being an indispensable part of the family. No matter where life takes us, we know we can always count on each other.

To my grandmother, Marge Focarile: Thank you for being supportive and ahead of your time in so many ways. Because of you our family has come a long way in just a few short generations.

To all my other family members, past, present, and future: Thanks for always having my back. I will always be there for you.

To my future in-laws, Paul & Stephanie Scheffler: Thank you for raising a daughter who exceeds all expectations. The love and support you have shown us over the years has made all the difference.

To my partner and fiancée, Sarah Scheffler: I eagerly await our marriage this summer at Bell Labs. It took countless unlikely circumstances for us to end up together, and yet I cannot imagine us any other way. You are the single most wonderful thing in my life. I am excited to see what life has in store for us. Above all else, we are a team – I love you.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Multi-robot systems can be used to improve the efficiency and robustness of large-scale tasks such as search & rescue [1], warehouse automation [2], and planetary exploration [3]. To operate and parallelize effectively, these systems typically need to know where each agent (and its peers) are located in a common reference frame. In practice this is often achieved by localizing within an *a priori* map or using an external measurement system like GPS or motion capture (mocap) [4, 5]. If these technologies are unavailable or infeasible, common approaches utilize both relative localization [6] and multi-agent SLAM [7–10] techniques. Motivated by nuclear non-proliferation, two such tasks that benefit from said approaches, are radiological search and mapping.

Radiological search is the task of localizing one or more radioactive point sources within a large environment. Detection is probabilistic in nature and dependent on numerous parameters such as source strength, cross sectional area of the sensor, and overall distance between sensor and source [11]. Additionally, even if a source is strong, typical shielding weakens externally detectable signals, in turn greatly reducing the distance for a reliable detection. This makes it infeasible to survey large areas from afar for shielded sources, and thus groups of up-close searchers can greatly improve search speeds.

The importance of radiological search made international headlines last year during the Western Australian radioactive capsule incident of 2023 [12]. Here, a haz-

ardous – potentially deadly – capsule of radioactive caesium-137 was lost during truck transportation. It was safely recovered from alongside a road after approximately three weeks at large, luckily with no reported injury or loss of life.

A related problem, radiological mapping, occurs when nuclear plumes are mapped and monitored as they evolve via changing wind patterns and general spread. The persistent nature and potentially large area of interest makes this a challenging distributed sensing problem.

The importance of radiological mapping became apparent to the international public through the tragic Fukushima nuclear power plant disaster of 2011 [13]. Here, a chain of complications stemming from a significant earthquake and subsequent tsunami caused the Fukushima Daiichi nuclear power plant to contaminate the area with radioactive materials. Documents from the National Nuclear Security Administration (NNSA) [14] show that over the year following the incident, the evolution of the primary plumes delivered significant radiation doses within an 80km radius, with lesser plumes traveling significantly beyond – most notably towards the major population center of Tokyo. According to [15], over 100,000 residents were displaced during evacuation and official counts show over 2,000 disaster-related deaths. In the aftermath, the Fukushima nuclear accident has served as a valuable academic case study to improve future nuclear incident response [16].

These types of nuclear first-response and disaster prevention technologies are of obvious great importance to our collaborators at the Enabling Technologies & Innovation (ETI) consortium and Lawrence Berkeley National Laboratory (LBNL), as well as the Department of Energy (DOE) at large. Some recent relevant work in this space includes [17–20]. Recently our LBNL collaborators developed a state-of-the-art system for detecting special nuclear materials (SNM) in real-time, the neutron gamma localization and mapping platform (NG-LAMP) [21, 22]. For this system to be effective in real-world first-response and disaster scenarios, it is advised to avoid reliance on *a priori* maps or GPS – specifically, the environment may have drastically changed due to earthquakes, tsunamis, explosions, etc. and GPS can be easily denied or become unreliable. Thus, simultaneous localization and mapping (SLAM) must

be used to navigate the unfamiliar changed environment (see Section 1.2.1).

Leveraging the prior NG-LAMP work, as well as Torgesen's thesis [23], we aim to extend NG-LAMP into a coordinated multi-agent system. Specifically, with additional agents and effective inter-agent collaboration, an autonomous swarm will increase search speed (e.g., parallel search), accuracy (e.g., search redundancy), and robustness (e.g., hardware redundancy) compared to a single agent – invaluable improvements due to the often dangerous and time-sensitive circumstances in which the NG-LAMP may be deployed. Works such as [24–26] highlight the advantages of multi-agent teaming for rapid radiological point source detection and mapping. Thus, to enable multi-agent radiological search, NG-LAMP's onboard SLAM engine must be extended to support multi-agent collaborative SLAM (CSLAM).

CSLAM differs from SLAM in that each agent makes its own local map, that then must be joined into a global map via inter-agent loop closures. Conventional SLAM sensors (e.g., cameras, LiDARs, and IMUs) compute inter-agent loop closures by visually detecting that two local maps share a point of commonality (i.e., two agents have "crossed paths") through a process called inter-agent place recognition. This has two clear limitations: (1) Agents must intentionally "cross paths" to create inter-agent loop closures, a sub-optimal search pattern for a time-sensitive mission. (2) To detect inter-agent loop closures, agents must continuously compare visual data to find commonalities, leading to potential scaling issues with increased team size and/or decreased communication network capability.

For these reasons we explore equipping each agent with multiple ultra-wideband (UWB) ranging sensors as a cost-effective way to produce frequent inter-agent pose estimates without the need to "cross paths" or strain communication systems. Thus, our work, when integrated as a component of this larger future CSLAM system, will allow for frequent inter-agent loop closures without retracing steps, enabling the creation of better maps in less time, which in turn improves the overall multi-agent radiological search system. Figure 1-1 demonstrates the current proposed system, thoroughly developed in Chapter 5.

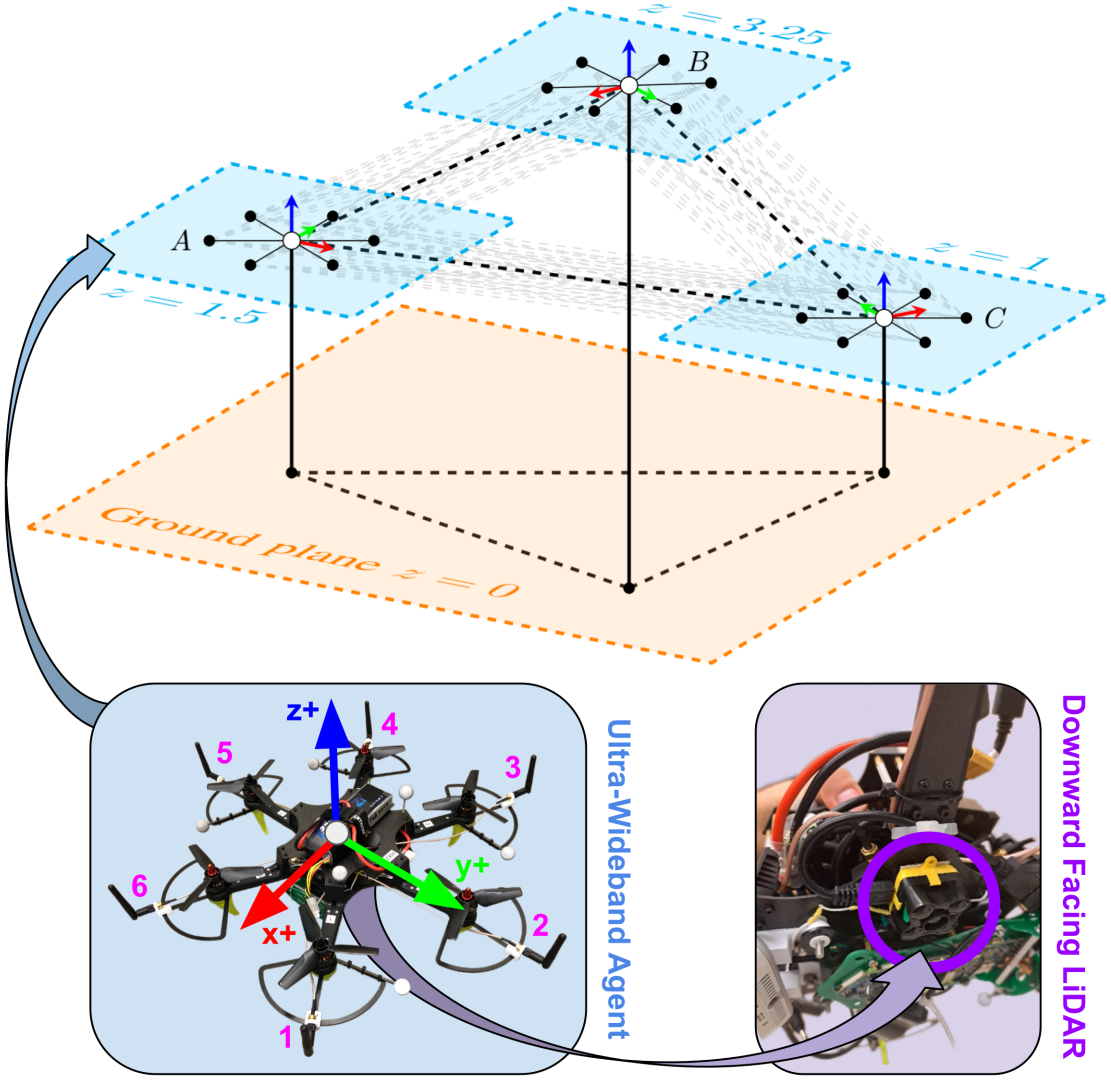


Figure 1-1: Diagram of proposed system. Here three agents fly at different altitudes while performing real-time 3D relative pose estimation. Each agent is equipped with six ultra-wideband (UWB) antennas, each capable of performing pairwise relative ranging between all other agents' individual antennas. By using trilateration, an improved sensor model, and *a priori* state constraints about altitude/roll/pitch, agents can perform *instantaneous* estimation entirely with *locally* collected UWB measurements (i.e., without the need to *continuously* transmit other measurements, such as odometry). Additionally, each agent *locally* monitors its *a priori* constraints via downward facing LiDAR and IMU, and thus only needs to transmit *one-off* messages with the swarm if these assumptions change or are violated.

1.1 Design Priorities

Toward the goal of achieving frequent inter-agent relative pose estimates with a minimal communication footprint, we take the following design approach in the upcoming chapters. See Chapter 2 for a literature review on UWB in robotics.

1. Each agent is equipped with multiple UWB ranging sensors in a known onboard configuration. This allows for a fully observable, infrastructure-free, *instantaneous* estimation of inter-agent relative pose with only current range measurements via a nonlinear least squares (NLLS) trilateration optimization problem.

2. Our work is intended as a component of a larger distributed multi-agent collaborative SLAM system with potentially hundreds/thousands of agents and limited communication network capabilities. Thus, to promote overall system scalability, our work prioritizes a minimal communication footprint over absolute estimation accuracy – in other words, although our system's absolute accuracy would be improved with the *continuous* sharing of odometry between agents, we do not want to rely on a centralized server or constant $O(n^2)$ inter-agent data exchange. This allows the finite communication budget to be allocated to other mission critical tasks such as swarm coordination/planning, detecting inter-agent visual loop closures, etc. How the overall system can best leverage our inter-agent relative pose estimates remains the subject of future work (see Chapter 6).

3. Since we do not rely on *continuously* transmitted measurements, we must make do with only *locally* collected UWB range measurements, *a priori* constraints, and *local* detections of when assumed state constraints are violated. This places a high importance on the error model of our UWB ranging sensors – that is, where other works can overlook or compensate for systematic ranging errors through the fusion of other *continuously* shared measurements (e.g., odometry), our estimation accuracy is highly reliant on understanding and accounting for these idiosyncrasies algorithmically. As such, we propose various

pose-dependent noise models that would benefit most other UWB-based mobile robotics work, even if they are not as reliant on it.

1.2 Background Topics

This section provides a gentle introduction to the core concepts utilized in this work. The provided citations and book recommendations can provide more thorough explanations as needed.

1.2.1 Simultaneous Localization and Mapping (SLAM)

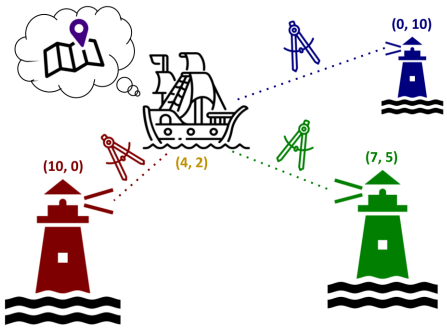
In a surprisingly rare technical feat, the fully expanded name "simultaneous localization and mapping" is actually a very descriptive definition of what SLAM is – specifically, SLAM is a computational robotics problem where an agent needs to both build a map of an unknown environment (i.e., *mapping*) while *simultaneously* tracking its location within said map (i.e., *localization*). In practice, there are many different types of SLAM, varying based on problem assumptions (e.g., feature-based vs. feature-less), environmental assumptions (e.g., static environment vs. time varying/deformable environment), computational machinery (e.g., filter vs. factor graph), etc. – the extensive survey paper [27] provides an excellent overview and historical account of the space.

A noteworthy open source full-stack visual SLAM system is 2015's ORB-SLAM [28], with its successor, ORB-SLAM3 [29], remaining relevant to this day. Additionally, Kimera [30, 31] is a popular SLAM library noteworthy for its novel 3D dynamic scene graph (DSG) representation, which allows it to capture metric and semantic aspects of the environment.

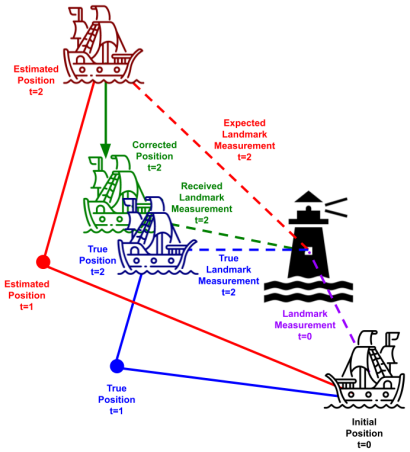
There are many references that provide an in-depth mathematical formulation of SLAM. The textbooks [32] and [33] cover general estimation theory, developing mathematical prerequisites that culminate in precise formulations of various SLAM approaches. In contrast, [34] is a text focused exclusively on developing modern visual SLAM. The book takes great care to discuss both the visual front-end and

optimization back-end, and comes with source code examples. Alternatively, the open source course materials for MIT's Visual Navigation (VNAV) class [35] covers similar material to [34], but in course structure. Finally, the tutorial paper [36] builds up factor graph theory, the back-end workhorse of many modern state-of-the-art SLAM systems. The related hands-on tutorial [37] is a shorter alternative formulated as a tutorial for GTSAM [38], a popular back-end optimization library.

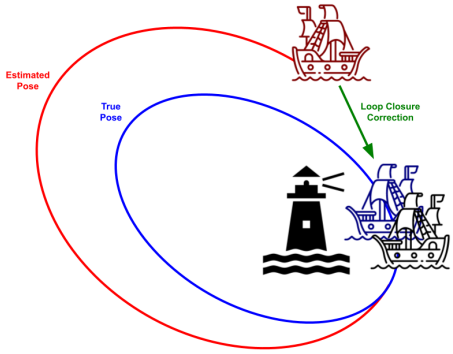
Collaborative SLAM (CSLAM) is a multi-agent extension to SLAM, with some notable examples being DOOR-SLAM [39] and Kimera-Multi [7, 8], the multi-agent extension of Kimera [30, 31]. Although implementation details vary between systems, generally, agents create individual local maps that are then placed into a common reference frame via inter-agent loop closures (i.e., inter-agent relative pose estimates). For typical visual CSLAM systems, these take the form of visual inter-agent loop closures, which in turn requires the agents to perform frequent inter-agent place recognition – that is, agents communicate compact bag-of-words representations of their current viewpoint to see if their maps share a point of commonality (i.e., they have "crossed paths"). Although these transmissions can be prioritized with estimates of loop closure detection likelihood, doing this in an intelligent, resource-aware way is a research topic in itself [40]. As the swarm grows, these techniques can scale poorly, becoming either more strenuous on the communications or necessitating the omission of lower priority potential loop closures. Additionally, Chapter 10 of [34] deals exclusively with loop closures and can be helpful for understanding how this thesis can contribute to an overall SLAM system.



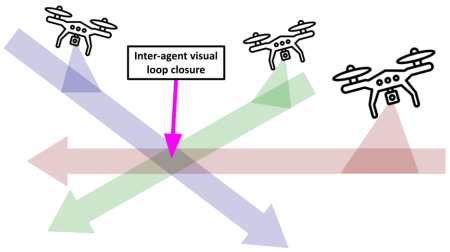
(a) Localization with an *a priori* map



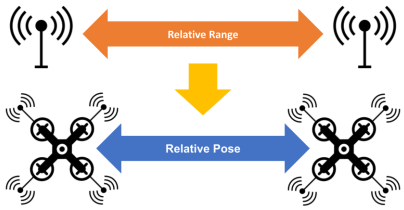
(b) Landmark SLAM



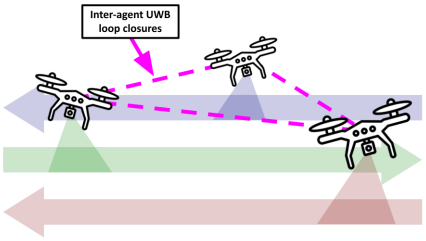
(c) Single-agent loop closure



(d) Visual inter-agent loop closure



(e) Multi-UWB relative pose



(f) UWB inter-agent loop closure

Figure 1-2: Visual summary of all core SLAM concepts as they relate to this thesis. Due to the length of each description, individual subfigures are discussed in Section 1.2.1.

The high-level core concepts of SLAM as they relate to this thesis are visually summarized in Figure 1-2 and discussed here:

- Figure 1-2a: Here, we see an agent (i.e., the boat) and three landmarks (i.e., the lighthouses). The agent has an *a priori* map, so it knows precisely where the landmarks are. It also has known correspondences (i.e., it can identify which landmark is which). Thus, by taking a measurement with respect to each landmark, it can solve for its current location. This process is called *localization*.

- Figure 1-2b: Here, we see an agent (i.e., the boat) and one landmark (i.e., the lighthouse). The agent does not have an *a priori* map (i.e., it doesn't know where the lighthouse is), but it knows the landmark is stationary. Initially at $t = 0$, the agent (represented by the black boat) sees the landmark and takes a measurement with respect to it. The agent then starts moving and integrating its odometry to estimate its position. Integrating noisy odometry causes gradual positional drift. Thus, at $t = 1$ the agent thinks it is at the red dot, when in reality it is at the blue dot. The agent then changes direction and continues to track and integrate its odometry. Then, at $t = 2$, the agent initially thinks it is at the red boat, when in reality it is at the blue boat. It then notices the landmark and takes a measurement with respect to it. The agent is able to correspond the landmark seen at $t = 0$ and $t = 2$ (i.e., it realizes it is the same lighthouse), and, since the landmark cannot move and measurements with respect to it are more certain than integrated odometry, the agent realizes its positional estimate has drifted significantly. Incorporating its new landmark measurement, it updates its positional estimate to the green boat location. We note that while the updated estimated location (i.e., the green boat) is not the same as the true location (i.e., the blue boat), it is a significantly better estimate than the initial $t = 2$ estimate (i.e., the red boat). This process of building a map of landmarks and localizing with respect to it is called *simultaneous localization and mapping (SLAM)*.

- Figure 1-2c: Here, we see an agent (i.e., the boat) and one landmark (i.e.,

the lighthouse). The agent does not have an *a priori* map (i.e., it does not know where the lighthouse is), but it knows the landmark is stationary. The agent (initially at the black boat), starts moving and integrating its odometry to estimate its position. Integrating noisy odometry causes gradual positional drift. At some later time, the agent thinks it is at some arbitrary position (i.e., the red boat), but by performing place recognition, it realizes it has actually returned to its starting location (i.e., the black/blue boat). Since the agent's starting position is stationary, it realized it has looped back to a previously explored location (i.e., it has "closed the loop"). The entire estimated trajectory can then be updated by constraining the first and last position to be related by some high confidence relative pose. This process of relating two states by a non-odometry relative pose constraint is called *loop closure* and is an important way to correct drift in a SLAM system. We note this is a distinct and separate process from the landmark based relation discussed in the previous description – here, this constraint relates two states directly, whereas the previous description related two states through a static landmark.

- Figure 1-2d: Here, we see three agents (i.e., the UAVs) each flying along its own search path constructing an individual local map. Since each agent initially has no information about where the other agents are relative to itself, and there is no means to determine an absolute position (e.g., the scenario is GPS-denied), there is no way to relate the local maps or odometry measurements between agents. The agents try to perform inter-agent place recognition – if any agents discover they have "crossed paths" with another (i.e., multiple agents have visited the same location), there is now a relationship between a pose in each agent's local trajectory, an inter-agent loop closure. This relationship now allows the local maps to be united into a shared reference frame. Additional inter-agent loop closures can further connect two local maps, but without any type of absolute position landmarks, a unified map can still suffer global drift (i.e., if both local maps drifted in similar ways, the agents would not be able

to recognize this). Since the agents are only equipped with conventional SLAM sensors (e.g., cameras, LiDARs, and IMUs), inter-agent loop closures can only be found visually, requiring that agents "cross paths" to discover inter-agent loop closures. This process is called *inter-agent visual loop closures* and is how two local maps are related into a common reference frame in CSLAM.

- Figure 1-2e: The UWB sensors used in this work are relative range sensors (see Section 1.2.5). Thus, a pair of antennas can only determine an ambiguous 1D distance between each other – not a relative pose, which is 3-DoF in 2D (i.e., x, y, and yaw) or 6-DoF in 3D (i.e., x, y, z, roll, pitch, and yaw). That being said, equipping each agent with multiple UWB ranging sensors in specific known configurations, allows us to *instantaneously* (i.e., using only current measurements) recover a full inter-agent relative pose (see Section 1.2.3). This process is additionally highlighted in Figures 4-1 and 1-1 for 2D and 3D respectively.

- Figure 1-2f: Here, we see three agents (i.e., the UAVs) each flying along its own search path constructing an individual local map. Since each agent is equipped with multiple UWB ranging sensors in a specific known configuration, each agent continuously receives full inter-agent relative pose measurements. These UWB-based inter-agent loop closures allow the local maps to be united into a shared reference frame without the need to "cross paths" (i.e., as with visual inter-agent loop closures). In the context of time-sensitive radiological search, this allows the creation of a unified CSLAM map, without requiring agents to "cross paths" for visual inter-agent loop closures.

1.2.2 Instantaneous Localization Problems

The goal of localization is to estimate an agent's pose (i.e., position and orientation) within a frame. Instantaneous localization algorithms are often geometric (and thus nonlinear) in nature, but have the advantage of being model-free and requiring only current-time measurements [41]. Typical approaches leverage a set of angular, range,

or pseudorange[1] measurements, which in turn allows one to produce a position estimate via triangulation, trilateration, or multilateration[2] respectively [43]. Examples of such systems include radar and GPS [44]. In reality, these measurements are actually indirectly observed by directly measuring features of a received signal, for example received signal strength (RSS) – that is, to indirectly observe angular, range, or pseudorange measurements, what is directly measured is actually often angle of arrival (AOA), time of arrival (TOA), or time difference of arrival (TDOA) respectively [45]. In robotics, these signals can take many possible forms: acoustic [46], Bluetooth Low Energy (BLE) [47], Radio Frequency Identification (RFID) [48], WiFi [49,50], or ultra-wideband (UWB) (see Section 1.2.5). Survey paper [51] provides an overview on indoor positioning systems and how different measurement technologies (i.e., WLAN, GSM, Bluetooth, UWB, etc.) can be used to compute position.

