Relative Transformation Estimation Based on Fusion of Odometry and UWB Ranging Data

Thien Hoang Nguyen, Student Member, IEEE, and Lihua Xie, Fellow, IEEE

Abstract—In this article, we study the problem of estimating the four-degree-of-freedom (3-D position and heading) robot-to-robot relative frame transformation using onboard odometry and interrobot distance measurements. First, we present a theoretical analysis of the problem, namely, the derivation and interpretation of the Cramér–Rao lower bound, the Fisher information matrix, and its determinant. Second, we propose optimization-based solutions, including a quadratically constrained quadratic programming (QCQP) formulation and its semidefinite programming (SDP) relaxation. Third, based on the theoretical results, we can detect singular configurations as well as measure the uncertainty of each individual parameter. We perform extensive simulations and real-life experiments with aerial robots to show that the proposed QCQP and SDP methods can outperform state-of-the-art approaches, especially in geometrically poor or large measurement noise conditions. In general, the QCQP method provides the best results at the expense of computational time, while the SDP method runs much faster and is sufficiently accurate in most cases.

Index Terms—Optimization, relative localization, sensor fusion, ultrawideband (UWB).

I. INTRODUCTION

MANY multirobot applications, such as search and rescue, environment monitoring, cooperative localization and mapping, target tracking, and entertainment, have gained increasing attention in recent years [1], [2], [3]. For these operations to succeed, it is important that the individual robot’s pose and sensor measurements are expressed in a common reference frame. Therefore, the problem of acquiring the instantaneous relative pose or the initial relative frame transformation (RFT) between the robots is of great interest. In this article, we refer to the former problem as relative pose estimation (RPE) and the latter as relative transformation estimation (RTE), although the terms are sometimes interchangeable in the literature.

When an external reference system is accessible (e.g., from GPS and compass, a priori known map or layout of the environment, or ultrawideband (UWB) anchors with known constellation), the robots’ own poses are readily available, and the relative poses can be easily computed. Hence, both RPE and RTE problems are solved. However, the application will be limited to within the area within the coverage of the external system. For example, the GPS would not be reliable in environments such as indoors, underground, underwater, forests, etc., no prior map would be available for an unknown environment, and localization with UWB anchors is limited to only the area where the anchors are installed.

Thus, much research on using sensor measurements to obtain the relative poses has been put forth. Depending on the exteroceptive sensor (e.g., camera, LiDAR, radar, and UWB) equipped onboard the robot, the interrobot relative measurements can be obtained in the form of relative bearing, distance, position, pose, or some combinations among them [4], [5]. However, one implicit assumption is that the neighbor robot has known shape, size, 3-D model, or markers. Furthermore, the complexity and hardware cost differ greatly depending on the sensor and should be taken into consideration [6]. When using camera or LiDAR, rich information regarding the neighbor robot as well as the environment can be extracted with sophisticated algorithms [7]. In general, the performance can be limited by [6]: 1) the sensor’s field of view (FOV) and detecting range; 2) environmental conditions such as lighting, rain, fog, etc.; 3) the complexity of the framework, which includes data processing (target detection from image/point cloud), target tracking, and reidentification over time; and 4) robustness against false detection and misclassification. In contrast, a relative localization method using only interrobot distance measurements and local self-odometry, which is the main target of this article, can naturally overcome these challenges. Such a solution would be of great appeal for many applications, such as cooperative swarm [8] or leader–follower formation control [9].

In this sense, the UWB sensor offers several key advantages: 1) the measurement is omnidirectional with centimeter-level accuracy and long range (up to hundreds of meters); 2) the measurement is unaffected by lighting conditions; 3) the ranging data are comparatively simpler to model and process; and 4) each UWB tag (and by extension, each robot) can be assigned a unique id without any ambiguity. Furthermore, UWB also provides a communication network [10] and is a more affordable, small, and lightweight sensor for a team of mobile robots compared to LiDAR [7]. On the other hand, the UWB sensor might...
suffer under non-line-of-sight (LOS) or multipath conditions and provide no information about the surrounding environment. A single ranging measurement is also not sufficiently informative for either RPE or RTE task. As a result, designing an appropriate framework that leverages the pros and alleviates the cons of UWB and existing localization systems is a promising direction that has attracted more attention recently [11].