There are many references that provide an in-depth mathematical handling of these localization topics. The textbook [44] is an exhaustive resource on all things position location system related. Alternatively, specialized textbooks such as [52,53] are primarily concerned with GPS, whereas [54] is mostly an introductory linear algebra book with a special focus on GPS. Furthermore, early discussions can be found in old laboratory technical reports for various satellite systems [55–57].

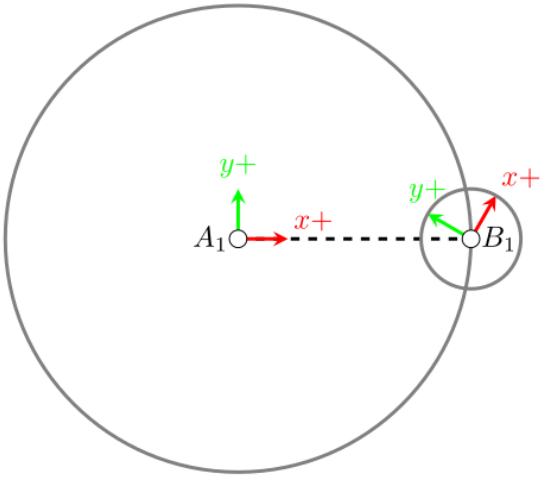
1.2.3 Trilateration for Pose Estimation

Trilateration is ultimately a fundamental 2D/3D geometry problem built upon the intersection of circles/spheres respectively [52,53,58,59].

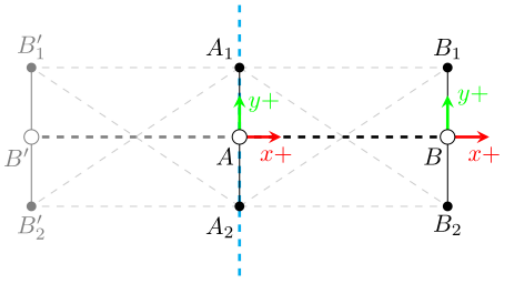
To understand how we can reconstruct relative pose with trilateration, we consider two agents, a base and a target, each equipped with multiple relative range sensors in a sufficient known configuration. Since the base agent is sufficiently equipped, it can estimate the position of one of the target's antennas through standard trilateration. That process can then be carried out independently for each target antenna. This in

[1]Here, *pseudorange* is a biased range measurement resulting from an unknown time offset [42]. It is commonly associated with GPS and other TDOA-based multilateration positioning systems.

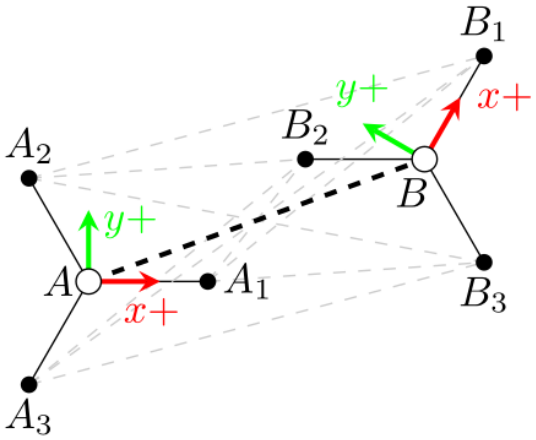
[2]Multilateration can also be called hyperbolic positioning [43].



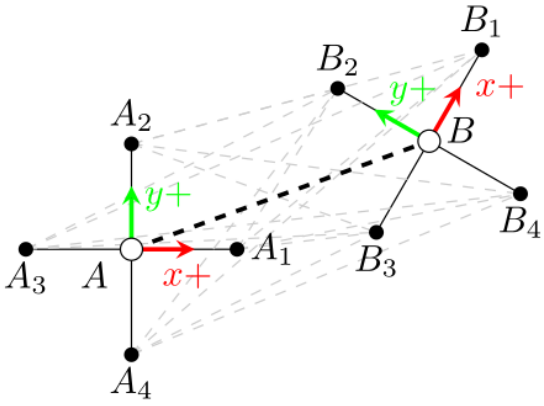
(a) **2D environment, 1 antenna per agent:** Cannot distinguish between any poses along the circle



(b) **2D environment, 2 antennas per agent:** Cannot distinguish between 2 poses, reflection over blue line

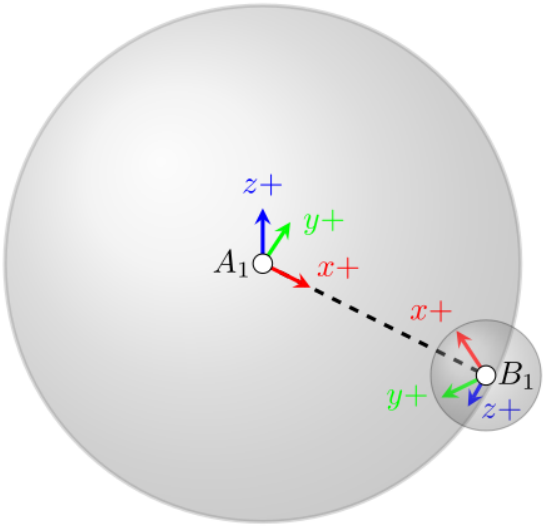


(c) **2D environment, 3 antennas per agent:** Unique solution, fully observable

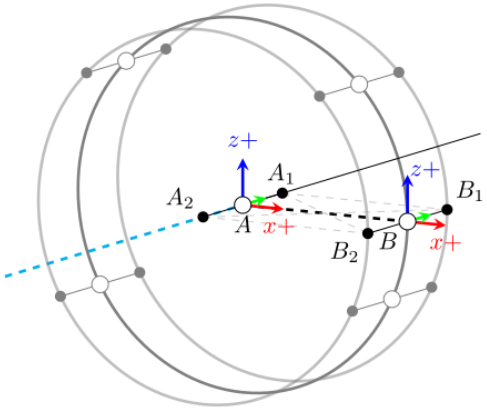


(d) **2D environment, 4 antennas per agent:** Unique solution, fully observable

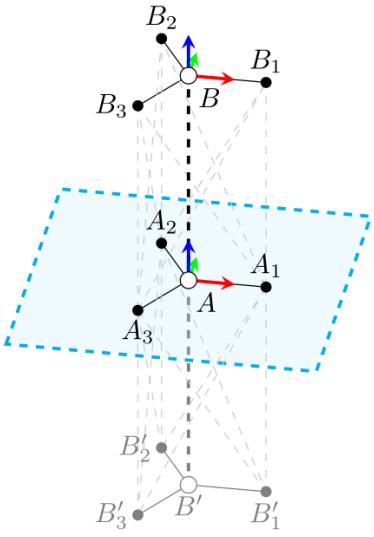
Figure 1-3: Diagrams showing agent A's ability to observe agent B's pose in a 2D environment when each is equipped with $i = \{1, 2, 3, 4\}$ antenna(s) respectively. Antennas maximally span their space.



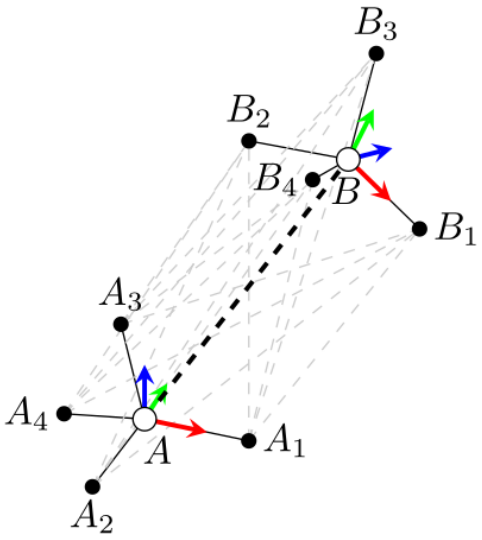
(a) **3D environment, 1 antenna per agent:** Cannot distinguish between any poses along the sphere



(b) **3D environment, 2 antennas per agent:** Cannot distinguish between any poses along the circle



(c) **3D environment, 3 antennas per agent:** Cannot distinguish between 2 poses, reflection over blue plane



(d) **3D environment, 4 antennas per agent:** Unique solution, fully observable

Figure 1-4: Diagrams showing agent A's ability to observe agent B's pose in a 3D environment when each is equipped with $i = \{1, 2, 3, 4\}$ antenna(s) respectively. Antennas maximally span their space.

turn gives us several trilaterated points on the target agent. Since the target's antenna layout is known and sufficient, it allows us to determine how it must be positioned and oriented without any ambiguities. Thus, for relative pose to be observable from relative range measurements, we must have: (1) The base agent being able to observe the location of a single target antenna. (2) Relative pose being observable from two sets of points with known correspondences (i.e., the known configuration and the measured points).

Observability of a Single Antenna: In 3D trilateration, a minimum of 4 non-coplanar antennas are required to uniquely determine a target's position [60]. That being said, 3 or more planar (but not colinear) antennas produce only a pair of solutions that are a reflection across the antenna plane (see Figure 5-1c). This ambiguity can be resolved with a simple z measurement (i.e., altitude). In 2D, these requirements are simplified to 3 nonlinear antennas [52].

Observability of Pose from Correspondences: Common in camera-robot registration, Horn's Method is a closed-form solution for finding the pose between two Cartesian coordinate systems from a set of corresponding point pairs [61–63]. Specifically, given exactly 3 planar points in two frames with known correspondences, Horn's Method fully specifies the 3D translation and orientation. Thus, by using trilateration to measure 3 points in a known configuration on our target, we can recover full pose. Additionally, we know this must also work in 2D since the 2D system requires 3 nonlinear (i.e., planar) antennas, thus satisfying Horn's Method.

Thus, we have reasoned 3D pose is observable when each agent has either 4 non-coplanar antennas, or 3 nonlinear antennas with a altitude measurement. Likewise, 2D pose is fully observable when each agent has 3 nonlinear antennas. In the presence of noise, we will have to solve an optimization problem (e.g., nonlinear least squares) instead of a closed form algebraic expression. That being said, this also allows us to add extra measurements, for redundancy and resilience against individual erroneous measurements. Figures 1-3 and 1-4 show the observability of different numbers of antennas in 2D environments and 3D environments respectively.

1.2.4 Dilution of Precision (DOP)

Most commonly associated with GPS, dilution of precision (DOP) is a trilateration (or multilateration) sensitivity analysis for quantifying how 1D ranging errors propagate to 2D/3D point estimation error [64]. Specifically, given trilateration's inherent nonlinearity, even identically noisy 1D range measurements can produce significantly different 2D/3D positional uncertainties – this is entirely determined by the given base station and target geometry (Figure 1-5). A lower DOP is good (i.e., less uncertainty), while a higher DOP is bad (i.e., more uncertainty). In general we note that increasing the "baseline" distance between base stations (i.e., the antenna separation on a single vehicle), or having a target closer to the base stations (i.e., decreasing distance between two vehicles), reduces trilateration's 2D/3D uncertainty (i.e., lowers DOP). See Figure 1-5 for several examples of DOP in a 2D scenario.

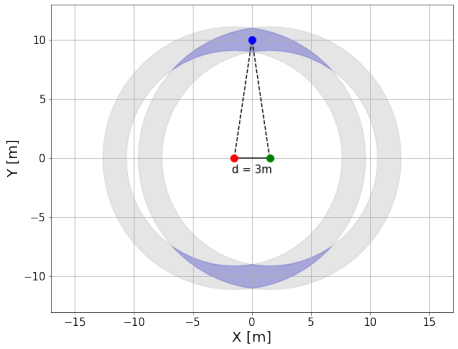
DOP can also make comparing results between works challenging. Specifically, even if two systems implement the exact same algorithm, if the baselines or operating ranges are vastly different, we would expect to see changes in the absolute accuracy of the trilateration-based results – this can be seen in action in [65], where otherwise identical configurations with larger baselines yield improved accuracy.

There are many references that provide an in-depth mathematical handling of DOP. The article [64] provides a brief but complete introduction, while [66] provides an analytic DOP analysis. The pair of volumes [67, 68] are near exhaustive on all things GPS. See [69, 70] for two very relevant UWB robotics papers discussing DOP as it pertains to 2D UWB localization.

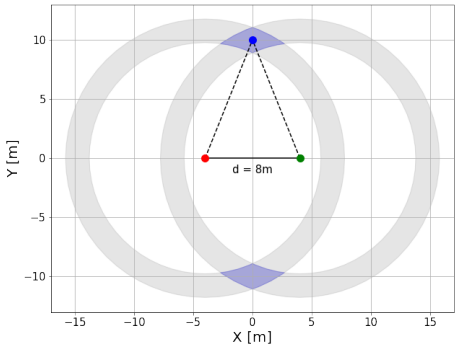
1.2.5 Ultra-Wideband

Ultra-wideband (UWB) is a multipurpose radio technology named for its characteristically large bandwidth [71, 72] – by definition "-10 dB bandwidth of at least 500 MHz or a -10 dB fractional bandwidth greater than 0.2" [71].

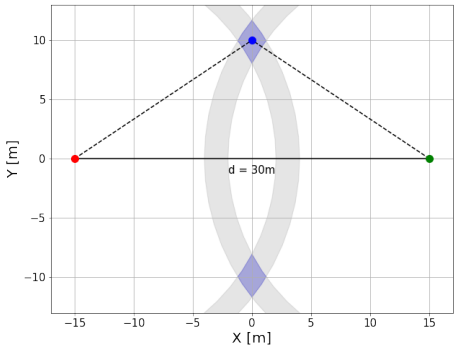
Within the last decade, ultra-wideband (UWB) has matured into a reliable, inexpensive, and commercially available RF solution for data transmission, relative



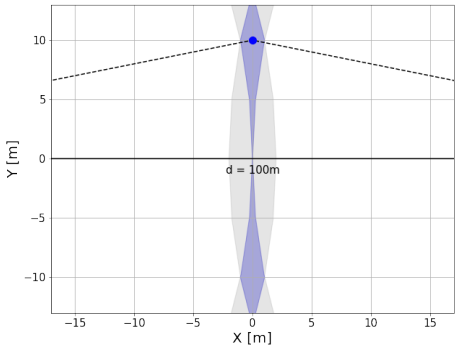
(a) DOP with small baseline



(b) DOP with medium baseline



(c) DOP with large baseline



(d) DOP with disproportionately large baseline

Figure 1-5: Simple example demonstrating dilution of precision (DOP) in several 2D ranging scenarios. Here, the red and green points are the base stations, and have a baseline distance of d. Both the red and green points measure the distance to the blue point with a consistent uncertainty of $\pm$1m. The position of the blue point is estimated by intersecting the measurement uncertainties, creating a blue feasibility region. Despite having the same 1D ranging uncertainty in all scenarios, we see the shape and size of the feasibility region changes. In general we see as the baseline gets larger (i.e., (a) $\rightarrow$ (b) $\rightarrow$ (c)), the estimation uncertainty shrinks (i.e., the DOP decreases). That being said, (d) shows that with disproportionately large baseline, DOP can begin to increase again. Ultimately, this shows when placing multiple antennas on each robot, we want to maximize the baseline distance to give us lowest uncertainty (i.e., best DOP geometry).

ranging, and localization [73] – UWB now comes as standard issue in many popular smartphone devices [74]. For robotics, UWB has several properties of note: precision of approximately 10cm, ranges up to 100m, resilience to multipath, operates in non-line of sight (NLOS) conditions, low power consumption, and 100Mbit/s communication speeds [75]. Recent devices even extend the recommended and operational ranges to 300m and 500m respectively [76]. Nevertheless, UWB measurements are not immune from ranging errors or noise (see Chapter 3), the modeling and correction of which is an active area of research [77–80]. See Chapter 2, for a discussion of how UWB is being used in robotics research.

1.3 Thesis Contributions

Following a literature review of UWB in robotics (Chapter 2), this thesis's main technical contributions are as follows:

(a) Exploration of current commercially available off-the-shelf (COTS) UWB devices for use in mobile robotics. By analyzing real data, insights into commonly overlooked sensor quirks are addressed through our improved sensor models. (Chapter 3)

(b) Development and testing of a novel 2D relative pose estimation system based on trilateration, leveraging multiple UWB ranging sensors per agent. (Chapter 4)

(c) Extension of said system to 3D environments. (Chapter 5)

(d) A list of recommendations and continuations for future work. (Chapter 6)

This thesis is based on the following papers [81, 82].

Chapter 2

Related Work

The survey [83] provides a general synopsis of aerial swarm robotics, while [84] is an overview of current UWB usage in robotics and IoT pipelines. While this thesis uses UWB for infrastructure-free inter-agent relative measurements, UWB technology can be leveraged by robotics in multiple seemingly orthogonal ways. First, in Section 2.1 we will briefly review various uses of UWB in robotics. Then in Section 2.2, we will discuss Table 2.1, which highlights the unique capability gap our work fills, as well as the most immediately relevant works to this thesis.

2.1 UWB in Robotics

UWB in mobile robotics naturally separates into several categories, each of which we discuss briefly.

Known Anchor Points: In these systems a target with a single UWB tag is localized by several UWB anchor points at known global locations via trilateration. Anchors are often assumed to be static with respect to each other. Since the target has a single tag, the target's orientation is not observable from just UWB measurements. These works produce improved state estimation results by fusing the global UWB position estimation with some combination of dynamics models, odometry (e.g. wheel odometry or visual odometry), and IMU data [85–90]. A recent outlier [91], uses an agent equipped with multiple UWB tags to estimate continuous-time relative pose

from several static anchor points – by modeling the system as a Gaussian process, they achieve comparable performance to state-of-the-art sensor-fusion methods with only UWB ranging measurements.

Static Features: These works assume there are one or more static UWB anchors with not necessarily known positions. The observed anchor(s) are treated like static SLAM feature(s) and fused with odometry data to perform localization [49, 92, 93]. This is also comparable to [50], but WiFi hotspot relative bearing measurements are used instead of UWB ranging measurements. Also in this category is [3, 94], where the robot deploys a stationary UWB node during exploration, effectively marking a static point in the environment useful for future loop-closure detection.

Swarm Ranging with GPS: Many projects – including ours – motivates UWB as a localization tool in GPS-denied environments. Works like [95–97], however, fuse UWB range measurements with GPS for improved results. The inclusion of absolute position information differentiates how these works generally leverage their UWB measurements. For instance, [95] uses a UWB pose estimate to adaptively improve GPS performance.

Observability Results: Mobile agents, each equipped with one (or possibly two in the case of [98]) ranging tags, take several temporally spaced relative range measurements while each traversing some trajectory. Agents track their local trajectory through odometry or IMU dead reckoning and share it with other agents. As shown in [6, 99], if the trajectories have sufficient relative motion then the relative pose between agents can be recovered.

Correcting UWB Ranging Errors: Since UWB manufacturers typically hide their algorithmic choices and calibration parameters, works modeling and correcting the ranging errors or design calibration schemes are valuable [77–80]. Additionally, recent papers with large datasets, like [100], have started to include UWB data, making it easier for future work in this space.

Range Aided Optimization Relaxations: In an increasingly popular trend, UWB relative range measurements are being incorporated into sophisticated optimization

relaxation protocols [101–104]. Specifically, with the addition of range measurements, problems can be carefully relaxed to semi-definite programs. This allows the original extremely non-convex problems to become less dependent on good initialization.

Multi-Tag Pose Estimation: Mobile agents are each equipped with multiple non-colocated tags at known relative positions on the agent. With sufficient tags, relative position and orientation can be observed with UWB measurements alone [105–107]. Furthermore, additional measurements, such as odometry, optical flow, IMU readings, and altitude measurements can be communicated between agents and fused with the UWB measurements, as in [108] and [109], improving result and observability respectively. Ref. [108] presents work most similar to our proposed 2D solution in Chapter 4. Their work differs by using a particle filter and needing to continuously share odometry between agents. The similarities between the hardware setup and experiments allow for a useful baseline of comparison for our experimental results in Chapter 4. A recent outlier [110], only uses one tag per agent, but achieves behavior like multi-tag by placing said tag on a large wheel rotating about the gravity vector.

Miscellaneous Works of Interest: Steup et al. [111] provides an interesting full pipeline approach, where each agent's role alternates between active swarm member and static anchor node. This provides an interesting way to reduce global map drift by having fixed UWB landmarks. Although not directly relevant to this thesis, architectures like these are relevant to future radiological search systems.

Wang et al. [112] uses a combination of UWB range and Wi artFi angle measurements as a way to exclude computation of unlikely inter-agent loop closures. This has similarities to ways our system can be leveraged by the larger CSLAM system.

2.2 Discussion of Most Comparable Work

All works in Table 2.1 use UWB for inter-agent ranging, and are thus most similar to our work in Chapters 4 and 5. That being said, it is challenging to directly compare numerical results given the vastly different operating assumptions and priorities – specifically access to additional sensors (e.g., cameras, LiDAR, etc.), the size of

Table 2.1: Comparison of relevant related works to highlight the capability gap this letter fills. Color emphasizes how a work fits in with our goals. Specifically, *green* indicates our needs are met, *yellow* indicates our needs are partially met, and *red* indicates our needs are not met.

2D / 3D	Related Work	# UWB Per Agent	UWB Noise Model	Data Exchanged
2D	Guo et al. (2020) [113]	Single	Bounded Error Assumption	UWB + Velocity
	Cao et al. (2021) [108]	Many (4)	NLLS	UWB + Odometry
	Zheng et al. (2022) [69]	Many (2 or 4)	ZM Gaussian + NLOS Rejection	UWB + VIO
	Zhang et al. (2023) [70]	Many (2 or 4)	ZM Gaussian + Outlier Rejection	UWB + Point Clouds + Odom + Keyframes
	[Chapter 4] (2022) [81]	Many (4)	NLLS + Data Informed Sensor Model	UWB
3D	Xu et al. (2020) [114]	Single	ZM Gaussian	UWB + VIO + Visual Tracks
	Qi et al. (2020) [97]	Single	ZM Gaussian	UWB + IMU + GPS Heading
	Xianjia et al. (2021) [65]	Many (2 or 4)	NLLS	UWB
	Xu et al. (2022) [115]	Single	ZM Gaussian + Occlusion Rejection	UWB + VIO + Visual Tracks
	Xun et al. (2023) [116]	Single	ZM Gaussian	UWB + IMU + IR Visual Tracks
	[Chapter 5] (2023) [82]	Many (4 or 6)	NLLS + Data Informed Sensor Model	UWB + *One-Off* Assumption Violations

agents with respect to their environment (see Section 1.2.4), and overall communication model (e.g., unlimited vs constrained communication) all significantly change performance and scalability. To contextualize this work, we discuss [70] and [115] since they are the most complete and comprehensive 2D and 3D UWB systems respectively, [116] since it has a unique compact hardware solution, and [65] since it is the most directly comparable to this work.

Zhang et al. [70] presents a centralized 2D range-aided cooperative localization and consistent reconstruction system, merging a tightly coupled visual odometry and multi-tag ranging front-end with a pose graph optimization (PGO) back-end. This work differs from ours by being 2D and relying on *continuous* transmission of point cloud, odometry, and keyframe information.

Omni-swarm [115] is a decentralized 3D swarm estimation system. This work differs from ours by relying primarily on visual tracks of neighboring agents from each agent's omnidirectional camera – omitting UWB measurements only degrades estimation by approximately 0.01m. Additionally, each agent has only a single UWB tag and requires *continuously* sharing measurements between all pairs of agents, scaling

poorly with larger swarms.