In this article, our focus is on the four-degree-of-freedom (DoF) RTE problem (i.e., estimating the 3-D position and relative heading between the local frames) for a number of reasons. First, if the robots’ odometry data are highly accurate, the RPE and RTE problems can be seen as equivalent since the relative pose can be obtained from the initial frame transformation and the current local poses. Given that state-of-the-art (SOTA) simultaneous localization and mapping (SLAM) systems have achieved great progress in terms of accuracy [7], [12], the RTE problem should be preferred over RPE since the state vector is much smaller. Second, an inertial measurement unit (IMU) is one of the most popular sensors for SLAM systems, thanks to the complementary advantages of fusing IMU and other sensors such as camera, LiDAR, radar, wheel encoder, etc. [13]. One of the advantages of IMU-based SLAMs is that, in principle, the roll and pitch angles can be removed from the state vector [14]. Effectively, the pose estimation problem, and by extension other problems that use odometry as inputs (e.g., pose graph optimization, map merging, point cloud alignment, etc.), can be reduced from six-DoF to four-DoF. Thus, the study of the four-DoF RTE problem would be useful to augment the existing IMU-based SLAM pipelines. Third, we are able to provide detailed theoretical analysis and interpretation for the four-DoF case, which is difficult for the general six-DoF case.

The main contributions of this article include the following:

- theoretical analysis for the four-DoF RTE problem and related subproblems, including the derivation and interpretation of the Cramér–Rao lower bound (CRLB), the Fisher information matrix (FIM), and its determinant;
- optimization-based approaches to solve the four-DoF RTE problem, namely, the quadratically constrained quadratic programming (QCQP) method and its semidefinite programming (SDP) relaxation, with analysis and experimental results to demonstrate the performance and tradeoff;
- a method to directly calculate the uncertainty of the results, detect singular configurations, and find which parameter is highly ambiguous in a given configuration;
- a comprehensive evaluation process that sweeps through all the prominent aspects that can affect the system performance, with which more statistically significant results can be produced and our findings would be more general than previous works.

The rest of this article is organized as follows. In Section II, we review the related works on the RTE problem and highlight our contributions. The preliminaries and problem formulation are presented in Section III. The theoretical analysis is then presented in Section IV, followed by the proposed optimization-based approaches in Section V. Next, simulation and experimental results in Section VI are used to verify the performance of our methods compared to SOTA, as well as compare the advantages and disadvantages of the proposed approaches. Finally, Section VII concludes this article.

II. LITERATURE REVIEW

In single-robot cases, if UWB anchors can be preinstalled in the environment, many methods have been introduced to fuse the UWB-based localization with the existing onboard localization (using IMU, camera, LiDAR, etc.) to reduce the error and accumulated drift [11], [15]. Recent research has relaxed the deployment requirements greatly, to even one anchor at an unknown position [16], [17]. In multirobot cases without UWB anchors, solving the RPE task by combining UWB and vision has been studied in [16] and [18]. However, these approaches still rely mainly on vision for the detection and tracking of neighbor robots, which means the disadvantages regarding limited FOV and detecting range still apply.

If an array of UWB antennas is available on at least one robot, the RPE problem can be solved instantaneously [10], [19], [20], [21]. Thus, camera or LiDAR can be used for other tasks. Such systems, however, require that the robot is relatively large in size in order to accommodate the UWB antenna configuration. Alternatively, various cooperative control or motion scheduling methods have been proposed [8], [22], [23], which can achieve RPE at the cost of time and energy. In [8], an approach for the RPE problem without cooperative control or motion scheduling was also proposed. However, this method only works for 2-D cases and assumes that all robots’ computation, communication, and sensing processes are synchronized. In contrast, we only require a single interrobot ranging link, and no synchronization between the robots is needed, which would be the most general assumptions in terms of practicality. We note that while compasses can also be added to simplify the problem, they might be unreliable in indoors or environments with large metallic structures, which can cause strong magnetic disturbances (e.g., container seaport and airport hangar). Hence, we do not rely on compasses to make our system more resilient to external interference.

Algebraic and analytical solutions for the RTE problem using any combination of range and bearing measurements have been proposed in [4] and [24]. In noisy cases, the solution might be suboptimal, and a refinement step would be necessary. As a result, a common strategy is to use a noniterative method to find an initial solution that is then refined with nonlinear least squares (NLS) or a variant of NLS [25], [26], [27], [28]. Given that many methods can be paired with an NLS refinement step, the quality of the initial guess obtained from the noniterative method would be the main contributing factor when tested with the same dataset. In this article, we compare the proposed methods against previous noniterative methods without the NLS refinement step as well as NLS with and without a good initial guess to better differentiate different approaches.

The main related works are summarized in Table I. Among them, Shariati et al. [29] address the three-DoF problem in 2-D, Martel et al. [25] and Ziegler et al. [30] tackle the same four-DoF problem as ours, and Jiang et al. [27] and Trawny et al. [31] solve the full six-DoF problem. However, the work in [29] follows a
The methods in [25], [27], [30], and [31] are included in our comparison.