CREPES [116] presents a custom compact hardware module, which tightly couples an IMU, UWB, IR LEDs, and IR fish-eye camera. The measurements are fused into 6-DoF relative pose via a centralized error-state Kalman filter (ESKF) and PGO, while also having the ability to provide an *instantaneous* estimate from a single frame of sensor measurements. This work differs from ours by relying heavily on visual tracks over UWB (similar to [115]). Additionally, the system relies on *continuously* transmitted measurements. Thus the state-of-the-art approaches in [70, 115, 116] achieve APE and AHE on the order of 0.10m and 1°, respectively, but they require additional sensors (e.g., LiDAR, fish-eye cameras, etc.) that must *continuously* transmit measurements, making them scale poorly to scenarios with larger swarm sizes or reduced communication throughput.

Xianjia et al. [65] is the most similar to our work in Chapter 5 (i.e., 3D environment, multiple UWB tags per agent, and using only UWB measurements). Thus we directly compare their work to our results in Chapter 5.

Chapter 3

UWB Sensors & Noise Modeling

This section overviews our hardware selection process and sensor noise modeling. These sensors and noise models are used throughout the rest of this thesis.

3.1 Hardware Selection

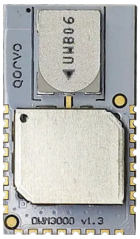
Figure 3-1 shows all sensors that, at some point, have been purchased and tested in some capacity. When purchasing the first batch in 2021, only three sensors were available – the Nooploop LinkTrack P, DWM1000 (and its variants), and the Pozyx UWB Tag. While the DWM1000 is not very user friendly (i.e., it is meant as a module to be soldered onto a circuit board), the Pozyx UWB Tag is a end-user board built with the same integrated circuit as the DWM1000[1]. Ultimately, we settled on the LinkTrack P because: (1) the LinkTrack P has a slim form factor (5.5cm x 3cm x 0.75cm), light weight (33g), and easy mounting via a 1/4-20 screw hole. (2) Nooploop provides an out of the box ROS driver. (3) In our trials using 8 tags, the LinkTrack P provided measurements at steady 50Hz, while the Pozyx Developer Tag fluctuated around 5Hz – they both had similar accuracy. (4) Most works in Table 2.1 use a version of the Nooploop LinkTrack P [76] for UWB ranging, except for [65, 113], which use the PulsON 440 and an unspecified device respectively.

[1] According to [116], the LinkTrack P modules are also based on DW1000 chips. We have not, however, been able to verify this in the official documentation.

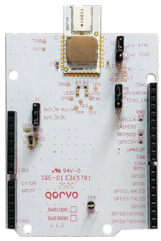
(a) Nooploop LinkTrack P[2]

(b) Nooploop LinkTrack P-B[3]

(c) DWM1000[4]

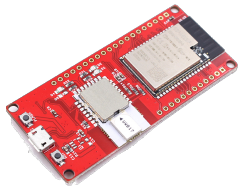


(d) DWM3000[5]



(e) DWM3000EVB[6]

(f) Pozyx UWB Tag[7]



(g) ESP32 UWB DW3000[8]

Figure 3-1: Commercially available off-the-shelf UWB sensors considered for this work.

3.2 Nooploop LinkTrack P and P-B – Helpful Notes

This section contains miscellaneous notes regarding the LinkTrack P and P-B and its configuration settings.

- The LinkTrack P advertises 1D and 2D typical positioning accuracy is 10cm, and 3D typical positioning accuracy is 30cm [76], maximum of 200 concurrent nodes, maximum refresh rate of 200Hz (refresh rate drops as nodes increase).

- The work in Chapter 4 uses the LinkTrack P. The work in Chapter 5 changed to use the newer LinkTrack P-B due to supply issues. According to [117], upgrading from the LinkTrack P (retroactively referred to as the P-A) to the P-B did the following: (1) Maximum communication distance: 120m → 500m; (2) Recommended distance: 40m → 300m; (3) Frequencies: 4,4.5,6.5GHz →

[2] Image reference: https://www.nooploop.com/wp-content/uploads/2020/03/LinkTrack-P_800.png, retrieved August 15, 2023

[3] Image reference: https://www.nooploop.com/media/uploads/2023/02/LinkTrack-P-B-600x600.png, retrieved August 15, 2023

[4] Image reference: https://www.mouser.com/images/marketingid/2020/img/159215248.png, retrieved August 15, 2023

[5] Image reference: https://www.mouser.com/images/marketingid/2020/img/136958127.png, retrieved August 15, 2023

[6] Image reference: https://www.mouser.com/images/marketingid/2020/img/177838291.png?v=070223.0226, retrieved August 15, 2023

[7] Image reference: https://global-uploads.webflow.com/612f4c781c90a5752d371287/615ac829a3248542e5e300ef_Developer-Tag.webp, retrieved August 15, 2023

[8] Image reference: https://www.makerfabs.com/image/cache/makerfabs/DW3000/ESP32%20UWB%20DW3000-1000x750.jpg, retrieved August 15, 2023

4,4.5GHz; (4) Small increase in physical size and weight (see Figures 3-1a and 3-1b). (5) When upgrading from LinkTrack P to LinkTrack P-B, we saw a decrease in mean bias error.

- All experiments are run in Channel 3 (3,744-4,243.2MHz) [118].

- The LinkTrack P has three different modes: (1) Local Positioning (LP) Mode: Sensors are configured as either mobile *tag* or static *anchor*. (2) Distributed Ranging (DR) Mode: All sensors are configured as *nodes* and perform pair-wise relative ranging with all sensors. (3) Digital Transmission (DT): For data transmission only. All experiments are run in *Distributed Ranging (DR) Mode* to enable measuring relative range without stationary anchors.

- In accordance to the LinkTrack manual [119], all antennas are all positioned upright, since their xy-plane has better omni-directivity than their z-axis.

- As per [119], when RSSI is less than 6dB, it is likely to be in the line-of-sight state, and when greater than 10dB, it is likely to be in the non-line-of-sight or multipath state. Despite this, we consistently observe `fp_rssi` and `rx_rssi` of approximately -81.0 dB and -78.0db respectively, with a 0.5dB minimum resolution.

- When in DR mode the sensor appears to use a time-division multiple access (TDMA) two-way ranging (TWR) scheme under the hood. TWR schemes have the advantage of removing the need to synchronize clocks between sensors while also observing the distance on both ends [120].

3.3 LinkTrack P Noise in 2D Environments

These experiments were conducted as part of [81] and are used in Chapter 4.

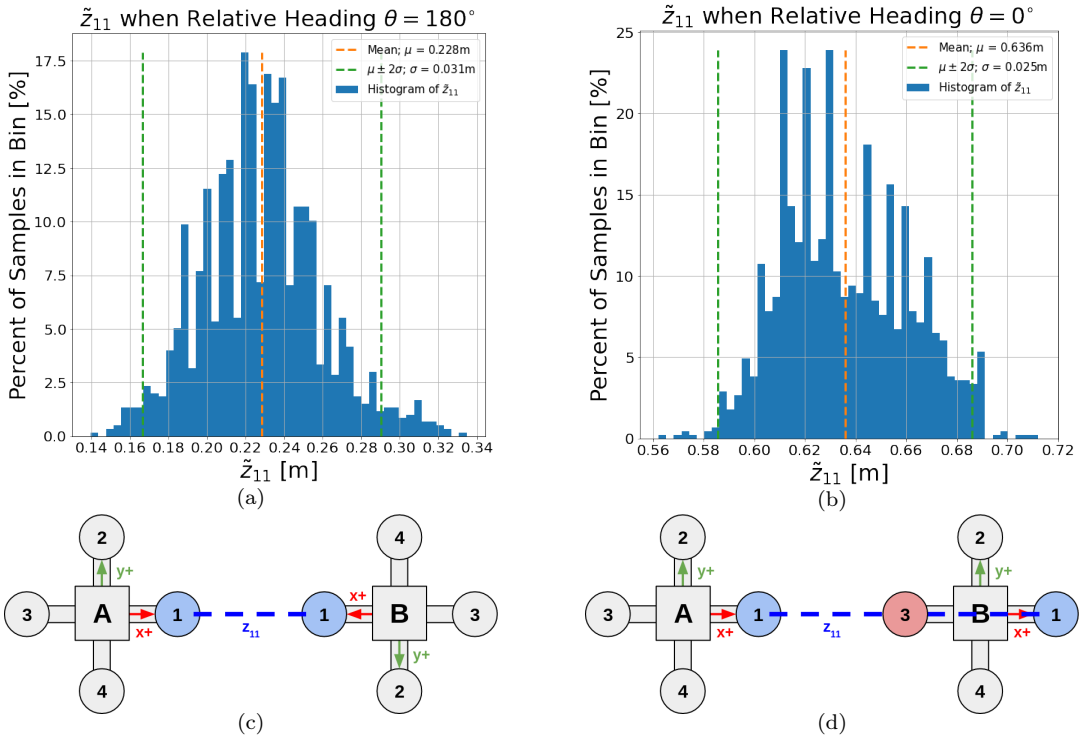


Figure 3-2: Histograms showing the measurement error for two separate data collections of $\tilde{z}_{11}$ (50Hz for 30s). Measurement error $\tilde{z}_{11}$ is the difference between the UWB range measurement and ground truth distance as measured by a Vicon system. Histogram (a)/(b) correspond to the error from the direct line-of-sight/obstructed configuration shown in (c)/(d) respectively. Neither of these distributions would be well-modeled as Gaussian. The orange lines and green lines show the calculated sample mean and $\pm$ two standard deviation boundaries respectively. With no obstruction, we see an average of 23cm over-estimation of z_{11}, and when obstructed, this increases by 41cm, to an average of 64cm over-estimation. Thus the error is not zero mean (and the mean increases further when obstructed).

3.3.1 Range Measurement Bias & Noise

Experiments were performed to characterize the UWB ranging noise. Figure 3-2a shows a sample of measurement error between two stationary UWB nodes with direct line-of-sight (LOS), as shown in Figure 3-2c. Three important observations are: (1) Contrary to the common measurement noise assumption, the error appears neither zero mean nor Gaussian (i.e. shown sample fails the `scipy.stats.normaltest` function [121], an implementation of the D'Agostino and Pearson's normal test [122], with a p-value of 2.36×10^{-6}). (2) Sensors tend to consistently over-estimate the relative distance between nodes. (3) Within our operating environment, the mean error between a pair of unobstructed UWB nodes remains approximately constant inde-

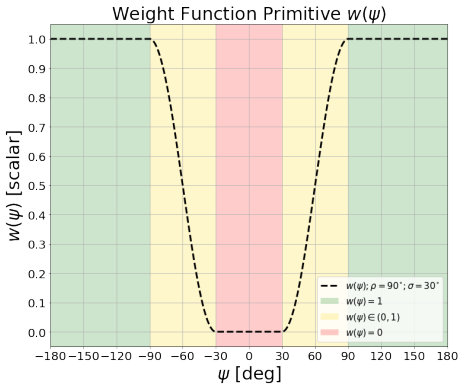
pendent of distance between nodes or when data was collected. Additional evidence of this claim can be seen in unobstructed portions of Figures 3-3b-3-3e, a follow-on experiment motivated in Section 3.3.2.

3.3.2 Antenna Obstruction & Interference

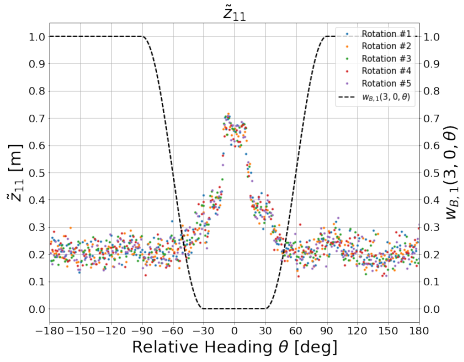
Since UWB relative range measurements are a TOA measuring scheme, any delays on receiving a signal (i.e. propagating through an obstruction) will result in a ranging over-estimation. This helps make sense of the substantial 41cm increase of mean error shown in Figures 3-2b and 3-2d when compared to Figures 3-2a and 3-2c. Furthermore, the LinkTrack P manual [119] offers several notes regarding obstruction: (1) Signal can propagate through 2 or 3 solid walls, but each wall introduces approximately 30cm of error and a decreased maximum ranging distance. (2) The distance between each node and the obstruction affects ranging accuracy; the best results occur when the obstruction is equally spaced between the antennas, the worst results occur when the obstruction is close to one of the antennas. This presents a noteworthy concern: as agents move relative to each other, fixed obstructions on the robot, such as a third antenna, can pass between the ranging pair causing an obstructed measurement. This obstruction is worsened by its guaranteed proximity to at least one of the ranging antennas. This eclipse-like effect can be observed in Figures 3-3b-3-3e, where clear over-estimation error spikes occur predictably at specific relative heading for each pair of antennas.

While it is difficult to pinpoint the exact cause of this obstruction, it appears to be a combination of proximity to other antennas, metal, and other components. Though it would be possible to mount these sensors in a different configuration to mitigate this, in reality there are times when obstructions cannot be avoided. Thus, we will treat this interference as part of the given hardware setup which in turn must be mitigated algorithmically.[9]

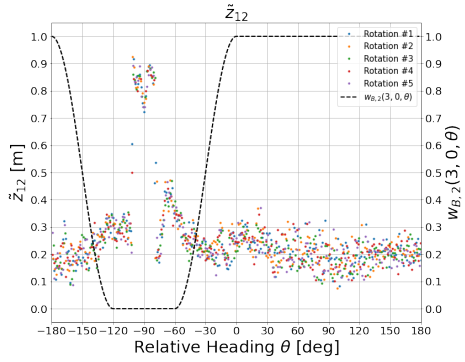
[9]Inspection of the received RSSI does not appear to provide any meaningful way to detect the current obstruction. As per [119], when RSSI is less than 6 dB, it is likely to be in the line-of-sight state, and when greater than 10dB, it is likely to be in the non-line-of-sight or multipath state. Despite this, we consistently observe `fp_rssi` and `rx_rssi` of approximately -81.0 dB and -78.0db



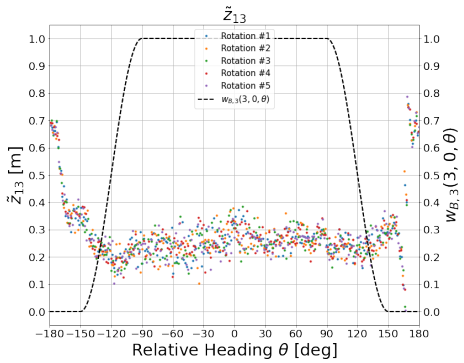
(a)



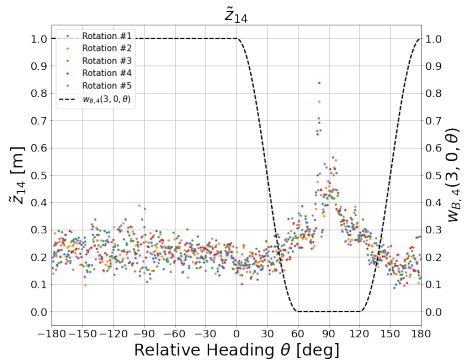
(b)



(c)



(d)



(e)

Figure 3-3: (a) Weight function primitive $w(\psi)$ with parameters $\sigma = 30°$ and $\rho = 90°$ (see Eq. 4.4). Colors represent the segments of the piecewise function, and align with the colors in Figure 4-2. (b-d) Plots of $\tilde{z}_{11}$ through $\tilde{z}_{14}$ with respect to R_B's relative heading. Shown data is all from the same data collection where R_A was kept stationary while R_B performed 5 in place counter-clockwise revolutions at approximately 60 deg/s. Each revolution is denoted in a different color to highlight this noise pattern is repeatable. The overlaid weight function $w_{B,j}(\mathbf{x})$ (see Eq. 4.7) shows that areas with error spikes correspond to relative headings correctly predicted and devalued by $w_{B,j}(\mathbf{x})$. Additionally, when the error is not spiking, we see additional evidence that the noise is not zero mean, as discussed in Section 3.3.1.

3.4 LinkTrack P-B Noise in 3D Environments

These experiments were conducted as part of [82] and are used in Chapter 5.

The following subsections motivate our hardware and algorithmic choices for operating in 3D (Section 5.1).

3.4.1 Characterizing UWB Noise

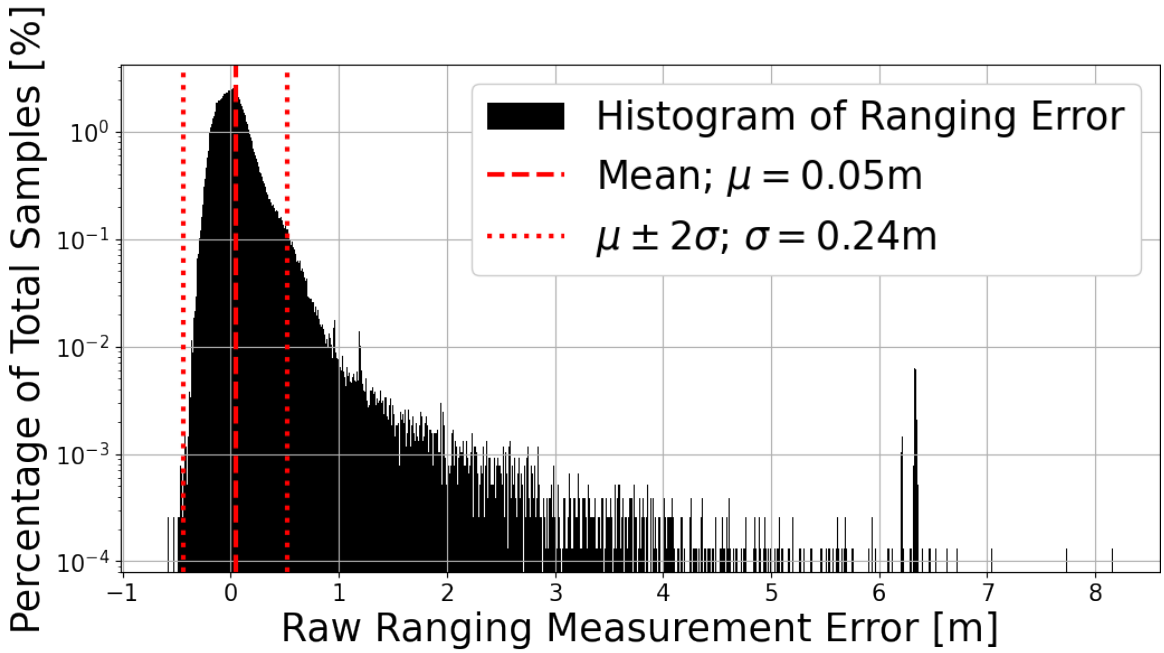
Long Tail Distributions: Contrary to common noise assumptions in Table 2.1, individual UWB ranging errors appear neither zero mean nor Gaussian. Instead, error distributions have long positive tails (see Figure 3-4a). A trend towards positive bias can be attributed to the many ways positive ranging error can be introduced to a UWB measurement (e.g., change of propagation medium, multipath). Furthermore, installing SMA cables between the RF device and antenna adds a consistent positive bias to all measurements.

Sensitivity to Obstruction: Antenna obstruction (i.e., NLOS conditions) introduces positive bias and increased variance to collected ranging measurements (see Figure 5-2b). While the UWB protocol is resilient to multipath and NLOS, UWB ranging is not completely absolved of these concerns. Furthermore, auxiliary metrics, like RSSI, do not appear to meaningfully indicate an obstructed measurement. Ref. [69,70,81,115] address this by rejecting suspect measurements, whether detected via statistical tests, robust loss functions, or hardcoded rejection criteria.

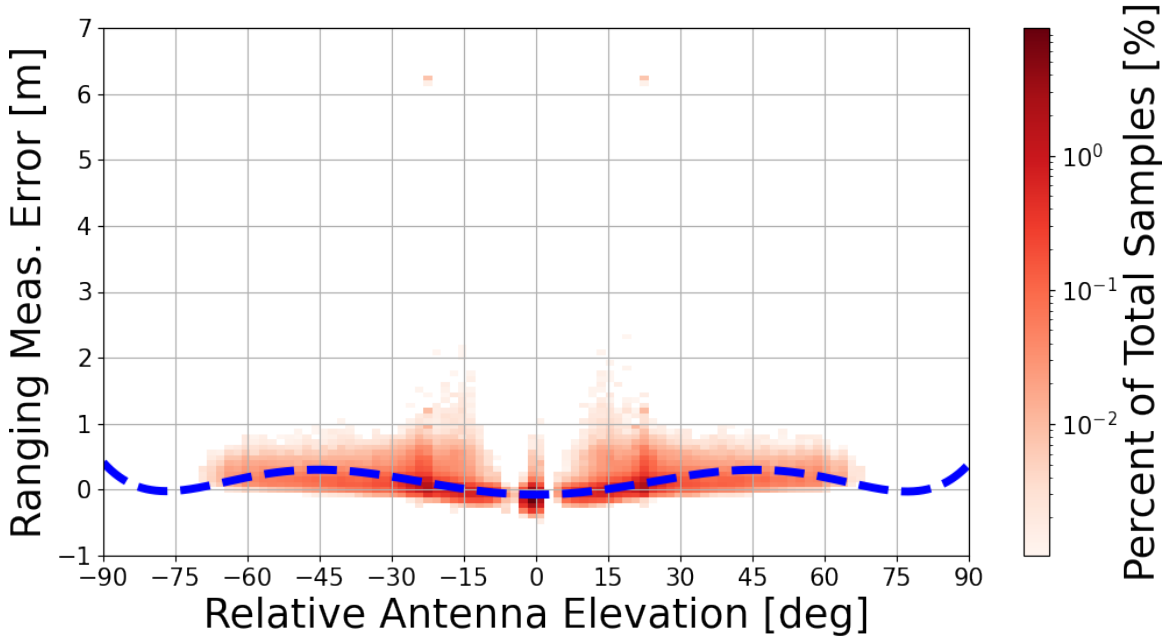
Dependence on Relative Pose: Measurement noise is dependent on relative pose between antennas. The interplay of antenna attenuation patterns is a core concern of RF designers, but gets understandably overlooked by many end users. The LinkTrack P series comes equipped with a standard dipole antenna which we would expect to produce the behavior observed in Figure 3-4b when aligned upright [76] – specifically, approximately uniform performance within an xy-plane (i.e., varying azimuth) and degraded performance outside the ground plane $z = 0$ (i.e., non-zero elevation). We note that both the mean bias and variance change with elevation. A relationship to

respectively, with a 0.5dB minimum resolution.

elevation is noted in [115], although it is attributed to NLOS conditions similar to Figure 5-2b, but [115] chooses to address this by simply omitting range measurements with more than 37° relative elevation (feasible only because of their reliance on other measurements).



(a)



(b)

Figure 3-4: Plots demonstrating the UWB noise and geometry characteristics outlined in Section 3.4. **(a)** Histogram of our entire set of UWB measurements binned by range error. Demonstrates a non-zero mean and long tail (i.e., violates the zero mean Gaussian assumption that is typically used). **(b)** Same data plotted as error with respect to relative elevation showing that the measurement error's mean and variance change significantly with relative elevation. The dotted blue line represents a learned 6-degree polynomial fit of measurement bias.