The sampling-based solution and, hence, is highly dependent on the number of samples the methods used and obviously would not scale well; works [25] and [31] are susceptible to UWB noise, with the translation error in the order of meters when the standard deviation of the distance measurement error is in the order of tens of centimeters; Ziegler et al. [30] propose an NLS solution with multiple overlapping sliding windows, which relies strongly on having a good initial guess that is not too far from the true value. In challenging cases where the configuration is degenerate or near-degenerate (e.g., near straight line), in [32], it has been shown that even with a good initial guess, such an NLS method can still get stuck in local minima. We note that the methods in [26] and [27] and our method all follow the same idea of reformulating an original nonconvex optimization problem into an SDP problem, but each method solves for a different state vector. However, the method in [26] only addresses the 2-D case, while our method solves the four-DoF case; the method in [27] requires solving a large problem (relative to the related works) with 16 variables and 14 constraints while also ignoring the first distance measurement $d_0$. Our method is much more efficient and accurate. Furthermore, neither [26] nor [27] provides the theoretical analysis of the problem. Last but not least, previous works only employ a limited set of trajectory configurations and noise levels, often limited to just one, in their performance evaluation. In contrast, our work directly addresses all of these issues, improves the performance, and generalizes the results and analysis to no specific trajectory shape, initial conditions, or noise levels.

### III. System Overview

#### A. Notations

A homogeneous transformation matrix in frame $\{A\}$ is denoted as

$$
A_T := \begin{bmatrix}
A_R & A_p \\
0^\top & 1
\end{bmatrix} \in SE(3)
$$

where $A_p \in \mathbb{R}^3$ and $A_R \in SO(3)$ are the position vector and rotation matrix in frame $\{A\}$, respectively. The corresponding quaternion of $A_R$ is $A_q \in \mathbb{H}$. Denote $A_T^B$ and $A_R^B$ as the transformation and rotation matrices from frame $\{B\}$ to $\{A\}$, respectively. The noisy measurement and the estimated value are indicated as $\tilde{\cdot}$ and $\hat{\cdot}$, respectively. $E[\cdot]$ is the expectation of a matrix. For simplicity, denote $T_{ij}$ as the $(i,j)$th element of the matrix $T$ and $x_i$ as the ith element of vector $x$. For a position vector $p \in \mathbb{R}^3$, denote its elements as $p := [p^x, p^y, p^z]^\top$. Finally, let $t_k$ be the time stamp of the latest UWB range measurement $d_k$ and $d_0$ be the distance between the frames’ origin of the two robots [see Fig. 1(a)], which is typically the first measured distance.

#### B. Problem Formulation

Fig. 1(a) shows an overview of the system, which consists of a pair of robots denoted as $R_n, n \in \{1, 2\}$. Each robot is equipped with a UWB sensor and an IMU-based onboard odometry system. In this article, we specifically consider visual-inertial odometry (VIO) as VIO is used in our experiments. However, other IMU-based modalities [7] still apply. Let $\{B_n\}$ and $\{L_n\}$ be the IMU body frame and local odometry frame, respectively. $\{L_n\}$’s $z$-axis aligns with gravity.

During the operation, the collected data include odometry from each robot and interrobot range measurements. At time $t_k$, the available dataset is

$$
J_k = \left\{ (d_i, \hat{\xi}_i^B \hat{p}_i, \hat{\xi}_i^B \hat{q}_i, \hat{\xi}_i^B \hat{q}_i, \hat{\xi}_i^B \hat{q}_i) \right\}_{i=1,...,k}
$$

where $d_i$ is the interrobot UWB ranging measurement and $\hat{\xi}_i^B \hat{p}_i$ and $\hat{\xi}_i^B \hat{q}_i$ are poses of the robots in their respective local frames $\{L_n\}$. In this article, the peer-to-peer two-way time-of-flight UWB ranging scheme is employed to avoid complicated clock synchronization between the sensors [15].

Without loss of generality, we assign $R_1$ as the host robot and $R_2$ as the target robot. Our goal is to estimate the RFT in the world frame $\{L_1\}$

$$
\xi_1 L_1 T := \begin{bmatrix}
\xi_1^L R \\
0^\top \\
1
\end{bmatrix}
$$

(3)

with $\xi_1^L R := C$ and $\xi_1^L p := t$ for simplicity. Since it has been shown that VIO systems have four unobservable directions [33],

### TABLE I

| Method | Estimation problem | Approach |
|--------|-------------------|----------|
| [29]   | 2-D translation, rotation, scale factors | Sampling-based convex optimization |
| [26]   | 2-D translation, heading | SDP |
| [25]   | 3-D translation, heading | Linear |
| [30]   | 3-D translation, heading | NLS |
| [31]   | 3-D translation, rotation | Algebraic |
| [27]   | 3-D translation, rotation | SDP |
| Ours   | 3-D translation, heading | QCQP, SDP |