Chapter 4

UWB-Based Inter-Agent Relative Pose Estimation in 2D

This chapter is based upon [81]. In this chapter we use UWB to demonstrate an *instantaneous* multi-tag approach to relative 2D pose estimation that achieved superior mean position accuracy and competitive performance on other metrics to Cao et al. [108] (the most comparable state-of-the-art work) using only *local* UWB measurements.

4.1 Optimization Formulation

The definitions and terminology presented here are specific to Chapter 4.

Consider two robots, `RobotA` and `RobotB`, denoted as R_A and R_B respectively. Labels are assigned such that R_A is trying to estimate R_B's relative pose with respect to R_A's reference frame. Let R_B's relative pose be described by the state vector $\mathbf{x}$. Each robot has N UWB range sensors, uniquely identified with an integer index from the set $S := \{1, \ldots, N\}$. Assuming R_A and R_B have the same antenna layout, the relative position vector of the kth antenna of a robot with a pose vector of $\mathbf{T}$ is defined as $\boldsymbol{\alpha}_k(\mathbf{T})$. Since we are in R_A's reference frame, $\boldsymbol{\alpha}_i(\mathbf{0})$ gives the position vector of R_A's ith antenna, while $\boldsymbol{\alpha}_j(\mathbf{x})$ gives the position vector of R_B's jth antenna. Thus, let d_{ij} be the distance between R_A's ith antenna and R_B's jth antenna when R_B is

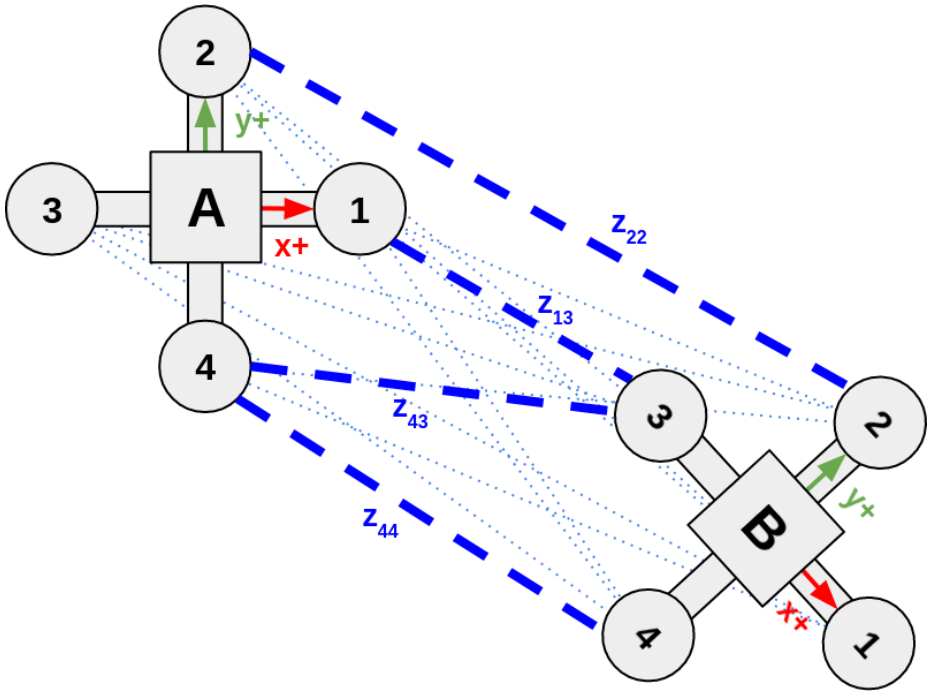


Figure 4-1: Diagram of two agents, `RobotA` and `RobotB`, each with their respective relative coordinate frames and UWB sensors labeled. Relative range measurements are shown between each pair of inter-agent antennas in blue (N.B.: the dashed and dotted blue lines are only differentiated for the sake of visual clarity, they represent the same type of measurements). These 16 unique measurements are denoted as z_{ij} where i and j are the corresponding antenna indices for `RobotA` and `RobotB` respectively.

at $\mathbf{x}$, such that:

$$d_{ij}(\mathbf{x}) = \|\boldsymbol{\alpha}_i(\mathbf{0}) - \boldsymbol{\alpha}_j(\mathbf{x})\|_2 \tag{4.1}$$

For each discrete timestep t, N^2 raw relative range measurements are taken, one per unique pair of R_A and R_B antennas. These measurements are denoted as $z_{ij}^{(t)}$ where i is R_A's ith antenna, j is R_B's jth antenna, and t is the given discrete timestep. Note that as long as we use only our locally collected $z_{ij}^{(t)}$ measurements, we will not require any additional information be exchanged between agents.

4.1.1 Calibrated Range Measurements

As noted in Section 3.3.1, raw UWB relative range measurements $z_{ij}^{(t)}$ are subject to biases between pairs of antennas μ_{ij} as well as general non-Gaussian noise. Thus,

we can improve quality and robustness of our UWB relative range measurement by: (1) Performing a one-time calibration process to measure the consistent measurement bias μ_{ij} between nodes i and j respectively (implementation discussed in Section 4.2.3). (2) Smoothing sequential UWB relative range measurements with a simple moving average filter, which should make the signal more closely track the mean while introducing a slight signal delay.

Let $\hat{z}_{ij}^{(t)}$ be a calibrated relative range measurement,

$$\hat{z}_{ij}^{(t)} = -\mu_{ij} + \frac{1}{W} \sum_{k=0}^{W-1} z_{ij}^{(t-k)}. \tag{4.2}$$

Here we are effectively running a moving average filter over the W most recent z_{ij} range measurements and subtracting out the mean bias μ_{ij}. The choice of W can be selected to trade-off between noise robustness and signal delay, but choosing it to be too large will make rapid relative yaw maneuvers unobservable (i.e., an in-place 360° spin within a single W period would appear as if the sensor did not move).

4.1.2 2D Formulation - Simple Trilateration on (x, y, θ)

Let R_A and R_B be operating in 2D space, making R_B's relative pose be described as $\mathbf{x} = [x, y, \theta]^T$, where x, y, θ are the relative x-coordinate, y-coordinate, and heading (yaw) respectively. Let each robot be equipped with $N \geq 3$ UWB relative range sensors and have their antennas arranged in a non-degenerative layout. Consider the nonlinear least square (NLLS) trilateration pose estimation problem:

$$\underset{\mathbf{x}}{\operatorname{argmin}} \sum_{i \in S} \sum_{j \in S} \left(d_{ij}(\mathbf{x}) - \hat{z}_{ij}^{(t)} \right)^2 \tag{4.3}$$

This formulation can be thought of as our baseline implementation that will be augmented in Section 4.1.3.

4.1.3 2D Formulation - Antenna Weighting

As noted in Section 3.3.2, antennas provide unreliable range measurements when obstructed by another antenna. Since these obstructions are reliably predictable given a specific hardware layout and state, we want to devalue NLLS terms involving an obstructed antenna. This approach is preferable since we are encoding our a priori system knowledge directly into our optimization problem, while more general techniques, such as the use of a Huber loss function, rely on rejecting data based on general outlier criteria. Our approach can be thought of as an analogue to a Maximum Likelihood Estimator formulation where a measurement covariance matrix is specified as a function of state.

Consider the 2π periodic weight function $w(\psi)$, specified here on the angular interval $\psi \in [-\pi, \pi]$, and shown plotted in Figure 3-3a:

$$w(\psi) = \begin{cases} 1 & \rho \leq |\psi| \leq \pi \\ \frac{1}{2}\cos\left(\frac{\pi(\psi+\rho)}{(\rho-\sigma)}\right) + \frac{1}{2} & -\rho < \psi < -\sigma \\ \frac{1}{2}\cos\left(\frac{\pi(\psi-\sigma)}{(\rho-\sigma)} + \pi\right) + \frac{1}{2} & \sigma < \psi < \rho \\ 0 & 0 \leq |\psi| \leq \sigma \end{cases} \tag{4.4}$$

where σ and ρ, related by $0 \leq \sigma \leq \rho \leq \pi$, are predefined constants defining the "stop-band" end angle and "pass-band" begin angle respectively. This function can be thought of as a piecewise step function with a smooth transition between high and low values, in the form of re-scaled and shifted cos segments. Compared to a standard step function, this weight function is differentiable, a useful property for optimization. This function will serve as a primitive for the more specialized weight functions developed in Eqs. 4.6, 4.7, and 4.8.

Although this overall formulation is agnostic to exact sensor numbers and layout, we will develop the remainder of this section using the experimental hardware setup described in Section 4.2.1. Let our 2D robots have an antenna layout as shown in

Figure 4-4a and described by the expression:

$$\boldsymbol{\alpha}_k(\mathbf{T}) = \begin{bmatrix} x + R\cos(\frac{\pi}{2}(k-1)+\theta) \\ y + R\sin(\frac{\pi}{2}(k-1)+\theta) \end{bmatrix} \tag{4.5}$$

where relative pose T has relative components x, y, θ and $R = 0.35$m.

Based on our hardware layout, consider the specialized weight functions $w_{A,i}(\mathbf{x})$ and $w_{B,j}(\mathbf{x})$ for discounting nonlinear least squared terms involving R_A's ith antenna and R_B's jth antenna respectively:

$$w_{A,i}(\mathbf{x}) = w(\arctan(y,x) - \frac{\pi}{2}(i+1)) \tag{4.6}$$

$$w_{B,j}(\mathbf{x}) = w(\theta - \arctan(y,x) - \frac{\pi}{2}(j-1)) \tag{4.7}$$

Note that $w_{A,i}(\mathbf{x})$ and $w_{B,j}(\mathbf{x})$ are just phase shifted versions of $w(\psi)$ based on the components in $\mathbf{x} = [x, y, \theta]^T$. As shown in Figures 3-3b-3-3e, these weight functions are aligned so that they devalue measurements involving antennas on the "far" side of either robot. See Figure 4-2 for a visual example.

For convenience, we can then combine $w_{A,i}(\mathbf{x})$ and $w_{B,j}(\mathbf{x})$ into a single weighting $w_{ij}(\mathbf{x}) = w_{A,i}(\mathbf{x}) \cdot w_{B,j}(\mathbf{x})$. Augmenting Eq. 4.3 with this weighting yields

$$\underset{\mathbf{x}}{\operatorname{argmin}} \sum_{i\in\{1,\ldots,N\}} \sum_{j\in\{1,\ldots,N\}} w_{ij}(\mathbf{x}) \Big(d_{ij}(\mathbf{x}) - \hat{z}_{ij}^{(t)} \Big)^2. \tag{4.8}$$

When using the parameters $\sigma = 30^\circ$ and $\rho = 90^\circ$, this means that, at any given time, at most 7 of 16 measurements between a pair of robots can be ignored (i.e. four measurements coming from R_A's ignored antenna, four measurements coming from R_B's ignored antenna, with one measurement being ignored twice). Although with reliable measurements only three antennas per agent are needed to have a fully observable 2D system, the redundant fourth antenna allows us to entirely ignore the obstructed measurements within a given pair of agents while maintaining full 2D observability.

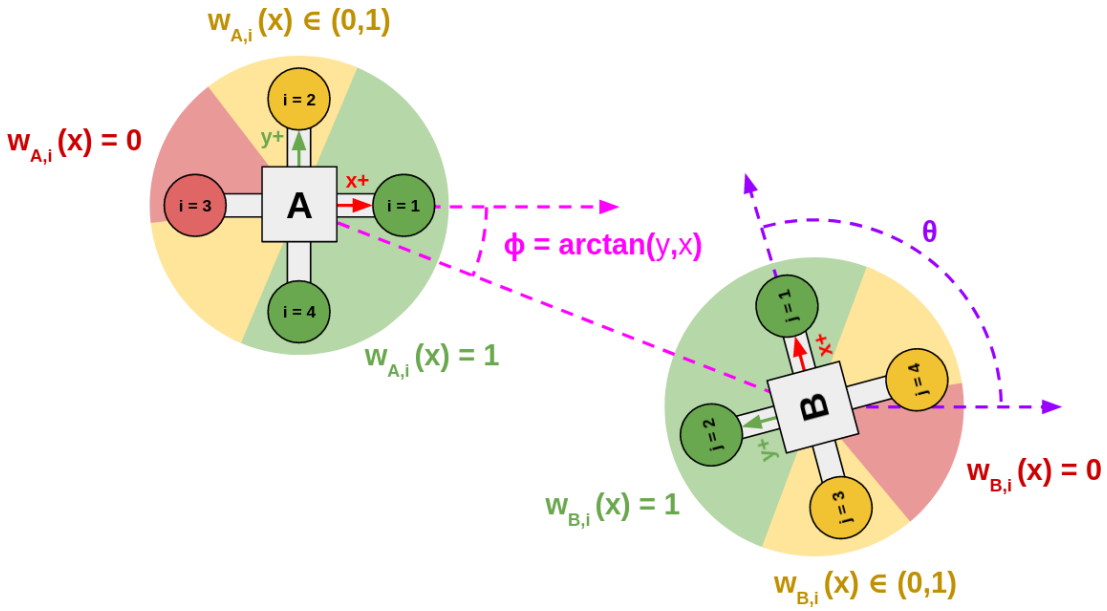
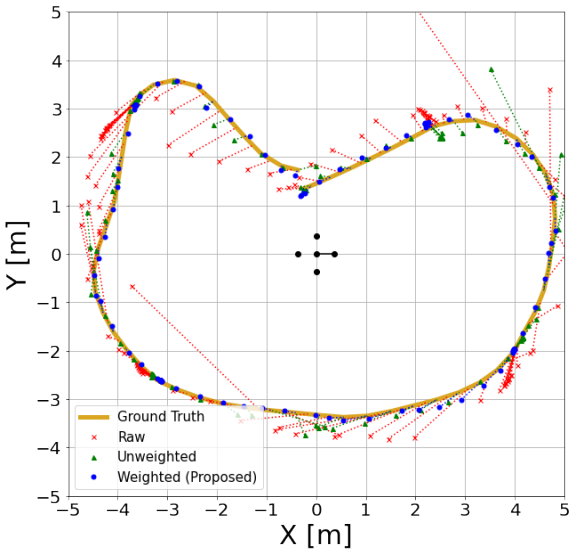


Figure 4-2: Visualization of weighting functions $w_{A,i}(\mathbf{x})$ and $w_{B,i}(\mathbf{x})$ when $\mathbf{x} = [3, -1, 100°]^T$. Here, the green, yellow, and red colors correspond to the colored weight segments of $w(\psi)$ in Figure 3-3a. In this scenario, optimization terms involving R_A's $i = 3$ antenna are ignored, while terms involving R_A's $i = 2$ antenna and/or R_B's $j = 3$ or $j = 4$ antennas are devalued.

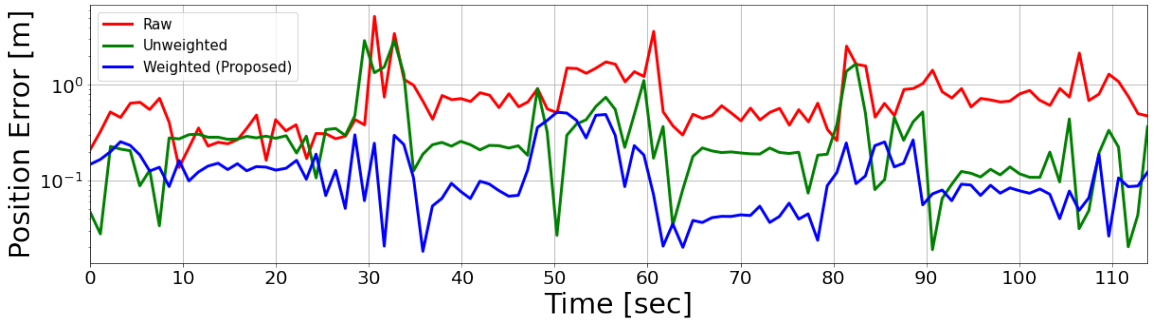
4.1.4 Optimization Initialization

When computationally solving the nonlinear problems in Eq. 4.3 or 4.8, the optimizer requires initial relative pose $\mathbf{x}_0$. Although we cannot guarantee convexity on either equation, as sufficiently erroneous measurements $z_{ij}^{(t)}$ can make either equation behave irregularly, typically we observe that optimizing Eq. 4.3 yields the same result regardless of selected the $\mathbf{x}_0$, while Eq. 4.8 very much depends on the selected $\mathbf{x}_0$. To address this, in practice we perform a two staged optimization, i.e. we initialize Eq. 4.8 with $\mathbf{x}_{res}$, which is the result of solving Eq. 4.3 when initialized at $\mathbf{x}_0 = \mathbf{0}$.

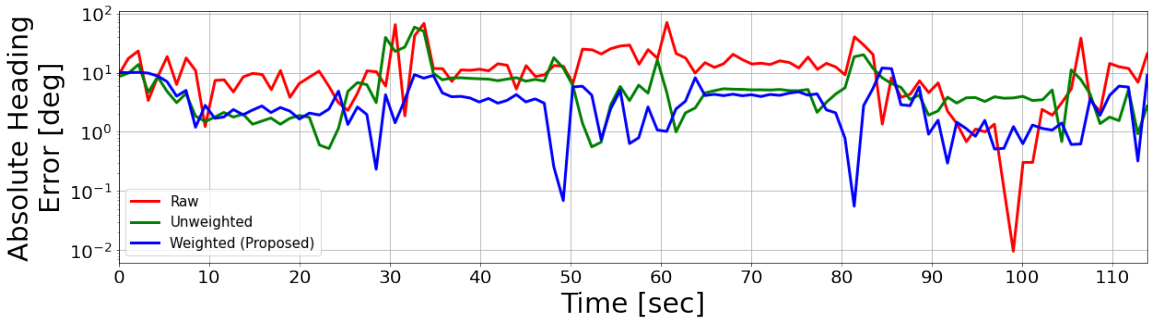
To demonstrate the utility of this two-step process, a simulation was written in which ground truth state $\mathbf{x}_{gt}$ was sampled from the uniform distribution $[\mathcal{U}_{[-5,5]}, \mathcal{U}_{[-5,5]}, \mathcal{U}_{[0,360]}]^T$ such that $\|[x, y]^T\|_2 \geq 1$ and z_{ij} was sampled by adding $\mathcal{N}(0, 0.2)$ to the ground truth distances. Table 4.1 shows the results after 10000 trials, which highlight that solving Eq. 4.3 effectively finds the same local minimum whether initialized with $\mathbf{0}$ or $\mathbf{x}_{gt}$, but Eq. 4.8 often finds the "wrong" local minimum if initialized with $\mathbf{0}$ and the "correct" local minimum if initialized with $\mathbf{x}_{res}$ (i.e., the proposed two-step process).



(a) Estimated trajectory of R_B w.r.t. R_A



(b) Log-Scale Position Error vs. Time



(c) Log-Scale Absolute Heading Error vs. Time

Figure 4-3: Results from experimental trial where R_B is manually driven in a kidney bean shape while pausing briefly throughout the trajectory. Within the trial, R_B has a maximum 1m/s velocity and 1rad/s rotation rate. To prevent visual clutter, only estimations from the `Raw`, `Unweighted`, and `Weighted (Proposed)` algorithms are shown alongside the mocap ground truth. Where applicable the parameters $\sigma = 30°$, $\rho = 90°$, and $W = 50$ (i.e. 1 sec of samples since measurements occur at 50Hz) are used. (a) Shows each algorithm's predicted position estimate (at 1Hz) and draws an error line between each estimated point and its corresponding ground truth point. (b) Shows each algorithm's log-scale positional error verses time for the given trial. (c) Similarly shows each algorithm's log-scale absolute heading error for the given trial. Examining the plots (b) and (c), we see `Weighted (Proposed)` provides the best estimate of position and heading a vast majority of the times within the trajectory. See specific calculated mean, max, and standard deviation values, as well as all other results, in Tables 5.2 and 4.3.

Table 4.1: Comparison of Mean Difference Predicted Position (MDPP) and Mean Difference Predicted Absolute Heading (MDPAH) over 10,000 trials for different optimization equations initialized with various $\mathbf{x}_0$. Note that MDPP and MDPAH are indicative of how the optimization results differ between the selected initial conditions.

Opt. Eq.	Compared $\mathbf{x}_0$	MDPP [m]	MDPAH [deg]
Eq. 4.3	$\mathbf{0}$ vs $\mathbf{x}_{gt}$	0.002	0.067
Eq. 4.8	$\mathbf{0}$ vs $\mathbf{x}_{gt}$	1.109	41.704
Eq. 4.8	$\mathbf{x}_{res}$ vs $\mathbf{x}_{gt}$	0.018	0.884

4.2 Experimental Results

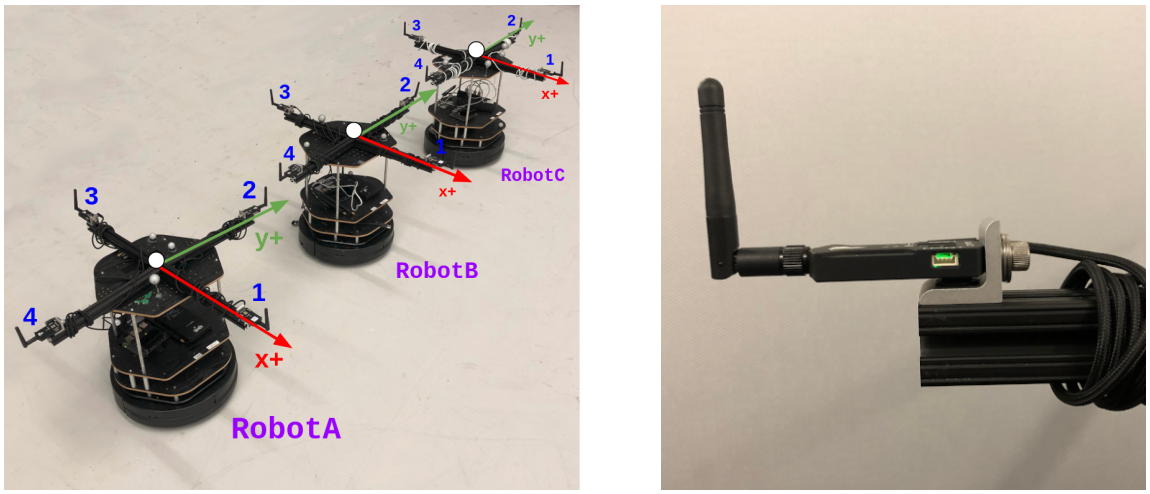
4.2.1 Robot and UWB Setup

All experiments were conducted with a set of robots (see Figure 4-4a) each equipped with a Turtlebot2 base, NVIDIA Jetson Nano, and four Nooploop UWB sensors. UWB sensors are positioned $R = 0.35$m from the center of the robot to the upright UWB antenna.[1] The mounting bracket is made of aluminum and UWB modules are mounted radially for antenna clearance (see Figure 4-4b). UWB nodes are numbered as in Figure 4-1. All tests were performed in a motion capture space for high-precision ground truth pose.

4.2.2 Experimental Setup & Implementation

Experimental trials were conducted using the hardware in Section 4.2.1 and the algorithms in Section 4.1. All code was written using Python 3 and ROS [123]. Optimization was done using `scipy.optimize.minimize` with the `trust-constr` method [121]. Calibrated measurements $\hat{z}_{ij}^{(t)}$ are generated at 50Hz, the sensor operating rate, and sampled by the optimization code as needed. The optimization implementation runs in real-time at approximately 10Hz on an Intel i7-6700 with 8GB of RAM. This could easily be sped up by switching to a C++ implementation.

[1] As the robustness and accuracy of the estimated pose computed from trilateration is positively correlated with the length of the sensor baseline the R separation was selected to closely resemble the arm length of a medium-sized quadrotor, in anticipation for future work, as well as allowing for a fair comparison to the results of [108], who tested a similar baseline.



(a) (b)

Figure 4-4: (a) Photo of our experimental robots. Each is comprised of a Turtlebot2 base, a Jetson Nano, and four Nooploop UWB sensors. Robots are manually controlled with a wireless Logitech Gamepad. (b) A close up showing how UWB sensors are attached radially from the aluminum mounting bracket. This is done to ensure antenna clearance and to avoid direct contact with metal.

Collected UWB data is post-processed with several alternative algorithms to compare results. When applicable, we use parameters $\sigma = 30°$, $\rho = 90°$, and $W = 50$. Specifically, these algorithms are:

- `Raw`: Optimizes Eq. 4.3, but uses the "raw" $z_{ij}^{(t)}$ measurement instead of the calibrated $\hat{z}_{ij}^{(t)}$.
- `Shift only`: Refers to Eq. 4.3, but uses the $z_{ij}^{(t)} - \mu_{ij}$ instead of the calibrated $\hat{z}_{ij}^{(t)}$.
- `MovingAvg only`: Refers to Eq. 4.3, but uses $\frac{1}{W}\sum_{k=0}^{W-1} z_{ij}^{(t-k)}$ instead of the calibrated $\hat{z}_{ij}^{(t)}$.
- `Unweighted`: Refers to Eq. 4.3 as written.
- `Weighted (Proposed)`: Optimizes Eq. 4.8.

Comparing the results of `Raw`, `Shift only`, and `MovingAvg only` with `Unweighted` will clearly show the benefits of using the calibrated $\hat{z}_{ij}^{(t)}$. Similarly, comparing the results of `Unweighted` and `Weighted` shows the benefits of adding the NLLS weighting $w_{ij}(\mathbf{x})$.

4.2.3 Calibration

Calibration means μ_{ij} were found by placing R_A and R_B a known distance apart (3m) and rotating R_B in place at approximately 60°/sec. After R_B completed a full revolution, R_A was rotated approximately 30° before R_B resumed rotating. The process took approximately 90 seconds and, afterwards, μ_{ij} values were calculated by averaging $\tilde{z}_{ij}$ while omitting regions of antenna obstructions spikes, similar to Figures 3-3b-3-3e. The following μ_{ij} values were computed:

$$\begin{array}{llll} \mu_{11} = 0.268 & \mu_{12} = 0.266 & \mu_{13} = 0.277 & \mu_{14} = 0.230 \\ \mu_{21} = 0.093 & \mu_{22} = 0.112 & \mu_{23} = 0.227 & \mu_{24} = 0.188 \\ \mu_{31} = 0.046 & \mu_{32} = 0.018 & \mu_{33} = 0.170 & \mu_{34} = 0.078 \\ \mu_{41} = 0.041 & \mu_{42} = 0.065 & \mu_{43} = 0.178 & \mu_{44} = 0.095 \end{array}$$

Although these values are specific to our hardware, they show the significance of these biases given the current operating scale as well as how much variation there is between pairs of sensors (i.e., as much as 25.9cm between our observed best and worst pair).

4.2.4 Trials

Several experiments were run with different trajectories and durations. In all trials the Turtlebots were driven at the maximum velocity and rotation rate (1m/s of 1rad/s respectively). The specific trials were:

- `rot-cw/rot-ccw`: R_A is kept stationary. R_B is placed a fixed distance away and rotated clockwise/counter-clockwise in place.

- `traj-cw/traj-ccw`: R_A is kept stationary. R_B is manually driven in a circular trajectory about R_A in a clockwise/counter-clockwise direction.

- `kidney-bean`: R_A is kept stationary. R_B is manually driven in a kidney bean shape while pausing briefly throughout the trajectory (See Figure 4-3).

- `box`: R_A is kept stationary. R_B is manually driven in an approximately 8m by 6m rectangle.

- `both-move`: Both R_A and R_B are manually driven arbitrarily within a 10m by 10m space without getting within 1m of each other.

All results are compiled into Tables 5.2 and 4.3, showing the mean, max, and standard deviation of the position and absolute heading errors respectively.

4.2.5 Interpreting Results

The results in Tables 5.2 and 4.3 show that `Weighted` and `Unweighted` consistently outperform the other approaches; this makes clear the advantage of the calibrated $\hat{z}_{ij}^{(t)}$ over the raw $z_{ij}^{(t)}$ or the individual mean shift/moving average corrections. Additionally, `Weighted` consistently outperforms `Unweighted` in mean positional error. Next, while `Unweighted` outperforms `Weighted` in a few select metrics in the simpler `rot-*` and `traj-*` trials, this is only by relatively small margins (i.e., at most 4cm or 5.1° respectively). When considering the more challenging `kidney-bean` and `box` trajectories, we see `Weighted` substantially outperforms `Unweighted` in all metrics. Finally, when examining `both-move`, we see `Weighted` outperforms all other methods in the three positional error metrics as well as mean absolute heading error. The spike observed in `Weighted`'s other two heading metrics appears to be the result of rapid relative yawing, possible when R_A and R_B yaw simultaneously at max speed in opposite directions.

4.2.6 Comparison to Literature

Our `box` trial was designed so that it can be compared to a similar experiment in [108]. Note that both experiments used two Turtlebots (one stationary, one moving), each equipped with four Nooploop LinkTrack P UWB modules separated by an approximately 70cm baseline, and performed relative pose estimation while traversing an approximately 8m by 6m rectangle. When comparing our results to the reported

Table 4.2: Evaluation of positional error in meters between algorithms and datasets.

Position Error [m]	Scenario																				
	rot-cw			rot-ccw			traj-cw			traj-ccw			kidney-bean			box			both-move		
Method	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**
Raw	0.44	0.98	0.16	0.38	0.65	0.08	0.98	5.71	0.88	0.59	1.54	0.30	0.76	5.20	0.68	0.93	4.99	0.70	0.82	3.50	0.54
Shift only	0.27	0.72	0.15	0.19	0.46	0.09	0.70	5.27	0.89	0.27	1.27	0.25	0.47	4.77	0.73	0.44	4.61	0.69	0.42	3.53	0.48
MovingAvg only	0.35	0.53	0.05	0.34	0.47	0.05	0.87	5.08	0.65	0.55	1.41	0.27	0.68	3.40	0.47	0.91	3.57	0.59	0.78	2.64	0.42
Unweighted (Eq. 4.3)	0.14	0.25	**0.02**	0.13	**0.19**	**0.02**	0.62	4.37	0.70	0.22	1.02	0.19	0.36	2.91	0.44	0.45	3.29	0.54	0.37	2.87	0.42
Weighted (Eq. 4.8) **[Proposed]**	**0.09**	**0.21**	0.06	**0.09**	0.21	0.05	**0.20**	**1.41**	**0.18**	**0.21**	**0.86**	**0.16**	**0.13**	**0.52**	**0.11**	**0.21**	**0.96**	**0.17**	**0.29**	**2.70**	**0.32**

Table 4.3: Evaluation of absolute heading error in degrees between algorithms and datasets.

Abs Heading Error [deg]	Scenario																				
	rot-cw			rot-ccw			traj-cw			traj-ccw			kidney-bean			box			both-move		
Method	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**
Raw	9.82	21.81	5.53	10.10	22.43	6.05	16.82	90.29	15.09	12.44	35.78	9.38	12.06	70.33	12.17	12.11	65.86	10.15	11.35	72.69	9.33
Shift only	6.38	16.44	4.19	6.45	14.20	4.09	12.40	90.41	15.52	6.99	31.55	6.44	8.10	80.20	12.72	6.34	64.33	9.76	6.97	78.23	9.46
MovingAvg only	7.05	14.70	4.63	7.37	17.59	4.68	14.48	77.47	11.49	11.27	35.41	8.70	10.32	57.96	8.73	11.55	50.65	9.01	11.11	55.54	**7.15**
Unweighted (Eq. 4.3)	**3.31**	**8.28**	**1.99**	**2.88**	**9.78**	**2.13**	10.77	**72.66**	11.65	**5.44**	**25.92**	**4.79**	6.15	58.70	8.38	5.99	51.71	7.91	6.38	**53.30**	7.64
Weighted (Eq. 4.8) **[Proposed]**	4.12	10.78	3.11	4.43	10.53	3.34	**5.88**	75.56	**8.09**	5.84	31.01	6.05	**3.95**	**11.97**	**2.98**	**4.22**	**13.31**	**3.03**	**5.45**	104.00	9.97

results in [108], our proposed approach, `Weighted`, achieved slightly better position error (0.21m vs 0.25m), but with slightly worse mean heading error (4.22° vs 2.02°) and standard deviations (our 0.17m and 3.03° vs their 0.09m and 1.47° respectively). Thus, our approach is competitive with the performance in [108], but that work assumes access to **continuously transmitted odometry estimates, whereas our approach does not**.

Chapter 5

UWB-Based Inter-Agent Relative Pose Estimation in 3D

This chapter is based upon [82]. In this chapter we use UWB to demonstrate an *instantaneous* multi-tag approach to relative 3D. By equipping each agent with multiple UWB antennas, our approach addresses these concerns by using only *locally* collected UWB range measurements, *a priori* state constraints, and detections of when said constraints are violated. Leveraging our learned mean ranging bias correction, we gain a 19% positional error improvement giving us experimental mean absolute position and heading errors of 0.24m and 9.5° respectively. When compared to other state-of-the-art approaches, our work demonstrates improved performance over similar systems [65], while remaining competitive with methods that have significantly higher communication costs.

5.1 System Design

Based on the noise characteristics (Section 3.4.1) and underlying geometry (Section 1.2.3), we propose the system design shown in Figure 1-1 for our multirotor drones. Specifically, each agent is equipped with 6 ranging antennas (attached 0.31m from the center at the end of each propeller arms) to maximize the baselines and improve DOP geometry. Since there are 6 coplanar antennas, this is an over-constrained problem

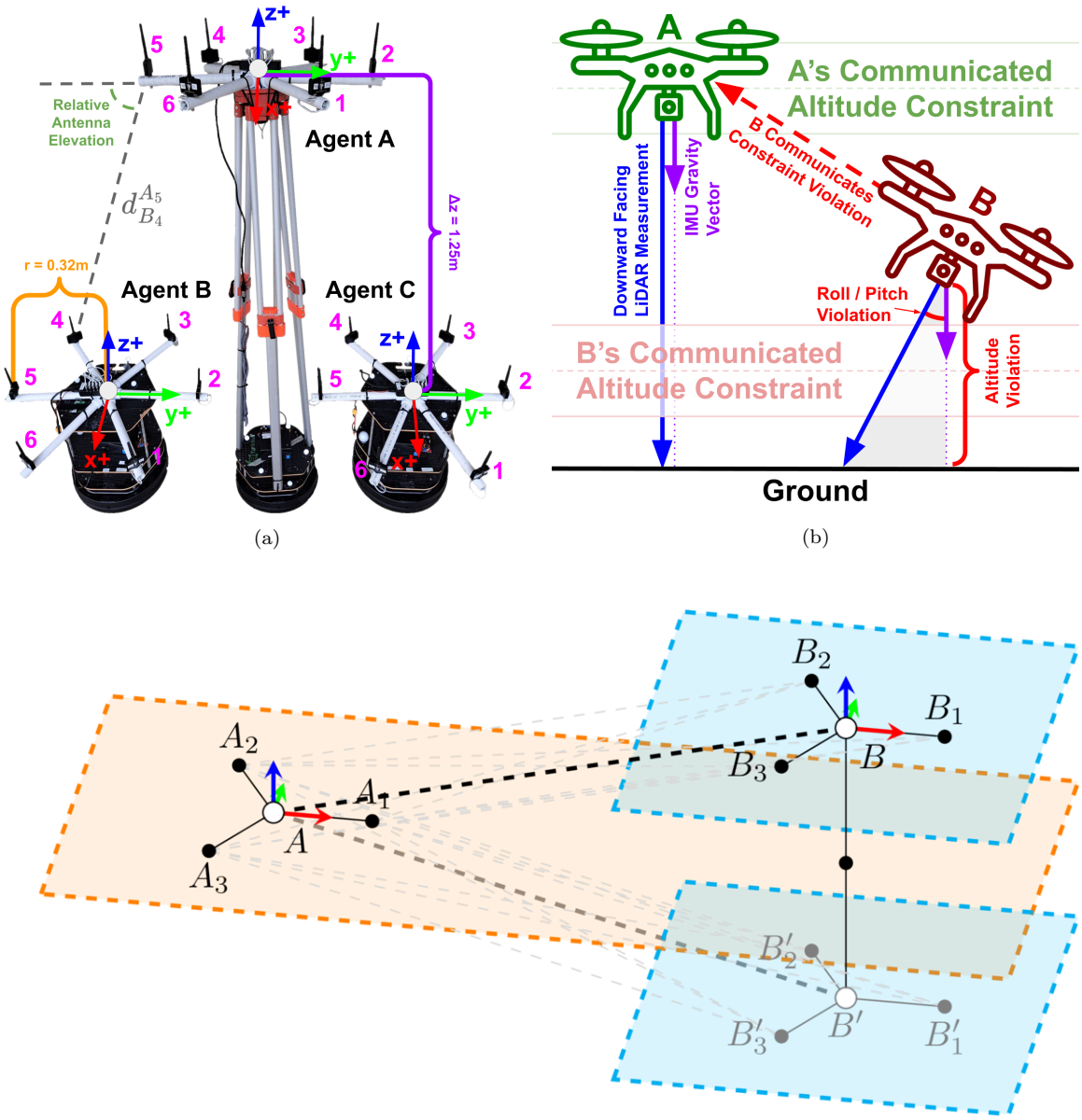
that still has a $\pm z$ ambiguity across the $z = 0$ plane. Although an additional nonplanar antenna could be added to each agent, it is challenging to get a large z-baseline without impacting the flight characteristics (i.e., heavy/awkward configuration). Instead, we note that altitude/roll/pitch can be measured within the flat-ground world frame *locally* via altimeter or downward facing LiDAR and onboard IMU. In many drone applications, it is common for multirotors to maintain roll/pitch near 0° and a constant altitude. Thus, we can specify a minimalist communication protocol that shares these intended constraints *a priori* and then monitors them *locally* to ensure they are satisfied. *One-off* communication would then only occur if a constraint is *locally* detected to have been violated/changed (Figure 5-1b). Additionally, having a constant z constraint helps mitigate the increased variance associated with relative elevation (Figure 3-4b).

5.2 Preliminaries

5.2.1 Pose Parameterization

Given a set of 3D reference frames $\mathcal{F}$, consider any pair of frames $A \in \mathcal{F}$ and $B \in \mathcal{F}$. Let $\mathbf{T}_B^A \in SE(3)$ be the relative pose between frames A and B (i.e., transforms a point from B's frame into A's frame). We note $\mathbf{T}_B^A$ is a 6-DoF value that can be equivalently parameterized as the 2-tuple $\langle \mathbf{R}_B^A, \mathbf{t}_B^A \rangle$ or 6-tuple $\langle x_B^A, y_B^A, z_B^A, \alpha_B^A, \beta_B^A, \gamma_B^A \rangle$, where:

- $\mathbf{R}_B^A \in SO(3)$ is the relative rotation between A and B parameterized by the 3-tuple of Euler angles $\langle \alpha, \beta, \gamma \rangle$, where $\alpha_B^A \in [-180^\circ, 180^\circ]$ is relative roll, $\beta_B^A \in [-90^\circ, 90^\circ]$ is relative pitch, and $\gamma_B^A \in [-180^\circ, 180^\circ]$ is relative yaw. The use of Euler angles over quaternions is motivated by the direct roll/pitch constraints to be introduced Equation 5.7. We define $\mathbf{R}(\alpha, \beta, \gamma)$ as the $\mathbb{R}^{3\times3}$ rotation matrix with respect to $\mathbf{R}_x(\alpha)$, $\mathbf{R}_y(\beta)$, $\mathbf{R}_z(\gamma)$, the $\mathbb{R}^{3\times3}$ rotation matrices about the x, y, z axes respectively. Thus $\mathbf{R}_B^A$ can be equivalently represented as a $\mathbb{R}^{3\times3}$ rotation



(a) (b) (c)

Figure 5-1: **(a)** Annotated diagram of three UGV agents used in Section 5.4.2. UGV agents are designed with the same baseline as the UAV in Figure 1-1, making them comparable surrogates to the UAV and its experiments in Section 5.4.1. **(b)** Diagram demonstrating agent B *locally* detecting violations in its *a priori* constraints, triggering a *one-off* communication. **(c)** Demonstrates how three (or more) ranging antennas within a single plane produce a pair of ambiguous solutions (i.e., if all antennas are in the base's plane $z = 0$, while the target's x and y coordinates are fully observable, the target's altitude has an ambiguity between $\pm z$).

matrix in the following ways:

$$\mathbf{R}_B^A \triangleq \mathbf{R}(\alpha_B^A, \beta_B^A, \gamma_B^A) \triangleq \mathbf{R}_z(\gamma_B^A)\mathbf{R}_y(\beta_B^A)\mathbf{R}_x(\alpha_B^A)$$

- $\mathbf{t}_B^A \in \mathbb{R}^3$ is a relative translation vector between A and B parameterized by the 3-tuple $\langle x_B^A, y_B^A, z_B^A \rangle$. Thus, $\mathbf{t}_B^A$ can be equivalently representedas 3 in the following ways:

$$\mathbf{t}_B^A \triangleq \mathbf{t}(x_B^A, y_B^A, z_B^A) \triangleq \begin{bmatrix} x_B^A & y_B^A & z_B^A \end{bmatrix}^\top$$

5.2.2 Rigid Body Transformations

For any frame $F \in \mathcal{F}$, let $\mathbf{p}^F \in \mathbb{R}^3$ be a point in frame F. Given $\mathbf{T}_B^A$ (i.e., the relative pose between frames A and B), points $\mathbf{p}^A$ and $\mathbf{p}^B$ (i.e., the same point but in A's or B's frame respectively) are related by $\mathbf{p}^A = \mathbf{R}_B^A \mathbf{p}^B + \mathbf{t}_B^A$, which can equivalently and succinctly be written in homogeneous coordinate notation [33] as $\breve{\mathbf{p}}^A = \mathbf{T}_B^A \breve{\mathbf{p}}^B$, where $\mathbf{T}_B^A \in \mathbb{R}^{4\times4}$ and $\breve{\mathbf{p}}^F \in \mathbb{R}^4$ are defined as:

$$\mathbf{T}_B^A \triangleq \begin{bmatrix} \mathbf{R}_B^A & \mathbf{t}_B^A \\ \mathbf{0}^\top & 1 \end{bmatrix} \qquad \breve{\mathbf{p}}^F \triangleq \begin{bmatrix} \mathbf{p}^F \\ 1 \end{bmatrix}$$

Thus $\mathbf{T}_B^A$ is equivalently represented in the following ways:

$$\begin{aligned} \mathbf{T}_B^A &\triangleq \mathbf{T}(x_B^A, y_B^A, z_B^A, \alpha_B^A, \beta_B^A, \gamma_B^A) \\ &\triangleq \begin{bmatrix} \mathbf{R}_B^A & \mathbf{t}_B^A \\ \mathbf{0}^\top & 1 \end{bmatrix} \triangleq \begin{bmatrix} \mathbf{R}(\alpha_B^A, \beta_B^A, \gamma_B^A) & \mathbf{t}(x_B^A, y_B^A, z_B^A) \\ \mathbf{0}^\top & 1 \end{bmatrix} \end{aligned}$$

5.3 Technical Approach

5.3.1 Local Robot Definitions

Let there be some 3D world frame W defined with respect to the level ground and gravity vector (i.e., $z = 0$ is the "floor" and gravity faces down). Consider $\mathcal{R}$, a set

of $N_{\mathcal{R}}$ robots operating in W. Each robot $R \in \mathcal{R}$, has N_R relative ranging antennas rigidly affixed to R's body, where R_k denotes R's kth antenna. Additionally, for each R let there be an *a priori* and *static* 3-tuple of known information $\mathcal{I}_R = \langle \hat{\boldsymbol{\kappa}}_R, \check{\boldsymbol{\kappa}}_R, \mathcal{P}_R \rangle$, where:

- $\hat{\boldsymbol{\kappa}}_R = \begin{bmatrix} \hat{z}_R^W & \hat{\alpha}_R^W & \hat{\beta}_R^W \end{bmatrix}^\top$ is a vector of *constant* global state constraints on R with respect to world frame W. In other words, $\hat{z}_R^W$ is the commanded altitude, while $\hat{\alpha}_R^W$ and $\hat{\beta}_R^W$ are commanded relative roll and pitch.

- $\check{\boldsymbol{\kappa}}_R = \begin{bmatrix} \check{z}_R^W & \check{\alpha}_R^W & \check{\beta}_R^W \end{bmatrix}^\top$ is a vector of *constant* absolute (i.e., $\pm$) tolerances on global constraints $\hat{\boldsymbol{\kappa}}_R$.

- $\mathcal{P}_R = \{\mathbf{T}_{R_1}^R, \mathbf{T}_{R_2}^R, \ldots, \mathbf{T}_{R_{N_R}}^R\}$ is an ordered set of N_R *constant* relative poses from R's body frame to R's kth antenna's frame. Additionally, from the information in $\mathbf{T}_{R_k}^R$, we can succinctly denote the point coordinate of R's kth antenna in R's frame as $\mathbf{p}_k^R$.

Since W is defined with respect to a level ground and gravity vector, the parameters $\langle z_R^W, \alpha_R^W, \beta_R^W \rangle$ – unlike $\langle x_R^W, y_R^W, \gamma_R^W \rangle$ – are *instantaneously* observable and *directly* measurable from R (e.g., via altimeter or downward facing LiDAR and onboard IMU). Thus, the directly measured values $\tilde{\boldsymbol{\kappa}}_R = \begin{bmatrix} \tilde{z}_R^W & \tilde{\alpha}_R^W & \tilde{\beta}_R^W \end{bmatrix}^\top$ can be continuously monitored by R *locally* (i.e., without relying on swarm communication). After an initial *one-off* transmission of $\mathcal{I}_R$ to all agents $\mathcal{R}$, agent R only needs to transmit again in the event of (1) wanting to change its commanded altitude/roll/pitch $\hat{\boldsymbol{\kappa}}_R$, or (2) it locally observes a violation of its previously communicated constraints, specifically:

$$\underbrace{|\hat{\boldsymbol{\kappa}}_R - \tilde{\boldsymbol{\kappa}}_R|}_{\text{element-wise abs deviation}} \leq \underbrace{\check{\boldsymbol{\kappa}}_R}_{\text{constraint tolerance}} \tag{5.1}$$

where $|\cdot|$ and $\leq$ are performed element-wise. In other words, all agents $\mathcal{R}$ can treat $\hat{\boldsymbol{\kappa}}_R$ as *known* states during estimation, unless R communicates otherwise.