The methods in [25], [27], [30], and [31] are included in our comparison.
the probability density function of $\hat{d}$ is given by

$$p(\hat{d}, \Theta) = \frac{1}{(2\pi)^{k/2} \sqrt{\det(\Sigma)}} \exp \left[-\frac{1}{2}(\hat{d} - f(\Theta))^\top \Sigma^{-1}(\hat{d} - f(\Theta))\right]$$

where $\Sigma = \sigma_r^2 I_{k \times k}$. Let $\hat{\Theta}$ be an estimate of $\Theta$. The CRLB is the lowest bound for the error covariance matrix of an unbiased estimator, i.e.,

$$\mathbb{E} \left[ (\hat{\Theta} - \Theta)(\hat{\Theta} - \Theta)^\top \right] \geq \text{CRLB} = F^{-1}$$

where $F$ is the FIM. The FIM encodes the amount of information provided by the set of measurements to estimate the parameters. In general, the $(i, j)$th element of $F$ is

$$F_{i,j} := \mathbb{E} \left[ \frac{\partial}{\partial \Theta_i} \ln \left( p(\hat{d}, \Theta) \right) \frac{\partial}{\partial \Theta_j} \ln \left( p(\hat{d}, \Theta) \right) \right]$$

with $c$ being a constant scalar. Under the i.i.d. zero-mean Gaussian noise assumption, the formulation of FIM is [35]

$$F = \left[ \frac{\partial f(\Theta)}{\partial \Theta} \right]^\top \Sigma^{-1} \left[ \frac{\partial f(\Theta)}{\partial \Theta} \right]$$

which can be rewritten as

$$F = \frac{1}{\sigma_r^2} \sum_{i=1}^{k} G_i^\top G_i$$

where

$$G_i = \left[ \frac{\partial f_i(\Theta)}{\partial \theta}, \frac{\partial f_i(\Theta)}{\partial \theta^2}, \frac{\partial f_i(\Theta)}{\partial \theta^3} \right]$$

$$= [\partial z f_i, \partial y f_i, \partial x f_i, \partial \theta f_i]$$

The derivations of the above Jacobians can be found in Appendix A. The FIM can be simplified as

$$F = \frac{1}{\sigma_r^2} \sum_{i=1}^{k} \left[ \begin{array}{ccc} (\partial z f_i)^2 & \ldots & (\partial z f_i)(\partial \theta f_i) \\ \vdots & \ddots & \vdots \\ (\partial \theta f_i)(\partial z f_i) & \ldots & (\partial \theta f_i)^2 \end{array} \right]$$

$$= \frac{1}{\sigma_r^2} \sum_{i=1}^{k} G_i^\top G_i = \frac{1}{\sigma_r^2} J^\top J$$

where the $i$th row of the matrix $J$ is $G_i$, i.e., $J := \begin{bmatrix} G_1 \\ \vdots \\ G_k \end{bmatrix}$. From (16), it can be seen that the FIM: is either positive definite or semidefinite; depends on the number of available measurements ($k$); and depends on the precision of the measurements ($\sigma_r$) as well as the Jacobians ($G_i$). If the FIM is nonsingular, i.e., $\det(F) \neq 0$, the CRLB can be obtained as the inverse of FIM.
For evaluation purposes, we additionally define the translation and heading CRLBs as

$$
CRLB_t = \frac{1}{\sigma^2} \sum_{i=1}^{3} |\mathbf{F}^{-1}|_{i,i} \quad \text{and} \quad CRLB_\theta = |\mathbf{F}^{-1}|_{4,4}
$$

(17)

The CRLB$_t$ and CRLB$_\theta$ are used to compare the mean square error (MSE) of different methods against the theoretical bound.

B. Determinant of the FIM

Applying the Cauchy–Binet formula to \( \mathbf{F} = \frac{1}{\sigma^2} \mathbf{J}^\top \mathbf{J} \), we have

$$
\det(\mathbf{F}) = \frac{1}{\sigma^2} \sum_{1 \leq i_1 < i_2 < j_1 < j_2 \leq k} (\det(\Lambda))^2
$$

where \( \Lambda \) denotes the 4×4 matrix consisting of the \( j_1, j_2, j_3, j_4 \)th rows of \( \mathbf{J} \). For clarity, in the following, we focus on \( \Lambda \) with \( (1−4) \)th rows of \( \mathbf{J} \), but the analysis can be generalized to any \((j_1−j_4)\)th rows. As shown in Appendix A, we can write

$$
\Lambda := \begin{bmatrix}
\mathbf{G}_1 \\
\mathbf{G}_2 \\
\mathbf{G}_3 \\
\mathbf{G}_4
\end{bmatrix} = \begin{bmatrix}
\mathbf{u}_1^\top \\
\mathbf{u}_2^\top \\
\mathbf{u}_3^\top \\
\mathbf{u}_4^\top
\end{bmatrix} \begin{bmatrix}
\mathbf{G}_1 \\
\mathbf{G}_2 \\
\mathbf{G}_3 \\
\mathbf{G}_4
\end{bmatrix}
$$