5.3.2 Inter-Agent Robot Definitions

Consider any arbitrary pair of robots $A \in \mathcal{R}$ and $B \in \mathcal{R}$. With a slight abuse of notation, let A and B also denote the robots respective reference frames, making $\mathbf{T}_B^A$ denote the relative pose between robots A and B. From our known information tuples $\mathcal{I}_A$ and $\mathcal{I}_B$ we know each agent has N_A and N_B antennas at relative poses $\mathcal{P}_A$ and $\mathcal{P}_B$ respectively. This in turn allows us to define our observation model as just a function of $\mathbf{T}_B^A$. In other words, the distance between A's ith antenna and B's jth antenna is defined as:

$$d_{B_j}^{A_i}(\mathbf{T}_B^A) \triangleq ||\mathbf{T}_B^A \breve{\mathbf{p}}_j^B - \breve{\mathbf{p}}_i^A||_2 \tag{5.2}$$

where $d_{B_j}^{A_i}(\mathbf{T}_B^A) \in \mathbb{R}_{\geq 0}$, as distances are non-negative, and $||\cdot||_2$ denotes the L_2 norm. Similarly, let $\tilde{d}_{B_j}^{A_i}$ be the current (i.e., most recent) noisy measurement between A's ith antenna and B's jth antenna as measured *locally* by A.[1] For convenience, $\tilde{\mathbf{d}}_B^A \in \mathbb{R}^{N_A N_B}$ denotes the stacked vector of all current pairwise $\tilde{d}_{B_j}^{A_i}$ measurements.

5.3.3 Noise Model Definition

Let $\mathbf{T}_{B_j}^{A_i}$ denote the relative pose between A's ith antenna and B's jth antenna. Given a known $\mathcal{P}_A$ and $\mathcal{P}_B$, and an arbitrary $\mathbf{T}_B^A$, we have:

$$\mathbf{T}_{B_j}^{A_i} = (\mathbf{T}_{A_i}^A)^{-1}\mathbf{T}_B^A\mathbf{T}_{B_j}^B = \mathbf{T}_A^{A_i}\mathbf{T}_B^A\mathbf{T}_{B_j}^B \tag{5.3}$$

so that $\mathbf{T}_{B_j}^{A_i}$ is a function of $\mathbf{T}_B^A$ given known $\mathcal{P}_A$ and $\mathcal{P}_B$. Let $\bar{d}(\mathbf{T}_{B_j}^{A_i}) \in \mathbb{R}$ be some provided (e.g., learned) sensor mean bias model as a function of $\mathbf{T}_{B_j}^{A_i}$ (i.e., antenna measurement bias is modeled as some function of the relative position and/or orientation between a pair of ranging antennas). We can then equivalently define:

$$\bar{d}_{B_j}^{A_i}(\mathbf{T}_B^A) \triangleq \bar{d}(\mathbf{T}_A^{A_i}\mathbf{T}_B^A\mathbf{T}_{B_j}^B) \tag{5.4}$$

[1] Note that while $d_{B_j}^{A_i}(\mathbf{T}_B^A) = d_{A_i}^{B_j}(\mathbf{T}_A^B)$ (i.e., the observation model is symmetric), since each agent *locally* and *separately* measures antenna distances, measurement noise makes it that generally $\tilde{d}_{B_j}^{A_i} \neq \tilde{d}_{A_i}^{B_j}$.

where $\bar{d}_{B_j}^{A_i}(\mathbf{T}_B^A) \in \mathbb{R}$ is a state-dependent mean bias estimate for measurement $\tilde{d}_{B_j}^{A_i}$.

5.3.4 Optimization Definition

Using our current measurements $\tilde{d}_{B_j}^{A_i}$, measurement bias correction function $\bar{d}_{B_j}^{A_i}(\mathbf{T}_B^A)$, and observation model $d_{B_j}^{A_i}(\mathbf{T}_B^A)$, we can formulate the error residual function with respect to antenna pair A_i and B_j as:

$$e_{B_j}^{A_i}(\mathbf{T}_B^A) \triangleq \underbrace{\left(\tilde{d}_{B_j}^{A_i} - \bar{d}_{B_j}^{A_i}(\mathbf{T}_B^A)\right)}_{\text{bias adjusted measurement}} - \underbrace{d_{B_j}^{A_i}(\mathbf{T}_B^A)}_{\text{expected measurement}} \tag{5.5}$$

Using our error residual function $e_{B_j}^{A_i}(\mathbf{T}_B^A)$, and some loss function ℓ, we can calculate an instantaneous estimate of $\mathbf{T}_B^A$ by minimizing the sum of loss of all error residuals for all inter-agent antenna pairs with respect to $\mathbf{T}_B^A$. That is:

$$\min_{\mathbf{T}_B^A \in SE(3)} \sum_{i=1}^{N_A} \sum_{j=1}^{N_B} \ell\left(e_{B_j}^{A_i}(\mathbf{T}_B^A)\right) \tag{5.6}$$

Recall that $\mathbf{T}_B^A$ is a 6-DoF variable parameterized by $\langle x_B^A, y_B^A, z_B^A, \alpha_B^A, \beta_B^A, \gamma_B^A \rangle$. From known information tuples $\mathcal{I}_A$ and $\mathcal{I}_B$, we have $\hat{\boldsymbol{\kappa}}_A$ and $\hat{\boldsymbol{\kappa}}_B$, which provide us with constraints $\langle \hat{z}_A^W, \hat{\alpha}_A^W, \hat{\beta}_A^W \rangle$ and $\langle \hat{z}_B^W, \hat{\alpha}_B^W, \hat{\beta}_B^W \rangle$. Furthermore, since these constraints are with respect to a common world frame W, see that: $\hat{z}_B^A = \hat{z}_B^W - \hat{z}_A^W$, $\hat{\alpha}_B^A = \hat{\alpha}_B^W - \hat{\alpha}_A^W$, and $\hat{\beta}_B^A = \hat{\beta}_B^W - \hat{\beta}_A^W$. Thus, given $\hat{\boldsymbol{\kappa}}_A$ and $\hat{\boldsymbol{\kappa}}_B$, we have the following constrained optimization:

$$\begin{aligned} \min_{\mathbf{T}_B^A \in SE(3)} \quad & \sum_{i=1}^{N_A} \sum_{j=1}^{N_B} \ell\left(e_{B_j}^{A_i}(\mathbf{T}_B^A)\right) \\ \text{s.t.} \quad & z_B^A = \hat{z}_B^A = \hat{z}_B^W - \hat{z}_A^W \\ & \alpha_B^A = \hat{\alpha}_B^A = \hat{\alpha}_B^W - \hat{\alpha}_A^W \\ & \beta_B^A = \hat{\beta}_B^A = \hat{\beta}_B^W - \hat{\beta}_A^W \end{aligned} \tag{5.7}$$

which can then be simplified to an equivalent 3-DoF unconstrained optimization problem:

$$\min_{\substack{x_B^A, y_B^A \in \mathbb{R} \\ \gamma_B^A \in [-180°, 180°]}} \sum_{i=1}^{N_A} \sum_{j=1}^{N_B} \ell \left(e_{B_j}^{A_i} \left(\mathbf{T}(\underbrace{x_B^A, y_B^A}_{\text{free}}, \underbrace{\hat{z}_B^A, \hat{\alpha}_B^A, \hat{\beta}_B^A}_{\text{constrained}}, \underbrace{\gamma_B^A}_{\text{free}}) \right) \right) \tag{5.8}$$

which can be rewritten as f to clearly see the dependencies:

$$\min_{\substack{x_B^A, y_B^A \in \mathbb{R} \\ \gamma_B^A \in [-180°, 180°]}} f(x_B^A, y_B^A, \gamma_B^A \mid \tilde{\mathbf{d}}_B^A, \mathcal{I}_A, \mathcal{I}_B, \ell) \tag{5.9}$$

That is, given our current ranging measurements $\tilde{\mathbf{d}}_B^A$, our known information tuples $\mathcal{I}_A$ and $\mathcal{I}_B$, and some loss function ℓ, our 3D pose can be found as aoutliers s 3-DoF minimization of our free variables $\langle x_B^A, y_B^A, \gamma_B^A \rangle$.

Using Equation 5.9, Algorithm 1 outlines a procedure independently followed by each agent A in a swarm $\mathcal{R}$ to individually estimate $\mathbf{T}_B^A$ between itself and all other agents $B \in (\mathcal{R} \setminus \{A\})$. Observe that *one-off* communication (i.e., `transmit`) only occurs on Line 4, when an agent *locally* detects it has violated its *a priori* constraints – this is then used by other agents on Line 7 (i.e., `invalid`) to skip calculating $\mathbf{T}_B^A$ for agents with violated constraints.

5.4 Experimental Results

We present two sets of experiments: UAV (Section 5.4.1) and UGV (Section 5.4.2). We first verify the 3D noise properties, flight hardware, and show altitude/roll/pitch constraints can be sufficiently *locally* monitored on a UAV. Once this is verified, we confidently attach surrogate UAVs at different altitudes to UGVs – this allows us to collect data more efficiently to evaluate our approach.

5.4.1 UAV Experiments

Hardware: Two agents, A and B, were used. A, a surrogate quadcopter, was a stationary Turtlebot2 UGV with 4 UWB sensors evenly distributed in the xy-plane 0.37m from the center (same robot used in Chapter 4). As pictured in Figure 1-1,

Algorithm 1 Procedure for robot $A \in \mathcal{R}$, a member of robot swarm $\mathcal{R}$, to generate instantaneous relative poses w.r.t. all other agents in swarm $B \in (\mathcal{R} \setminus \{A\})$

Input: $\mathcal{R}$, $A \in \mathcal{R}$, $\{\mathcal{I}_R \; \forall \; R \in \mathcal{R}\}$, ℓ
Output: $\{ \; \mathbf{T}_B^A \; \forall \; B \in (\mathcal{R} \setminus \{A\})$ s.t. $\neg\texttt{invalid}(B) \; \}$

function LocalProcedure
  $\tilde{\boldsymbol{\kappa}}_A \leftarrow$ *local* measurements of A w.r.t. W
  if $\neg(|\hat{\boldsymbol{\kappa}}_A - \tilde{\boldsymbol{\kappa}}_A| \leq \check{\boldsymbol{\kappa}}_A)$ **then**
    $\texttt{transmit}(A)$
  end if
  $\mathcal{T} \leftarrow \{\}$
  for $B \in (\mathcal{R} \setminus \{A\})$ s.t. $\neg\texttt{invalid}(B)$ **do**
    $\tilde{\mathbf{d}}_B^A \leftarrow$ *local* measurements of B w.r.t. A
    $\langle \hat{z}_B^A, \hat{\alpha}_B^A, \hat{\beta}_B^A \rangle \leftarrow \hat{\boldsymbol{\kappa}}_B^W - \hat{\boldsymbol{\kappa}}_A^W$
    $\langle x_B^A, y_B^A, \gamma_B^A \rangle \leftarrow$ optimize Eq. 5.9
    $\mathbf{T}_B^A \leftarrow \mathbf{T}(x_B^A, y_B^A, \hat{z}_B^A, \hat{\alpha}_B^A, \hat{\beta}_B^A, \gamma_B^A)$
    $\mathcal{T} \leftarrow \mathcal{T} \cup \{\mathbf{T}_B^A\}$
  end for
  return $\mathcal{T}$
end function

B is a custom hexrotor equipped with 6 UWB sensors evenly distributed in the xy-plane 0.31m from the center. Additionally, B is equipped with a downward facing TeraRanger Evo 15m LiDAR for altitude measurements [124].

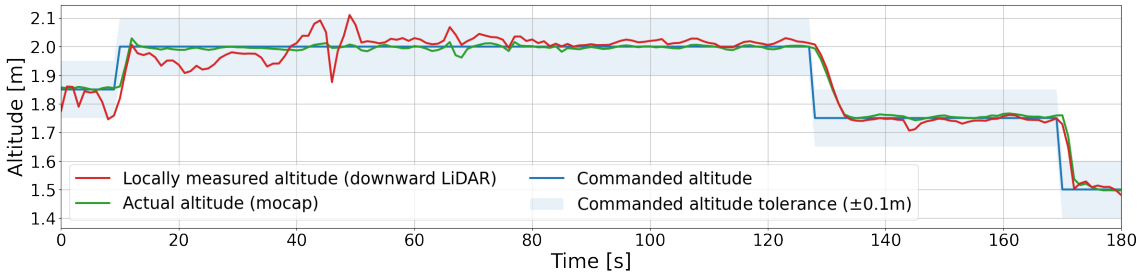
Trials: Five experimental trials were conducted in a large 10m×10m mocap space. For all trials, A was placed on a table for additional height and kept stationary, while B flew around. In the first three trials, B took off and flew simple line patterns at different altitudes (see Table 5.1 and Figure 5-2a). For the latter two trials, B flew tight circles in the center of the room – first *1m above* and then *level* with A (see Figure 5-2b). Each trial had a max speed of 1m/s and lasted approximately 3-5min (i.e., the length of a battery charge).

Discussion of Results: Results are shown in Table 5.1 and Figure 5-2. The main takeaways are as follows. (1) Verified UWB measurements were not corrupted by EMI of nearby propeller motors. (2) Confirmed the robot's body and other antennas cause significant NLOS measurement noise and bias when eclipsing a measurement path. In our Chapter 4, obstruction was only a function of relative azimuth and yaw and was addressed by a custom weighting function. In 3D, the addition of $\langle z, \alpha, \beta \rangle$ makes

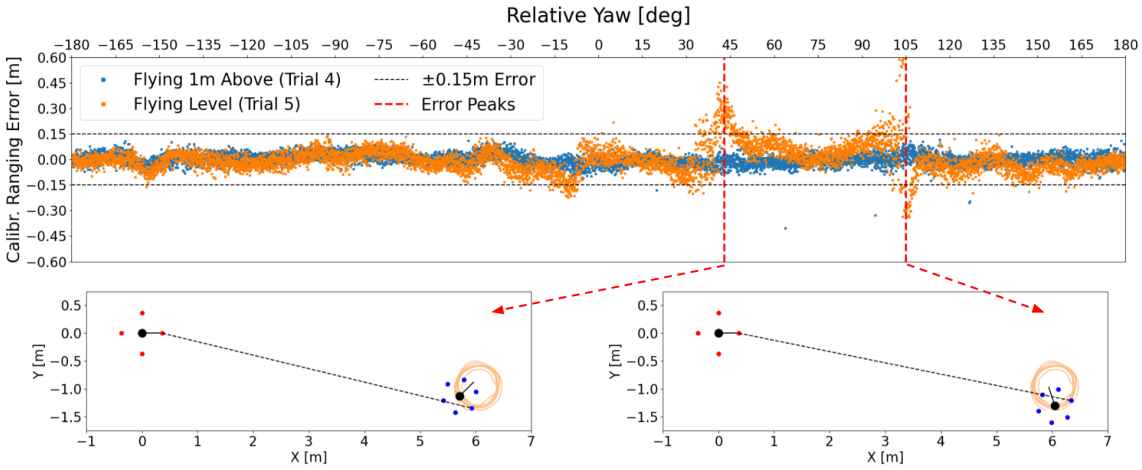
this relationship overall more complex – when agents are not in the same z-plane and remain level, these NLOS effects go away (Figure 5-2b). (3) Confirmed the onboard downward facing LiDAR and IMU can *locally* monitor altitude/roll/pitch accurately with only small variation from the mocap ground truth. Figure 5-2a shows data from Table 5.1's Trial 2, demonstrating the TeraRanger Evo provides a reasonable *local* measurement of altitude. (4) Confirmed varying within our *a priori* tolerances are not a major source of APE error, supported by the small APE differences between the latter three rows of Table 5.1. Specifically, this table shows the results of the proposed algorithm when z is free (`z_free`), or fixed to the commanded (`z_comm`), LiDAR measured (`z_meas`), or true (`z_true`) value. Together, takeaways (3) and (4) show that our minimal communication model introduces only a small amount of additional error. Furthermore, all the takeaways together indicate our extensive UGV tests are sufficient to fairly evaluate our approach in Section 5.4.2.

Table 5.1: Evaluation of positional error in meters with different z-constraints. In practice, the second row (i.e., *a priori* constraints) will be used, but the small difference between `z_comm`, `z_meas`, and `z_true` shows these constraints are not a major source of error. See Section 5.4.1.

Abs Position Error [m]		Scenarios (Section 5.4.1)								
		Trial 1			Trial 2			Trial 3		
color	**alt constraint**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**
	`z_free`	1.17	2.83	1.06	1.04	1.53	0.24	0.45	0.94	0.19
	`z_comm`	0.26	0.45	0.09	0.28	0.87	0.14	0.25	0.64	0.13
	`z_meas`	0.26	0.44	0.09	0.27	0.87	0.14	0.24	0.66	0.14
	`z_true`	0.25	0.44	0.10	0.26	0.87	0.15	0.22	0.64	0.14



(a) Shows Table 5.1's Trial 2 alongside the commanded, measured, and ground truth altitude, where the colors correspond with Table 5.1. While the TeraRanger Evo does not perfectly track ground truth, it stays within the expected commanded tolerance providing a reasonable *local* measurement of altitude.



(b) Top plot shows calibrated ranging error (i.e., mean bias subtracted) w.r.t. relative yaw for the *1m above* and *level* flights (i.e., Trial 4 and 5 respectively). Observe the error of *1m above* flight remains consistent, while the *level* flight has anomalous error peaks corresponding to specific relative yaws. The lower left and right diagrams show the relative pose of the agents at the two most prominent peaks, where the path between antennas A_1 and B_5 is eclipsed by antennas B_3 and B_1 respectively. While this NLOS effect is consistent with Figure 3-3, we note the more complex 3D relationship resolves the occlusion with varying elevation.

Figure 5-2: Plots from the UAV flight experiments.

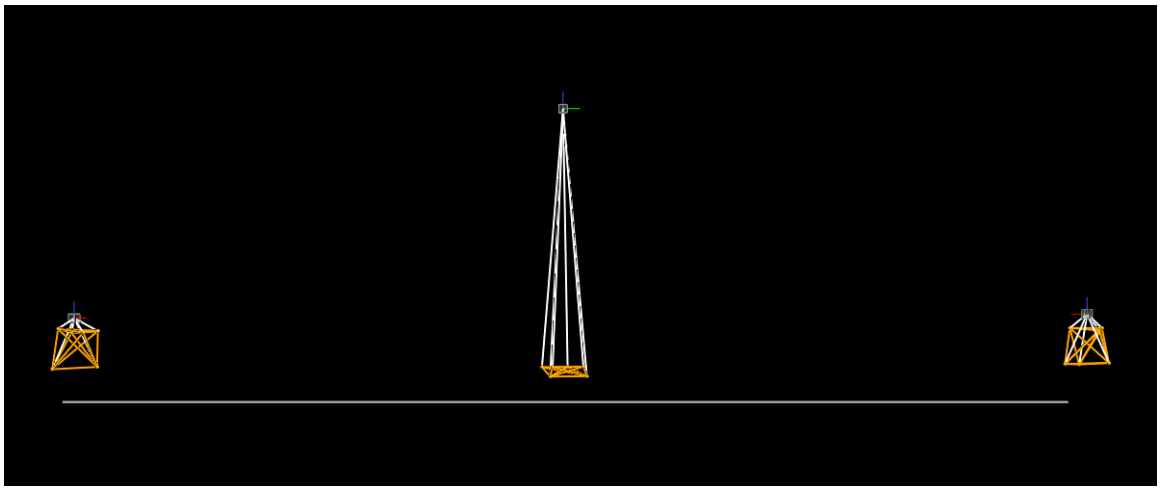


Figure 5-3: Image of mocap system tracking UGVs.

5.4.2 UGV Experiments

Hardware: Three agents, A, B and C, were used (see Figure 5-1a). All are Turtlebot2 UGV with 6 UWB sensors evenly distributed in the xy-plane 0.32m from the center. Agent A, B, and C's sensors are mounted 1.75m, 0.5m, and 0.5m above the ground respectively (i.e., $z_B^A = -1.25$m).

Trials: A full 22 datasets were collected with all three agents, each with 6 antennas, moving in various ways within the mocap space. Ranging measurements are performed by each pair of antennas between agents at 25Hz. Together the datasets total to nearly 6h (effectively creating over 200h of pairwise measurements). This dataset was used for Figures 3-4a and 3-4b. Tables 5.2 and 5.3 show results for the 5 trials where all three agents are continuously moving in various arbitrary patterns with max positional and angular velocities of 1m/s and 1rad/s respectively.

Parameters: Our approach, outlined in Section 5.3 and culminating in Equation 5.9 and Algorithm 1, is configured as such: (1) Our mean bias correction function, $\bar{d}_{B_j}^{A_i}(\mathbf{T}_B^A)$, is set to the learned 6-degree polynomial shown in Figure 3-4b. This incorporates the observed mean bias with respect to elevation into our approach. (2) The current range measurements, $\tilde{d}_{B_j}^{A_i}$, and pose estimates are smoothed by a moving average filter with a 1s and 4s window length respectively. (3) Our loss function, ℓ, is

selected to be the Huber loss $\rho_\delta(a)$ with $\delta = 0.06$, where:

$$\rho_\delta(a) = \begin{cases} \frac{1}{2}a^2 & \text{for } |a| \leq \delta \\ \delta \cdot (|a| - \frac{1}{2}\delta) & \text{otherwise} \end{cases}$$

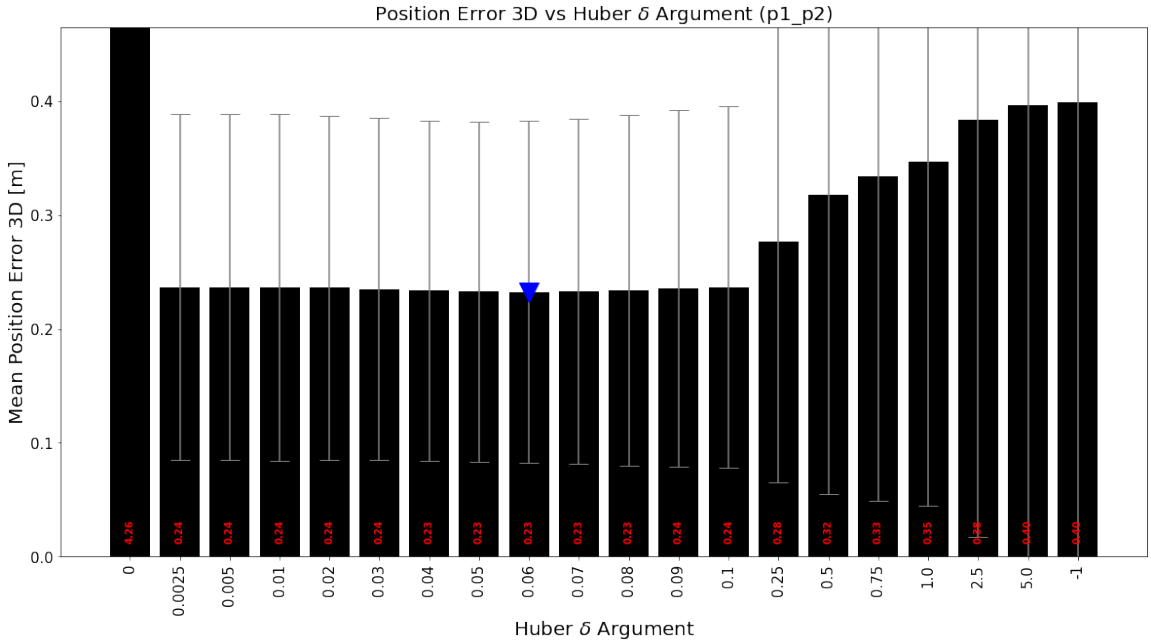
As with [69,70,115], Huber loss was selected due to its reduced outlierggles sensitivity compared to a traditional squared error loss [125]. The parameter $\delta = 0.06$ was selected after doing a parameter search over our results (see Figure 5-4).