(19)

where \( \mathbf{u}_i = \frac{\alpha}{\sigma \omega_2} \mathbf{p}_i \) (\( i \in \{1, \ldots, 4\} \)) is the unit vector parallel to \( \mathbf{a}_2 \mathbf{p}_i \) and

$$
\Phi_i = \left( \mathbf{u}_i \times \mathbf{C}_{\omega_3}^z \mathbf{p}_i \right) \cdot \mathbf{u}_i = \rho_i \sin \gamma_i
$$

with \( \gamma_i = \frac{\pi}{2} - \angle(\mathbf{u}_i \times \mathbf{C}_{\omega_3}^z \mathbf{p}_i, \mathbf{u}_i) \), \( \rho_i \) is the length of the projection of \( \mathbf{C}_{\omega_3}^z \mathbf{p}_i \) on the \( xy \) plane of \( (\mathcal{L}_i) \), and \( \mathbf{u}_z = [0, 0, 1]^\top \) is the unit vector in the z-axis.

By applying the Laplace expansion along the last column of \( \Lambda \) in (19), we have

$$
\det(\Lambda) = -\Phi_1 T_1 + \Phi_2 T_2 - \Phi_3 T_3 + \Phi_4 T_4
$$

(21)

where the minors are

$$
T_1 = \mathbf{u}_2^\top \mathbf{u}_3, \quad T_2 = \mathbf{u}_3^\top \mathbf{u}_4, \quad T_3 = \mathbf{u}_4^\top \mathbf{u}_1, \quad T_4 = \mathbf{u}_1^\top \mathbf{u}_2
$$

(22)

For any vectors \( \mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{R}^3 \), we have

$$
\det \begin{bmatrix}
\mathbf{a}^\top \\
\mathbf{b}^\top \\
\mathbf{c}^\top
\end{bmatrix} = \det(\mathbf{a} \times \mathbf{b} \times \mathbf{c}) = (\mathbf{a} \times \mathbf{b}) \cdot \mathbf{c}
$$

$$
= ||\mathbf{a}|| ||\mathbf{b}|| ||\mathbf{c}|| \sin \angle(\mathbf{a}, \mathbf{b}) \cos \angle(\mathbf{a}, \mathbf{b}, \mathbf{c}).
$$

(23)

As such, \( T_1 \) can be written as

$$
T_1 = (\mathbf{u}_2 \times \mathbf{u}_3) \cdot \mathbf{u}_4 = ||\mathbf{u}_2|| ||\mathbf{u}_3|| ||\mathbf{u}_4|| \sin \angle(\mathbf{u}_2, \mathbf{u}_3) \cos \angle(\mathbf{u}_2 \times \mathbf{u}_3, \mathbf{u}_4)
$$

$$
= ||\mathbf{u}_2|| ||\mathbf{u}_3|| ||\mathbf{u}_4|| \sin \alpha_1 \sin \beta_1
$$

(24)

where \( \alpha_1 = \angle(\mathbf{u}_2, \mathbf{u}_3) \) and \( \beta_1 = \frac{\pi}{2} - \angle(\mathbf{u}_2 \times \mathbf{u}_3, \mathbf{u}_4) \). Similarly, we can write

$$
T_2 = (\mathbf{u}_1 \times \mathbf{u}_3) \cdot \mathbf{u}_4 = ||\mathbf{u}_1|| ||\mathbf{u}_3|| ||\mathbf{u}_4|| \sin \alpha_2 \sin \beta_2
$$

$$
T_3 = (\mathbf{u}_1 \times \mathbf{u}_2) \cdot \mathbf{u}_4 = ||\mathbf{u}_1|| ||\mathbf{u}_2|| ||\mathbf{u}_4|| \sin \alpha_3 \sin \beta_3
$$

$$
T_4 = (\mathbf{u}_1 \times \mathbf{u}_2) \cdot \mathbf{u}_3 = ||\mathbf{u}_1|| ||\mathbf{u}_2|| ||\mathbf{u}_3|| \sin \alpha_4 \sin \beta_4.
$$

(25)

Combining the above equations, we have

$$
\det(\mathbf{F}) = \frac{1}{\sigma^2} \sum_{S} \left( \sum_{i=1}^{4} (-1)^i \Phi_i T_i \right)^2
$$

$$
= \frac{1}{\sigma^2} \sum_{S} \left( \sum_{i=1}^{4} (-1)^i \left( ||\mathbf{u}_2|| ||\mathbf{u}_3|| ||\mathbf{u}_4|| \sin \angle(\mathbf{u}_2, \mathbf{u}_3) \cos \angle(\mathbf{u}_2 \times \mathbf{u}_3, \mathbf{u}_4) \right) \left( ||\mathbf{u}_1|| ||\mathbf{u}_p|| \right) \right)^2
$$

$$
= \frac{1}{\sigma^2} \sum_{S} \left( \sum_{i=1}^{4} (-1)^i \rho_i \sin \alpha_i \sin \beta_i \sin \gamma_i \right)^2
$$

(26)

where \( S = \{(j_1, j_2, j_3, j_4) | 1 \leq j_1 < j_2 < j_3 < j_4 \leq k\} \), \( m = j_1, (l, p, q) \in \{ j_1, j_2, j_3, j_4 \} \setminus m, l < p < q \), and

$$
\alpha_i = \angle(\mathbf{u}_1, \mathbf{u}_p), \quad \beta_i = \frac{\pi}{2} - \angle(\mathbf{u}_1 \times \mathbf{u}_p, \mathbf{u}_q), \quad \gamma_i = \frac{\pi}{2} - \angle(\mathbf{u}_z \times \mathbf{C}_{\omega_3}^z \mathbf{p}_m, \mathbf{u}_m), \quad \rho_i = ||\mathbf{C}_{\omega_3}^z \mathbf{p}_m|| \sin \angle(\mathbf{u}_z \times \mathbf{C}_{\omega_3}^z \mathbf{p}_m, \mathbf{u}_m). \quad \text{(27)}
$$

Fig. 2 illustrates the components that build up \( \det(\mathbf{F}) \). Note that \( \beta_i \) is defined only when \( \mathbf{u}_1 \times \mathbf{u}_p \neq 0 \), i.e., \( \mathbf{u}_1 \) and \( \mathbf{u}_p \) are nonparallel. Otherwise, if \( \mathbf{u}_1 \times \mathbf{u}_p = 0 \), then \( T_i = 0 \) and \( \beta_i \) is not relevant. Similarly, \( \gamma_i \) is defined only when \( \mathbf{u}_z \times \mathbf{C}_{\omega_3}^z \mathbf{p}_i \neq 0 \), i.e., \( \mathbf{u}_z \) and \( \mathbf{C}_{\omega_3}^z \mathbf{p}_i \) are nonparallel; otherwise, \( \Phi_i = 0 \) and \( \gamma_i \) is
irrelevant. In the next section, we will discuss in more details the physical meaning of $\alpha$, $\beta$, $\gamma$, and $\rho$.

**Proposition IV1:** From the general problem of 3-D RTE with unknown $\theta$ (i.e., without a common heading reference between the robots), one can derive the subproblems, the corresponding state vector, and the determinant of the FIM as follows.

1) 3-D RTE with known $\theta$, $\Theta := [t^x, t^y, \theta]^T$,

$$\det(\mathbf{F}) = \frac{1}{\sigma^2} \sum_{1 \leq i < j \leq k} \sin^2 \alpha \sin^2 \beta$$

(28)

where $\alpha = \angle(u_i, u_j)$, $\beta = \frac{\pi}{2} - \angle(u_i \times u_j, u_l)$.

2) 2-D RTE with unknown $\theta$, $\Theta := [t^x, t^y, \theta]^T$,

$$\det(\mathbf{F}) = \frac{1}{\sigma^2} \sum_{3} \left[ \sin \gamma_i \right]^2$$

(29)

where $S = \{(j_1, j_2, j_3) \mid 1 \leq j_1 < j_2 < j_3 \leq k\}$, $\alpha_i = \tan \left( |u_p \times u_q| \right)$, $u_p, u_q$.

$\gamma_i = \frac{\pi}{2} - \angle(u_z, C_{a_2}^{L_2} p_m, u_1)$

with $l = j_1$, $p, q \in \{j_1, j_2, j_3\} \setminus l$, $p < q$, and $[.]_z$ extracts the $z$ element of the argument vector in $\mathbb{R}^3$.

3) 2-D RTE with known $\theta$, $\Theta := [t^x, t^y]^T$,

$$\det(\mathbf{F}) = \frac{1}{\sigma^2} \sum_{1 \leq i < j \leq k} \sin^2 \alpha$$

(30)

where $\alpha = \angle(u_i, u_j)$.