Discussion of Results: Results are shown in Tables 5.2 and 5.3 and Figure 5-5. To demonstrate the value of individual algorithmic decisions, the tables toggle `el_bias`, `z_fixed`, `Huber`, where the final row (red) represents the proposed approach. Specifically, a check indicates:

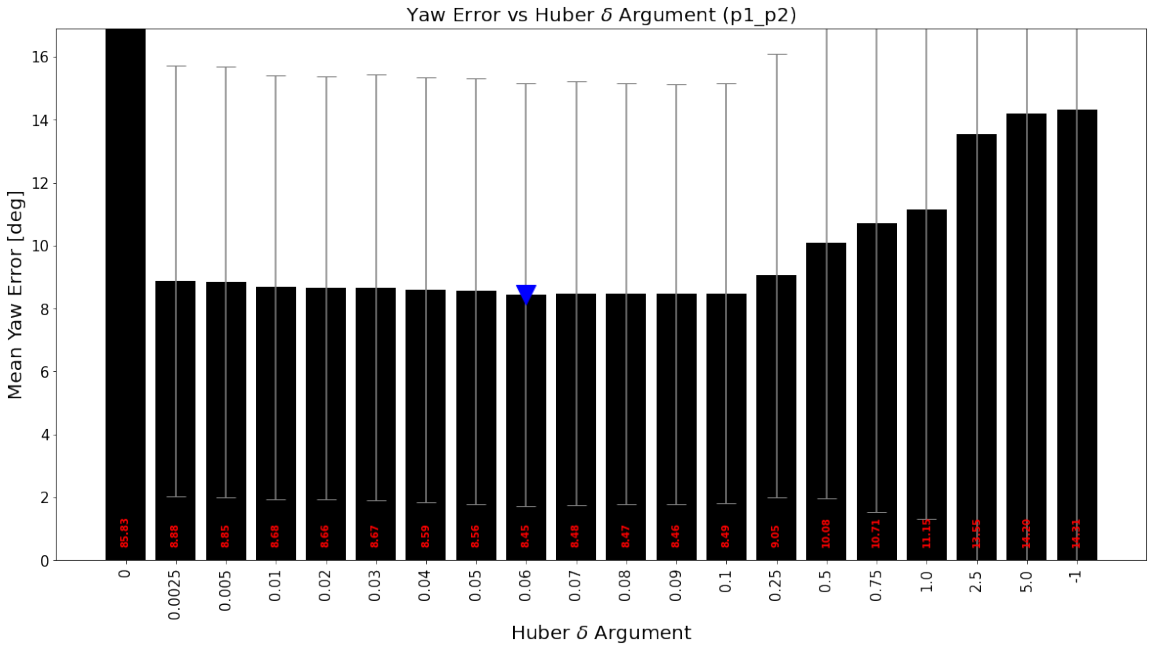
- `el_bias`: $\bar{d}^{A_i}_{B_j}(\mathbf{T}^A_B)$ is the learned mean bias correction (see Figure 3-4b), otherwise 0.
- `z_fixed`: z is constrained to z^A_B, otherwise free.
- `Huber`: $\ell(a) \triangleq \rho_{0.06}(a)$, otherwise squared error loss.

Our data shows the proposed approach (red) provides a 9x improvement over a direct NLLS trilateration (purple), similar to that used by [65]. The addition of the `el_bias` improves our approach's mean APE by an average of 19% (i.e., red vs orange). Overall, constraining z leads to the largest APE gains. Unlike APE, there is not a clear best approach for AHE, but the proposed (red) is only at most 1° behind the best for any given trial.

When their agents have similar antenna baselines to our work, they achieve a mean xy-positional error of approximately 0.40m, 0.65m, and 0.85m for simulated 8m×8m flights at various fixed altitudes. Furthermore, for real flights the mean xy-positional error becomes approximately 1m with a variable altitude and a (beneficial) larger baseline. By comparison, although our experimental environment differs, we achieve a mean APE and AHE of 0.24m and 9.5° respectively with real experiments operating in a 10m×10m mocap space. Ref. [65] differs from our approach in two important



(a)



(b)

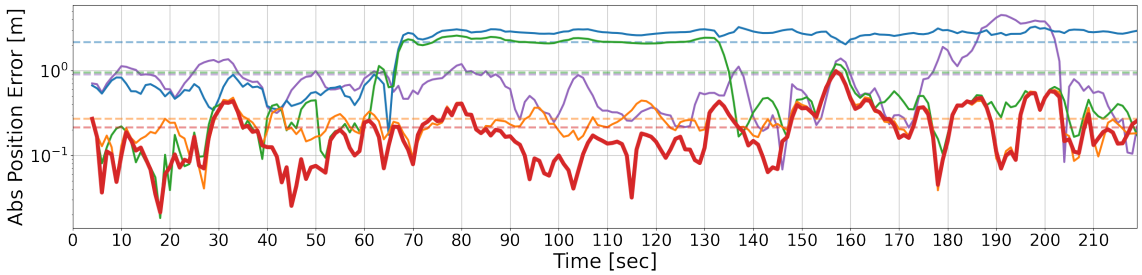
Figure 5-4: Plots from Huber loss parameter search.

Table 5.2: Evaluation of positional error in meters between algorithms. The final row represents the proposed approach. See Section 5.4.2.

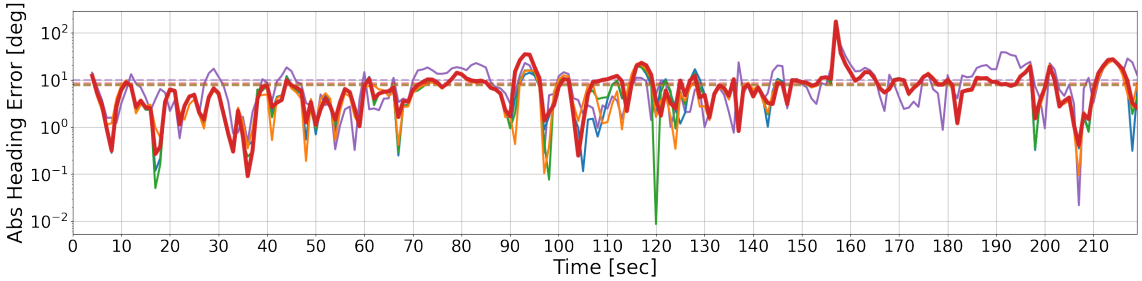
Abs Position Error [m]				Scenarios (Section 5.4.2)												
				Trial 1			Trial 2			Trial 3			Trial 4			Trial 5		
color	el_bias	z_fixed	Huber	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**
				2.21	4.40	1.17	0.89	4.50	0.88	2.37	4.82	1.13	2.26	4.58	1.12	2.77	4.32	0.59
	✓			2.55	4.95	0.64	0.89	4.46	0.97	1.32	3.94	0.95	1.33	3.97	0.97	1.55	3.71	0.90
		✓		0.45	2.87	0.38	0.42	1.50	0.28	0.57	2.97	0.52	0.45	2.40	0.36	0.40	1.16	0.23
	✓	✓		0.34	2.86	0.43	0.34	1.49	0.32	0.53	2.96	0.55	0.38	2.41	0.39	0.30	1.16	0.28
			✓	1.14	3.29	1.12	2.15	3.29	1.03	1.24	3.60	1.01	2.30	3.57	1.02	1.30	3.24	1.16
	✓		✓	0.44	1.64	0.30	0.95	2.58	0.90	0.85	3.10	0.75	0.85	2.69	0.82	1.44	2.88	0.90
		✓	✓	0.29	0.91	**0.14**	0.27	1.02	**0.14**	0.32	1.03	**0.20**	0.28	0.78	**0.14**	0.29	0.71	**0.13**
	✓	✓	✓	**0.22**	**0.90**	0.17	**0.21**	**0.98**	0.16	**0.30**	**0.97**	0.21	**0.23**	**0.75**	0.15	**0.22**	**0.63**	0.14

Table 5.3: Evaluation of yaw error in meters between algorithms. The final row represents the proposed approach. See Section 5.4.2.

Abs Heading Error [deg]				Scenarios (Section 5.4.2)												
				Trial 1			Trial 2			Trial 3			Trial 4			Trial 5		
color	el_bias	z_fixed	Huber	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**	**Mean**	**Max**	**Std**
				13.4	133.0	18.8	10.0	139.9	12.2	15.0	146.0	19.3	15.5	129.6	19.2	13.1	162.2	14.0
	✓			13.4	134.8	18.8	10.0	139.9	**12.1**	15.1	146.0	19.5	15.6	132.9	19.4	13.3	162.1	14.0
		✓		13.5	132.8	18.8	10.1	139.9	**12.1**	15.0	146.5	19.3	15.5	131.9	19.3	13.2	152.4	13.5
	✓	✓		13.8	133.0	18.7	10.4	**139.0**	12.2	15.0	146.2	19.3	16.8	170.1	23.1	14.0	166.7	14.4
			✓	8.0	47.5	5.5	7.7	177.7	12.8	8.5	57.5	**6.1**	**10.1**	111.4	**10.7**	**10.4**	47.5	**7.6**
	✓		✓	7.8	47.7	5.7	7.8	177.3	12.8	8.5	**57.3**	**6.1**	10.3	**111.2**	11.0	**10.4**	46.5	**7.6**
		✓	✓	**7.6**	**42.4**	**5.1**	**7.6**	172.6	12.5	**8.3**	58.4	**6.1**	10.4	122.0	11.2	10.6	48.5	**7.6**
	✓	✓	✓	8.6	46.7	5.8	8.4	172.8	12.8	8.7	59.8	6.5	10.7	111.8	11.2	11.0	**42.3**	7.7



(a) Proposed approach (red) generally outperforms other approaches with respect to absolute position error – only a few select points being the exception.



(b) Proposed approach (red) generally equivalent to other approaches with respect to absolute heading error, but is outperformed at a few select points.

Figure 5-5: Error vs. Time plots for Trial 2 in Table 5.2 and 5.3 respectively. The line color corresponds with the algorithm's color column in the associated tables. Each colored dashed line represent that algorithm's overall mean in that trial.

ways: (1) the authors model UWB ranging error as a zero mean Gaussian with a 0.10m standard deviation, which does not reflect real data (see Section 3.4.1); and (2) we define an explicit communication protocol that allows us to address several capability gaps without *continuously* transmitting measurements. Specifically, by constraining altitude/roll/pitch and improving our UWB noise model, we can better address UWB noise (see Section 3.4.1) and an observability ambiguity (see Section 1.2.3 and Figure 5-1c). See Section 5.1 for more details.

Chapter 6

Conclusion

Motivated by the applications of nuclear non-proliferation, radiological search, and radiological mapping, this thesis explores leveraging multiple ultra-wideband (UWB) ranging sensors to produce frequent inter-agent pose estimates with minimal communication overhead. This work is intended as a component of a larger multi-agent simultaneous localization and mapping (SLAM) system (also known as collaborative SLAM or CSLAM), where persistent UWB-based inter-agent pose estimates provide a valuable alternative source of inter-agent loop closures. When compared to other state-of-the-art approaches, our work demonstrates improved performance over similar systems, while remaining competitive with methods that have significantly higher communication costs.

In the future, we hope to extend this work in the following ways:

- As planned, integrate the UWB relative pose system into a full SLAM stack. Use relative pose as a heuristic for performing inter-agent visual loop closures.
- Starting from [126–128], use a neural network to learn the sensor model – mean bias correction, NLOS conditions, etc.
- Extend CORA [104] to use multiple antennas per agent.
- Continue flight tests at LBNL (see Figure 6-2) – fly multiple large drones fast!

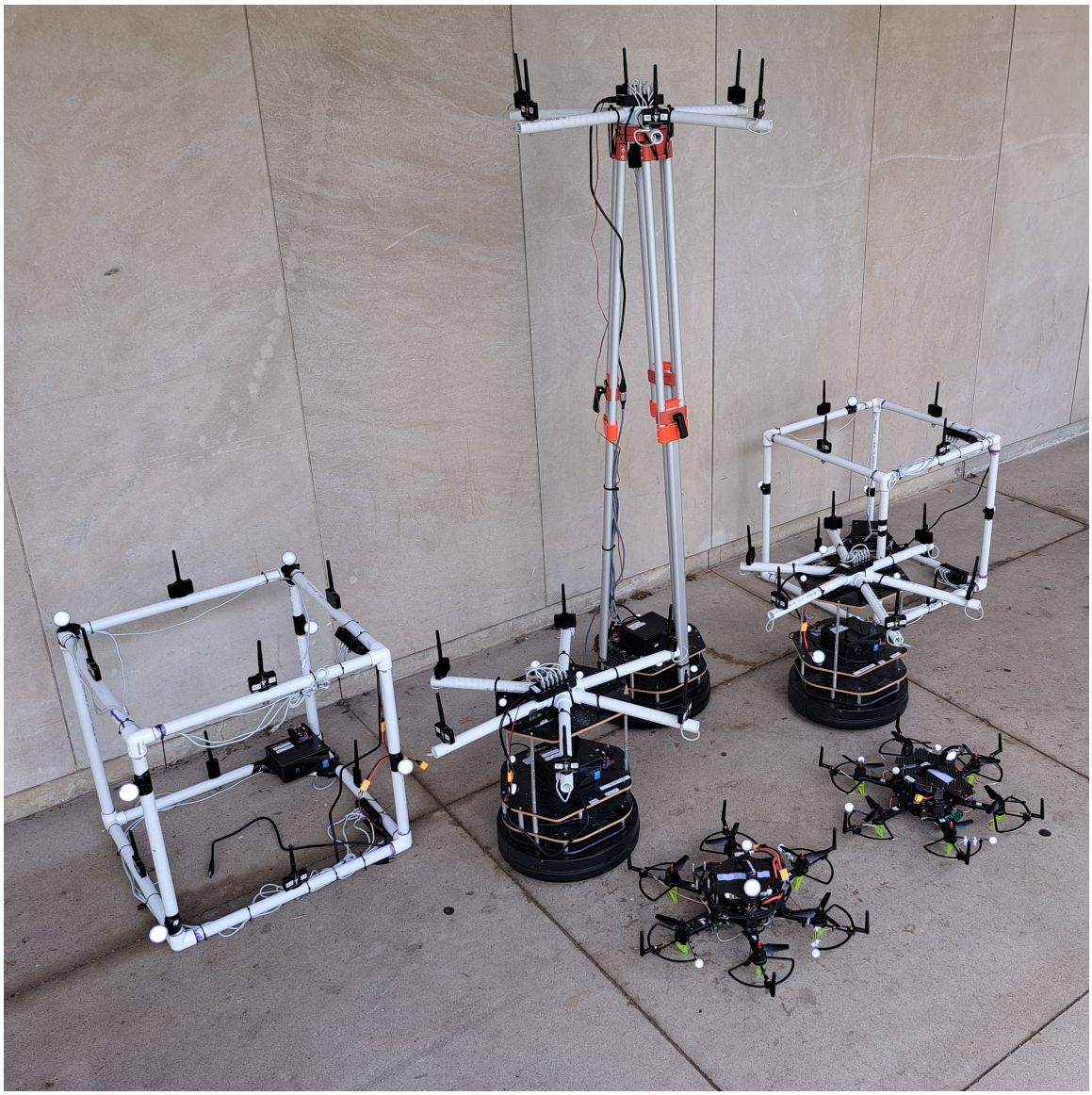


Figure 6-1: The UWB fleet used for past and future testing.

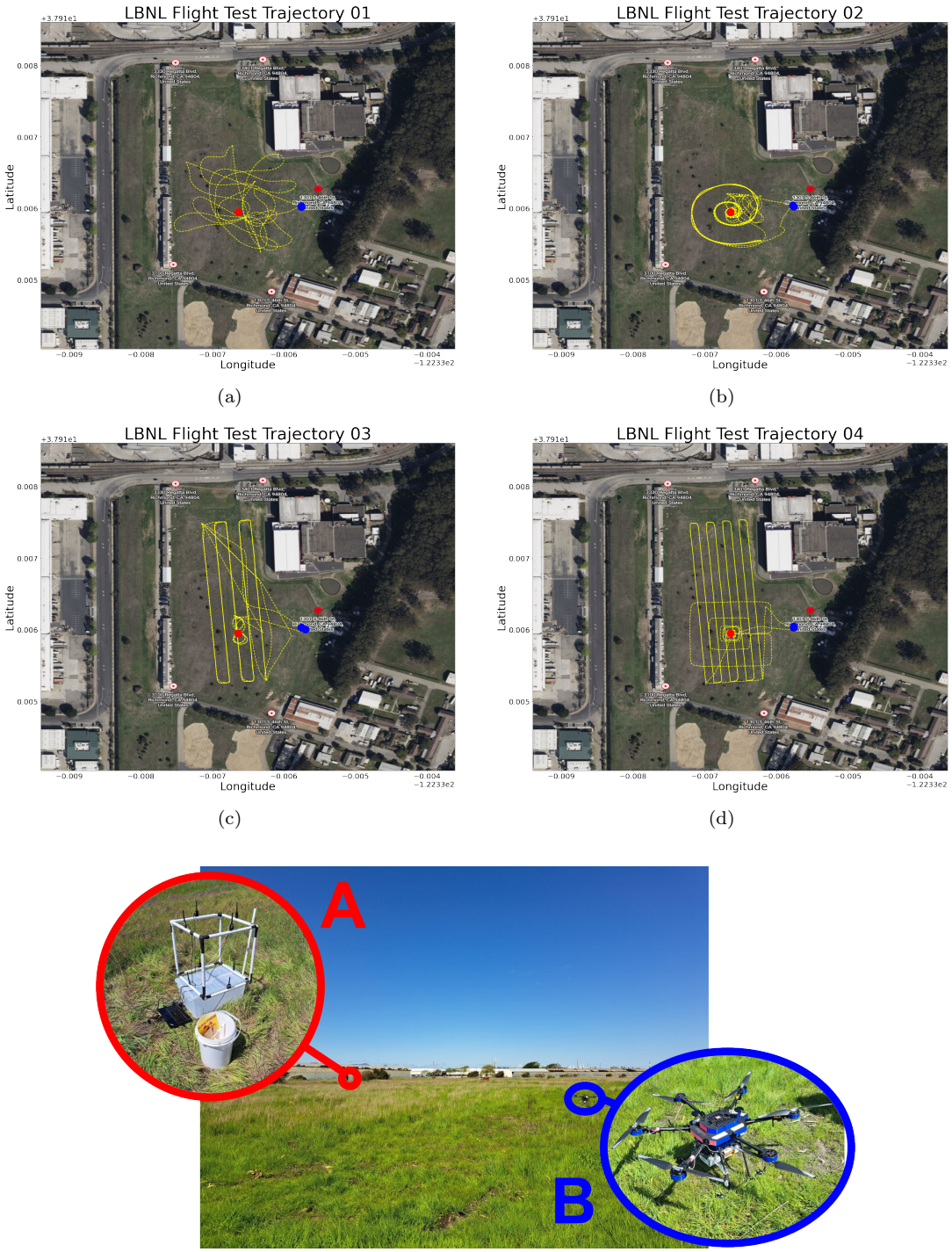


Figure 6-2: Images from initial LBNL flight tests. (a)-(d) show the test paths flown. The area flown in trajectory 04 is approximately 250m by 100m. (e) Shows the UAV and surrogate UGV.

Bibliography

[1] Yulun Tian, Katherine Liu, Kyel Ok, Loc Tran, Danette Allen, Nicholas Roy, and Jonathan P. How. Search and rescue under the forest canopy using multiple uavs. Int. J. Robotics Res., 39(10-11), 2020.

[2] Xiulong Liu, Jiannong Cao, Yanni Yang, and Shan Jiang. Cps-based smart warehouse for industry 4.0: A survey of the underlying technologies. Comput., 7(1):13, 2018.

[3] Ali Agha, Kyohei Otsu, Benjamin Morrell, David D. Fan, Rohan Thakker, and et al. Nebula: Quest for robotic autonomy in challenging environments; TEAM costar at the DARPA subterranean challenge. CoRR, abs/2103.11470, 2021.

[4] David Filliat and Jean-Arcady Meyer. Map-based navigation in mobile robots:: I. a review of localization strategies. Cognitive systems research, 4(4):243–282, 2003.

[5] Muhammad Saim, Khalid Munawar, and Ubaid Al-Saggaf. An overview of localization methods for multi-agent systems. Int J Eng Res Appl, 7(01):19–24, 2017.

[6] Xun S. Zhou and Stergios I. Roumeliotis. Robot-to-robot relative pose estimation from range measurements. IEEE Trans. Robotics, 24(6):1379–1393, 2008.

[7] Yulun Tian, Yun Chang, Fernando Herrera Arias, Carlos Nieto-Granda, Jonathan P. How, and Luca Carlone. Kimera-Multi: Robust, Distributed, Dense Metric-Semantic SLAM for Multi-Robot Systems. IEEE Transactions on Robotics, 38(4):2022–2038, August 2022.

[8] Yun Chang, Yulun Tian, Jonathan P. How, and Luca Carlone. Kimera-multi: a system for distributed multi-robot metric-semantic simultaneous localization and mapping. In IEEE International Conference on Robotics and Automation, ICRA 2021, Xi'an, China, May 30 - June 5, 2021, pages 11210–11218. IEEE, 2021.

[9] Pierre-Yves Lajoie, Benjamin Ramtoula, Yun Chang, Luca Carlone, and Giovanni Beltrame. DOOR-SLAM: distributed, online, and outlier resilient SLAM for robotic teams. IEEE Robotics Autom. Lett., 5(2):1656–1663, 2020.

[10] Titus Cieslewski, Siddharth Choudhary, and Davide Scaramuzza. Data-efficient decentralized visual SLAM. In 2018 IEEE International Conference on Robotics and Automation, ICRA 2018, Brisbane, Australia, May 21-25, 2018, pages 2466–2473. IEEE, 2018.

[11] Glenn F Knoll. Radiation detection and measurement. John Wiley & Sons, 2010.

[12] Wikipedia contributors. Western Australian radioactive capsule incident — Wikipedia, the free encyclopedia, 2023. [Online; accessed 10-January-2024].

[13] Wikipedia contributors. Fukushima nuclear accident — Wikipedia, the free encyclopedia, 2024. [Online; accessed 10-January-2024].

[14] Radiation Monitoring Data from Fukushima Area 04/18/2011, April 2011.

[15] Fukushima Daiichi Accident - World Nuclear Association. `https://www.world-nuclear.org/information-library/safety-and-security/safety-of-plants/fukushima-daiichi-accident.aspx` [Online; accessed 10-January-2024].

[16] Alexis L Reed. Us doe's response to the fukushima daiichi reactor accident: Answers and data products for decision makers. Health Physics, 102(5):557–562, 2012.

[17] Ying Huang, Xiaoyu Shi, Yan Zhou, and Zhenhua Xiong. Autonomous navigation of mobile robot in radiation environment with uneven terrain. International Journal of Intelligent Robotics and Applications, 7(3):497–509, 2023.

[18] Andrew West, Ioannis Tsitsimpelis, Mauro Licata, Anze Jazbec, Luka Snoj, Malcolm J Joyce, and Barry Lennox. Use of gaussian process regression for radiation mapping of a nuclear reactor with a mobile robot. Scientific reports, 11(1):13975, 2021.

[19] Erin A Miller, Sean M Robinson, Kevin K Anderson, Jonathon D McCall, Amanda M Prinke, Jennifer B Webster, and Carolyn E Seifert. Adaptively reevaluated bayesian localization (arbl): A novel technique for radiological source localization. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 784:332–338, 2015.

[20] Branko Ristic, Mark Morelande, and Ajith Gunatilaka. Information driven search for point sources of gamma radiation. Signal Processing, 90(4):1225–1239, 2010.