**Proof:** See Appendix B.

C. **Geometric Interpretation of det(F)**

We can interpret the components that build up $\det(\mathbf{F})$ in (26) and (27) as follows: $u_{\parallel}$ is the unit vector parallel to the relative position vector in the world frame $\{L_1\}$ [see Fig. 2(a)]. $C_{a_2}^{L_2} p_m$ is the local position vector of the target robot $R_2$ as measured in the world frame. $T_i$ is the signed volume of the parallelepiped with three vectors $u_i, u_p, u_q$ as edges [see Fig. 2(b)]. $\Phi_i$ is the signed volume of the parallelepiped with three vectors $C_{a_2}^{L_2} p_m, u_m, u_z$ as edges [see Fig. 2(c)]. Essentially, for every set of four measurements at $j_1, \ldots, j_4$, $\det(\mathbf{A})$ is a combination of both $T_i$ and $\Phi_i$. Finally, $\det(\mathbf{F})$ is computed over all the possible sets of four measurements $S$. The larger $\det(\mathbf{F})$ is, the smaller the uncertainty volume [36].

Intuitively, $T_i = |\sin \alpha_i| |\sin \beta_i|$ represents the information gain regarding the translation parameters $(t^x, t^y, t^z)$ and is directly correlated with the volume occupied by the relative position vectors (encoded by $\alpha_i$ and $\beta_i$) but not the absolute positions of the robots. This suggests that in order to improve the estimates of $\mathbf{t}$, one should focus on enlarging the angles between the measurements, which is exactly the strategy that was presented in [37] for a static target. On the other hand, $\Phi_i = |\rho_i| \sin \gamma_i$ represents the information gain regarding the relative heading parameter ($\theta$) and is directly influenced by the horizontal displacement of the target robot in its local frame ($\rho_i$) and how perpendicular is the relative position vector ($u_{\parallel}$) to the vertical plane that contains $C_{a_2}^{L_2} p_m$ (which has the normal vector $C_{a_2}^{L_2} p_m \times u_z$), measured by $\sin \gamma_i$.

For a given configuration, if $T_i = 0$ or $\Phi_i = 0$ or $\sum_{i=1}^{4} (-1)^i \Phi_i T_i = 0$ for the whole set $S$, then $\det(\mathbf{F}) = 0$, in which case the associated parameters are at the critical points of $\mathbf{f}(\Theta)$ and their uncertainty cannot be reduced regardless of the number of measurements.

1) If the robots move in parallel, so do the relative position vectors, i.e., $u_i \parallel u_p$ or $\alpha_i = \rho_i = 0 \forall i$, then $T_i = 0 \forall i$ and $\det(\mathbf{F}) = 0$. In this case, the relative translation $\mathbf{t}$ cannot be fully resolved [see Fig. 3(a)].

2) If the trajectories of the robots are linear and on the same 2-D plane, then their relative position vectors are also on the same 2-D plane, which leads to $(u_i \times u_p) \perp u_q$ or equivalently $\beta_i = 0 \forall i$ and, hence, $T_i = 0 \forall i$ and $\det(\mathbf{F}) = 0$. In this case, the solution can be recovered up to a flip ambiguity [see Fig. 3(b)].

3) If the target robot is stationary or only moves on the $z$-axis, i.e., $C_{a_2}^{L_2} p_m = 0$ or $\rho_i = 0$, then $\Phi_i = 0 \forall i$ and $\det(\mathbf{F}) = 0$. In this case, the relative heading $\theta$ cannot be fully resolved [see Fig. 3(c)].

4) If the host robot is stationary, then $\sum_{i=1}^{4} (-1)^i \Phi_i T_i = 0$ and $\det(\mathbf{F}) = 0$. In this case, both $\mathbf{t}$ and $\theta$ cannot be fully resolved since the solution is invariant to rotation around the host robot [see Fig. 3(d)].

These examples with clear physical interpretations have also been stated separately in [8] and [38]. In addition, we provide...
simulation results in Section VI-D to demonstrate the above observations. Besides identifying degenerate cases, det(F) can also be used to study the optimal configurations and online trajectory planning. Specifically, from a given number of measurements, by maximizing det(F) (or equivalently minimizing the uncertainty volume) subjected to limited sensing range and other motion constraints, we can find the optimal configurations. The online trajectory planning problem can also use the D-optimal cost function \( J = -\ln(\text{det}(F)) \) to plan the next best sensing positions given the current estimates. Our system does not include any trajectory planning or motion scheduling step, as these studies are outside the scope of this article. Instead, we assume that the trajectories will be sufficient to avoid degenerate cases, detect whether the configuration is degenerate before commencing the estimation process, and measure the uncertainty of each parameter, as will be described in Section V-D.

V. PROPOSED APPROACHES

In this section, the proposed methods are presented in detail. First, we briefly summarize the squared distance weighted least squares (SD-WLS) problem [28]. Then, the SD-WLS problem is reformulated as a nonconvex QCQP problem and further as an SDP relaxation problem.