[21] R Pavlovsky, JW Cates, WJ Vanderlip, THY Joshi, A Haefner, E Suzuki, R Barnowski, V Negut, A Moran, K Vetter, et al. 3d gamma-ray and neutron mapping in real-time with the localization and mapping platform from

unmanned aerial systems and man-portable configurations. arXiv preprint arXiv:1908.06114, 2019.

[22] Kai Vetter, Ross Barnowski, Joshua W Cates, Andrew Haefner, Tenzing HY Joshi, Ryan Pavlovsky, and Brian J Quiter. Advances in nuclear radiation sensing: Enabling 3-d gamma-ray vision. Sensors, 19(11):2541, 2019.

[23] Andrew J Torgesen. Autonomous Sensing and Navigation in Challenging Environments Using Unmanned Air Vehicles in Single-and Multi-Agent Settings. PhD thesis, Massachusetts Institute of Technology, 2021.

[24] Samuel Kemp and Jonathan Rogers. Uav-ugv teaming for rapid radiological mapping. In 2021 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), pages 92–97. IEEE, 2021.

[25] David Woller and Miroslav Kulich. Path planning algorithm ensuring accurate localization of radiation sources. Applied Intelligence, 52(8):9574–9596, 2022.

[26] Frank Mascarich, Mihir Kulkarni, Paolo De Petris, Taylor Wilson, and Kostas Alexis. Autonomous mapping and spectroscopic analysis of distributed radiation fields using aerial robots. Autonomous Robots, 47(2):139–160, 2023.

[27] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian Reid, and John J Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. IEEE Transactions on robotics, 32(6):1309–1332, 2016.

[28] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. IEEE transactions on robotics, 31(5):1147–1163, 2015.

[29] Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, and Juan D Tardós. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. IEEE Transactions on Robotics, 37(6):1874–1890, 2021.

[30] Antoni Rosinol, Andrew Violette, Marcus Abate, Nathan Hughes, Yun Chang, Jingnan Shi, Arjun Gupta, and Luca Carlone. Kimera: From slam to spatial perception with 3d dynamic scene graphs. The International Journal of Robotics Research, 40(12-14):1510–1546, 2021.

[31] Antoni Rosinol, Marcus Abate, Yun Chang, and Luca Carlone. Kimera: an open-source library for real-time metric-semantic localization and mapping. In 2020 IEEE International Conference on Robotics and Automation (ICRA), pages 1689–1696. IEEE, 2020.

[32] Sebastian Thrun. Probabilistic robotics. Communications of the ACM, 45(3):52–57, 2002.

[33] Timothy D Barfoot. State estimation for robotics. Cambridge University Press, 2017.

[34] Xiang Gao and Tao Zhang. Introduction to visual SLAM: from theory to practice. Springer Nature, 2021.

[35] Luca Carlone, Kasra Khosoussi, Vasileios Tzoumas, Golnaz Habibi, Markus Ryll, Rajat Talak, Jingnan Shi, and Pasquale Antonante. Visual navigation for autonomous vehicles: An open-source hands-on robotics course at mit. In 2022 IEEE Integrated STEM Education Conference (ISEC), pages 177–184. IEEE, 2022.

[36] Frank Dellaert, Michael Kaess, et al. Factor graphs for robot perception. Foundations and Trends® in Robotics, 6(1-2):1–139, 2017.

[37] Frank Dellaert. Factor graphs and gtsam: A hands-on introduction. Georgia Institute of Technology, Tech. Rep, 2:4, 2012.

[38] Frank Dellaert and GTSAM Contributors. borglab/gtsam, May 2022.

[39] Pierre-Yves Lajoie, Benjamin Ramtoula, Yun Chang, Luca Carlone, and Giovanni Beltrame. Door-slam: Distributed, online, and outlier resilient slam for robotic teams. IEEE Robotics and Automation Letters, 5(2):1656–1663, 2020.

[40] Yulun Tian, Kasra Khosoussi, and Jonathan P. How. A resource-aware approach to collaborative loop-closure detection with provable performance guarantees. Int. J. Robotics Res., 40(10-11), 2021.

[41] Luc Jaulin. Mobile robotics. John Wiley & Sons, 2019.

[42] Richard B Langley et al. Dilution of precision. GPS world, 10(5):52–59, 1999.

[43] Y Baudoin and Maki K Habib. Using robots in hazardous environments: Landmine detection, de-mining and other applications. Elsevier, 2010.

[44] Reza Zekavat and R Michael Buehrer. Handbook of position location: Theory, practice and advances, volume 27. John Wiley & Sons, 2011.

[45] Xinya Li, Zhiqun Daniel Deng, Lynn T Rauchenstein, and Thomas J Carlson. Contributed review: Source-localization algorithms and applications using time of arrival and time difference of arrival measurements. Review of Scientific Instruments, 87(4):041502, 2016.

[46] Erin Marie Fischell, Nicholas Rahardiyan Rypkema, and Henrik R. Schmidt. Relative autonomy and navigation for command and control of low-cost autonomous underwater vehicles. IEEE Robotics Autom. Lett., 4(2):1800–1806, 2019.

[47] Kang Eun Jeon, James She, Perm Soonsawad, and Pai Chet Ng. BLE beacons for internet of things applications: Survey, challenges, and opportunities. IEEE Internet Things J., 5(2):811–828, 2018.

[48] Taweesak Sanpechuda and La-or Kovavisaruch. A review of rfid localization: Applications and techniques. In 2008 5th international conference on electrical engineering/electronics, computer, telecommunications and information technology, volume 2, pages 769–772. IEEE, 2008.

[49] Fernando Herranz, Angel Llamazares, Eduardo J. Molinos, and Manuel Ocaña. A comparison of SLAM algorithms with range only sensors. In IEEE International Conference on Robotics and Automation, ICRA, pages 4606–4611. IEEE, 2014.

[50] Aditya Arun, Roshan Ayyalasomayajula, William Hunter, and Dinesh Bharadia. P2slam: Bearing based wifi slam for indoor robots. IEEE Robotics and Automation Letters, 2022.

[51] Xiansheng Guo, Nirwan Ansari, Fangzi Hu, Yuan Shao, Nkrow Raphael Elikplim, and Lin Li. A survey on fusion-based indoor positioning. IEEE Communications Surveys & Tutorials, 22(1):566–594, 2019.

[52] Pratap Misra and Per Enge. Global positioning system: signals measurements. 2011.

[53] Elliott D Kaplan and Christopher Hegarty. Understanding GPS/GNSS: principles and applications. Artech house, 2017.

[54] Gilbert Strang and Kai Borre. Linear algebra, geodesy, and GPS. Siam, 1997.

[55] Harry B. Lee and Lincoln Laboratory. Accuracy limitations of range-range (spherical) multilateration systems. Technical Report DOT/TSC-RA-3-8-(1), October 1973.

[56] Harry B. Lee. Accuracy Limitations of Hyperbolic Multilateration Systems. IEEE Transactions on Aerospace and Electronic Systems, AES-11(1):16–29, January 1975. Conference Name: IEEE Transactions on Aerospace and Electronic Systems.

[57] PR Escobal, HF Fliegel, RM Jaffe, PM Muller, KM Ong, and OH Vonroos. A 3-d multilateration: A precision geodetic measurement system. JPL Quart. Tech. Rev., 2(3), 1972.

[58] Eric W. Weisstein. Circle-Circle Intersection. Publisher: Wolfram Research, Inc.

[59] Eric W. Weisstein. Sphere-Sphere Intersection. Publisher: Wolfram Research, Inc.

[60] Harry B Lee et al. Accuracy limitations of range-range (spherical) multilateration systems. Technical report, Lincoln Laboratory, 1973.

[61] Berthold K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. JOSA A, 4(4):629–642, April 1987. Publisher: Optica Publishing Group.

[62] Berthold K. P. Horn, Hugh M. Hilden, and Shahriar Negahdaripour. Closed-form solution of absolute orientation using orthonormal matrices. JOSA A, 5(7):1127–1135, July 1988. Publisher: Optica Publishing Group.

[63] Registration Techniques in Robotics, May 2020.

[64] Richard B. Langley. Dilution of Precision, May 1999.

[65] Yu Xianjia, Li Qingqing, Jorge Pena Queralta, Jukka Heikkonen, and Tomi Westerlund. Cooperative UWB-Based Localization for Outdoors Positioning and Navigation of UAVs aided by Ground Robots. In 2021 IEEE International Conference on Autonomous Systems (ICAS), pages 1–5, Montreal, QC, Canada, August 2021. IEEE.

[66] Binghao Li, Andrew G. Dempster, and Jian Wang. 3D DOPs for Positioning Applications Using Range Measurements. Wireless Sensor Network, 03(10):334, October 2011. Number: 10 Publisher: Scientific Research Publishing.

[67] Bradford W. Parkinson and James J. Spilker. The global positioning system : theory and applications. Progress in astronautics and aeronautics ; v. 163-164. American Institute of Aeronautics and Astronautics, Washington, DC, 1996.

[68] Bradford W. Parkinson, James J. Spilker, Penina. Axelrad, and Per. Enge. Global Positioning System theory and applications. Volume II. Progress in astronautics and aeronautics ; v. 164. American Institute of Aeronautics and Astronautics, Inc., Washington, D.C, 1996.

[69] S. Zheng, Z. Li, Y. Liu, H. Zhang, P. Zheng, X. Liang, Y. Li, X. Bu, and X. Zou. UWB-VIO Fusion for Accurate and Robust Relative Localization of Round Robotic Teams. IEEE Robotics and Automation Letters, 7(4):11950–11957, October 2022.

[70] Haifeng Zhang, Zhitian Li, Shuaikang Zheng, Pengcheng Zheng, Xingdong Liang, YanLei Li, Xiangxi Bu, and Xudong Zou. Range-Aided Drift-Free Cooperative Localization and Consistent Reconstruction of Multi-Ground Robots. IEEE Robotics and Automation Letters, 8(4):2094–2101, April 2023. Conference Name: IEEE Robotics and Automation Letters.

[71] Characteristics of ultra-wideband technology. Technical Report Rec. ITU-R SM.1755-0, International Telecommunication Union Radiocommunication Sector (ITU-R), 2006.

[72] Yusnita Rahayu, Tharek Abd Rahman, Razali Ngah, and Peter S Hall. Ultra wideband technology and its applications. In 2008 5th IFIP International Conference on Wireless and Optical Communications Networks (WOCN'08), pages 1–5. IEEE, 2008.

[73] Elly Schietse. The State Of UWB in 2022, 2022. https://get.pozyx.io/hubfs/Brochures/State-Of-UWB-2022-Pozyx.pdf [Online; accessed 10-January-2024].

[74] Wikipedia contributors. List of uwb-enabled mobile devices — Wikipedia, the free encyclopedia, 2023. [Online; accessed 8-December-2023].

[75] Abdulrahman Alarifi, AbdulMalik S. Al-Salman, Mansour Alsaleh, Ahmad Alnafessah, Suheer Alhadhrami, Mai A. Al-Ammar, and Hend S. Al-Khalifa. Ultra wideband indoor positioning technologies: Analysis and recent advances. Sensors, 16(5):707, 2016.

[76] Noooploop. LinkTrack – Model Selection, 2023. https://www.noooploop.com/en/linktrack/.

[77] Nour Smaoui, Omprakash Gnawali, and Kyungki Kim. Study and mitigation of platform related UWB ranging errors. In 2020 International Conference on COMmunication Systems & NETworkS, COMSNETS 2020, Bengaluru, India, January 7-11, 2020, pages 346–353. IEEE, 2020.

[78] Anton Ledergerber and Raffaello D'Andrea. Ultra-wideband range measurement model with gaussian processes. In IEEE Conference on Control Technology and Applications, CCTA 2017, Mauna Lani Resort, HI, USA, August 27-30, 2017, pages 1929–1934. IEEE, 2017.

[79] Anton Ledergerber and Raffaello D'Andrea. Calibrating away inaccuracies in ultra wideband range measurements: A maximum likelihood approach. IEEE Access, 6:78719–78730, 2018.

[80] Michael Hamer and Raffaello D'Andrea. Self-calibrating ultra-wideband network supporting multi-robot localization. IEEE Access, 6:22292–22304, 2018.

[81] Andrew Fishberg and Jonathan P. How. Multi-agent relative pose estimation with UWB and constrained communications. In IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2022, Kyoto, Japan, October 23-27, 2022, pages 778–785. IEEE, 2022.

[82] Andrew Fishberg, Brian Quiter, and Jonathan P How. Murp: Multi-agent ultra-wideband relative pose estimation with constrained communications in 3d environments. arXiv preprint arXiv:2312.17731, 2023.

[83] Soon-Jo Chung, Aditya Avinash Paranjape, Philip Dames, Shaojie Shen, and Vijay Kumar. A survey on aerial swarm robotics. IEEE Transactions on Robotics, 34(4):837–855, 2018.

[84] Xianjia Yu, Qingqing Li, Jorge Peña Queralta, Jukka Heikkonen, and Tomi Westerlund. Applications of UWB networks and positioning to autonomous robots and industrial systems. In 10th Mediterranean Conference on Embedded Computing, MECO 2021, Budva, Montenegro, June 7-10, 2021, pages 1–6. IEEE, 2021.

[85] Abhishek Goudar, Wenda Zhao, and Angela P Schoellig. Range-visual-inertial sensor fusion for micro aerial vehicle localization and navigation. IEEE Robotics and Automation Letters, 9(1):683–690, 2023.

[86] Michael Strohmeier, Thomas Walter, Julian Rothe, and Sergio Montenegro. Ultra-wideband based pose estimation for small unmanned aerial vehicles. IEEE Access, 6:57526–57535, 2018.

[87] Bo Yang, Jun Li, and Hong Zhang. Uvip: Robust uwb aided visual-inertial positioning system for complex indoor environments. In 2021 IEEE International Conference on Robotics and Automation (ICRA), pages 5454–5460. IEEE, 2021.

[88] Daniele Fontanelli, Farhad Shamsfakhr, Paolo Bevilacqua, and Luigi Palopoli. UWB indoor global localisation for nonholonomic robots with unknown offset compensation. In IEEE International Conference on Robotics and Automation, ICRA 2021, Xi'an, China, May 30 - June 5, 2021, pages 5795–5801. IEEE, 2021.

[89] Thien Hoang Nguyen, Thien-Minh Nguyen, and Lihua Xie. Range-focused fusion of camera-imu-uwb for accurate and drift-reduced localization. IEEE Robotics Autom. Lett., 6(2):1678–1685, 2021.

[90] Mark W. Müller, Michael Hamer, and Raffaello D'Andrea. Fusing ultra-wideband range measurements with accelerometers and rate gyroscopes for quadrocopter state estimation. In IEEE International Conference on Robotics and Automation, ICRA 2015, Seattle, WA, USA, 26-30 May, 2015, pages 1730–1736. IEEE, 2015.

[91] Abhishek Goudar, Timothy D. Barfoot, and Angela P. Schoellig. Continuous-Time Range-Only Pose Estimation, May 2023. arXiv:2304.09043 [cs].

[92] Saman Fahandezh-Saadi and Mark W. Mueller. Optimal measurement selection algorithm and estimator for ultra-wideband symmetric ranging localization. CoRR, abs/1804.09773, 2018.

[93] Sungjae Shin, Eungchang Mason Lee, Junho Choi, and Hyun Myung. MIR-VIO: mutual information residual-based visual inertial odometry with UWB fusion for robust localization. CoRR, abs/2109.00747, 2021.

[94] Nobuhiro Funabiki, Benjamin Morrell, Jeremy Nash, and Ali-akbar Aghamohammadi. Range-aided pose-graph-based SLAM: applications of deployable ranging beacons for unknown environment exploration. IEEE Robotics Autom. Lett., 6(1):48–55, 2021.

[95] Adyasha Mohanty, Asta Wu, Sriramya Bhamidipati, and Grace Gao. Precise relative positioning via tight-coupling of gps carrier phase and multiple uwbs. IEEE Robotics and Automation Letters, 7(2):5757–5762, 2022.

[96] Weisong Wen, Xiwei Bai, Guohao Zhang, Shengdong Chen, Feng Yuan, and Li-Ta Hsu. Multi-agent collaborative gnss/camera/ins integration aided by inter-ranging for vehicular navigation in urban areas. IEEE Access, 8:124323–124338, 2020.

[97] Yang Qi, Yisheng Zhong, and Zongying Shi. Cooperative 3-D relative localization for UAV swarm by fusing UWB with IMU and GPS. Journal of Physics: Conference Series, 1642(1):012028, September 2020.

[98] Mohammed Shalaby, Charles Champagne Cossette, James Richard Forbes, and Jerome Le Ny. Relative position estimation in multi-agent systems using attitude-coupled range measurements. IEEE Robotics Autom. Lett., 6(3):4955–4961, 2021.

[99] Charles Champagne Cossette, Mohammed Shalaby, David Saussié, James Richard Forbes, and Jerome Le Ny. Relative position estimation between two UWB devices with imus. IEEE Robotics Autom. Lett., 6(3):4313–4320, 2021.

[100] Frederike Dümbgen, Mohammed A Shalaby, Connor Holmes, Charles C Cossette, James R Forbes, Jerome Le Ny, and Timothy D Barfoot. Star-loc: Dataset for stereo and range-based localization. arXiv preprint arXiv:2309.05518, 2023.

[101] Frederike Dümbgen, Connor Holmes, Ben Agro, and Timothy D Barfoot. Toward globally optimal state estimation using automatically tightened semidefinite relaxations. arXiv preprint arXiv:2308.05783, 2023.

[102] Tianyue Wu and Fei Gao. Distributed optimization in sensor network for scalable multi-robot relative state estimation. arXiv preprint arXiv:2303.01242, 2023.

[103] Abhishek Goudar, Frederike Dümbgen, Timothy D Barfoot, and Angela P Schoellig. Optimal initialization strategies for range-only trajectory estimation. arXiv preprint arXiv:2309.09011, 2023.

[104] Alan Papalia, Andrew Fishberg, Brendan W O'Neill, Jonathan P How, David M Rosen, and John J Leonard. Certifiably correct range-aided slam. arXiv preprint arXiv:2302.11614, 2023.

[105] Chiara Bonsignori, Fabio Condomitti, Marco Del Gamba, Federico Garzelli, Leonardo Lossi, Francesco Mione, Alessandro Noferi, and Alessio Vecchio. Estimation of user's orientation via wearable UWB. In 16th International Conference on Intelligent Environments, IE 2020, Madrid, Spain, July 20-23, 2020, pages 80–83. IEEE, 2020.

[106] Ehab Ghanem, Kyle O'Keefe, and Richard Klukas. Testing vehicle-to-vehicle relative position and attitude estimation using multiple UWB ranging. In 92nd IEEE Vehicular Technology Conference, VTC Fall 2020, Victoria, BC, Canada, November 18 - December 16, 2020, pages 1–5. IEEE, 2020.

[107] Ernst-Johann Theussl, Dimitar Ninevski, and Paul O'Leary. Measurement of relative position and orientation using UWB. In IEEE International Instrumentation and Measurement Technology Conference, I2MTC 2019, Auckland, New Zealand, May 20-23, 2019, pages 1–6. IEEE, 2019.

[108] Zhiqiang Cao, Ran Liu, Chau Yuen, Achala Athukorala, Benny Kai Kiat Ng, Muraleetharan Mathanraj, and U-Xuan Tan. Relative localization of mobile robots with multiple ultra-wideband ranging measurements. In IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2021, Prague, Czech Republic, September 27 - Oct. 1, 2021, pages 5857–5863. IEEE, 2021.

[109] Thien-Minh Nguyen, Abdul Hanif Bin Zaini, Chen Wang, Kexin Guo, and Lihua Xie. Robust target-relative localization with ultra-wideband ranging and communication. In 2018 IEEE International Conference on Robotics and Automation, ICRA 2018, Brisbane, Australia, May 21-25, 2018, pages 2312–2319. IEEE, 2018.

[110] Jinxin Liu and Guoqiang Hu. Relative localization estimation for multiple robots via the rotating ultra-wideband tag. IEEE Robotics and Automation Letters, 2023.

[111] Christoph Steup, Jonathan Beckhaus, and Sanaz Mostaghim. A single-copter uwb-ranging-based localization system extendable to a swarm of drones. Drones, 5(3):85, 2021.

[112] Weiying Wang, Anne Kemmeren, Daniel Son, Javier Alonso-Mora, and Stephanie Gil. Wi-closure: Reliable and efficient search of inter-robot loop closures using wireless sensing. In 2023 IEEE International Conference on Robotics and Automation (ICRA), pages 2069–2075. IEEE, 2023.

[113] Kexin Guo, Xiuxian Li, and Lihua Xie. Ultra-Wideband and Odometry-Based Cooperative Relative Localization With Application to Multi-UAV Formation Control. IEEE Transactions on Cybernetics, 50(6):2590–2603, June 2020.

[114] Hao Xu, Luqi Wang, Yichen Zhang, Kejie Qiu, and Shaojie Shen. Decentralized visual-inertial-uwb fusion for relative state estimation of aerial swarm. In 2020 IEEE International Conference on Robotics and Automation, ICRA 2020, Paris, France, May 31 - August 31, 2020, pages 8776–8782. IEEE, 2020.

[115] Hao Xu, Yichen Zhang, Boyu Zhou, Luqi Wang, Xinjie Yao, Guotao Meng, and Shaojie Shen. Omni-swarm: A decentralized omnidirectional visual-inertial-uwb state estimation system for aerial swarm. CoRR, abs/2103.04131, 2021.

[116] Zhiren Xun, Jian Huang, Zhehan Li, Zhenjun Ying, Yingjian Wang, Chao Xu, Fei Gao, and Yanjun Cao. CREPES: Cooperative RElative Pose Estimation System, March 2023. arXiv:2302.01036 [cs].

[117] Nooploop. LinkTrack – Product, 2023. `https://www.nooploop.com/en/linktrack/`.

[118] Nooploop. LinkTrack Datasheet, v2.1 edition.

[119] Nooploop. LinkTrack User Manual, v2.1 edition.

[120] Dae-Ho Kim and Jae-Young Pyun. NLOS Identification Based UWB and PDR Hybrid Positioning System. IEEE Access, 9:102917–102929, January 2021.

[121] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, et al., and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. Nature Methods, 17:261–272, 2020.

[122] Ralph B d'Agostino. An omnibus test of normality for moderate and large size samples. Biometrika, 58(2):341–348, 1971.

[123] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, Andrew Y Ng, et al. Ros: an open-source robot operating system. In ICRA workshop on open source software, volume 3, page 5. Kobe, Japan, 2009.

[124] Rue Henri Fabre. TeraRanger Evo 15m Specification Sheet, 2021.

[125] Richard Hartley and Andrew Zisserman. Multiple view geometry in computer vision. Cambridge university press, 2003.

[126] Luis Pineda, Taosha Fan, Maurizio Monge, Shobha Venkataraman, Paloma Sodhi, Ricky T. Q. Chen, Joseph Ortiz, Daniel DeTone, Austin Wang, Stuart Anderson, Jing Dong, Brandon Amos, and Mustafa Mukadam. Theseus: A Library for Differentiable Nonlinear Optimization, 2022. Publication Title: Advances in Neural Information Processing Systems original-date: 2021-11-18T20:28:27Z.

[127] Mohamad Qadri, Zachary Manchester, and Michael Kaess. Learning Covariances for Estimation with Constrained Bilevel Optimization, September 2023. arXiv:2309.09718 [cs].

[128] Mohamad Qadri and Michael Kaess. Learning Observation Models with Incremental Non-Differentiable Graph Optimizers in the Loop for Robotics State Estimation, September 2023. arXiv:2309.02525 [cs].