A. Four-DoF Squared Distance Weighted Least Squares

The true interrobot distance at time \( t_k \) can be written as
\[
d_k = \|w_k\| = \sqrt{w_k^\top w_k}
\]
where \( w_k = t + C_{a2}^{t} \bar{p}_k - \bar{t}_i \). The ranging error vector is
\[
e_r := \left( \|w_1\| - \bar{d}_1, \|w_2\| - \bar{d}_2, \ldots, \|w_k\| - \bar{d}_k \right)^\top.
\]
The squared distance measurement
\[
d_k^2 = d_k^2 + 2d_k\eta_k + \eta_k^2
\]
has the noise term \( \nu_k := 2d_k\eta_k + \eta_k^2 \), which is not zero-mean Gaussian. However, this non-Gaussian probability distribution function can be approximated as one [28]
\[
\bar{s}_k = \bar{d}_k^2 + \bar{\nu}_k = w_k^\top w_k + \bar{\nu}_k
\]
\[
\bar{s}_k \approx \bar{d}_k^2 - \bar{\nu}_k, \quad \bar{\nu}_k := \mathbb{E}[\nu_k] = \Sigma_{k,k}
\]
\[
\eta_k^* = [\eta_1^*, \ldots, \eta_k^*] \sim \mathcal{N}(\eta^*:0,\Sigma).
\]
The elements of the covariance matrix \( \Sigma \) are computed as
\[
S_{i,i} := \mathbb{E}[(\nu_i - \bar{\nu}_i)^2] = \Sigma_{i,i} \left( 4\bar{d}_i^2 + 2\Sigma_{i,i} \right)
\]
\[
S_{i,j} := \mathbb{E}[(\nu_i - \bar{\nu}_i)(\nu_j - \bar{\nu}_j)] = \Sigma_{i,j} \left( 4\bar{d}_i\bar{d}_j + 2\Sigma_{i,j} \right).
\]
The four-DoF SD-WLS problem is defined as
\[
\min_{\Theta} \frac{1}{2} e_s^\top S^{-1} e_s
\]
where \( e_s \) is the vector of the squared distance errors
\[
e_s := [w_1^\top w_1 - \bar{s}_1, w_2^\top w_2 - \bar{s}_2, \ldots, w_k^\top w_k - \bar{s}_k]^\top.
\]
The original SD-WLS [28] was established for the 2-D case. Here, we change the parameters to that of the four-DoF case.

B. Nonconvex QCQP

Expanding and simplifying \( w_k^\top w_k \) lead to
\[
w_k^\top w_k = (t + C_{a2}^{t} \bar{p}_k - \bar{t}_i)^\top (t + C_{a2}^{t} \bar{p}_k - \bar{t}_i) = t^\top t + C_{a2}^{t} \bar{p}_k + C_{a2}^{t} C_{a2}^{t} \bar{p}_k + 2(t - \bar{t}_i)(C_{a2}^{t} \bar{p}_k - \bar{t}_i) = t^\top t + 2C_{a2}^{t} \bar{p}_k - 2\bar{t}_i C_{a2}^{t} \bar{p}_k - \bar{t}_i^\top \bar{t}_i.
\]
Let \( x := [x_1, x_2, \ldots, x_9]^\top \) be the state vector
\[
x := [t^x, t^y, t^z, \cos \theta, \sin \theta, t^x \cos \theta + t^y \sin \theta, t^y \cos \theta - t^x \sin \theta, (t^x)^2 + (t^y)^2 + (t^z)^2, 1]^\top \in \mathbb{R}^{9\times 1}.
\]
Each row of \( e_s \) can be rearranged as \( w_k^\top w_k - \bar{s}_i = A_k x \), where
\[
A_k := [-2\varphi_i^x, -2\varphi_i^y, 2(\alpha_i^z - \varphi_i^z), -2(\varphi_i^x \alpha_i^z + \varphi_i^y \varphi_i^z), 2(\varphi_i^x \alpha_i^z - \varphi_i^y \varphi_i^z), 2\alpha_i^z, \theta_i, 1, \varepsilon_i] \in \mathbb{R}^{9\times 9}
\]
\[
\varepsilon_i := C_{a2}^{t} \bar{p}_k - \bar{t}_i \bar{p}_k - 2\varphi_i^x \alpha_i^z - \bar{s}_i \quad \text{and} \quad i = 1, \ldots, k,
\]
(37)
(38)
(39)
(40)
(41)
(42)
(43)
(44)
(45)
Proposed Algorithms for QCQP algorithm given \( P \)
do
5
\[
\text{Calculate the } i\text{th row of } S \text{ according to (35)}
\]
\[
\text{Calculate the } i\text{th row of } B \text{ according to (40)}
\]
\[
i \leftarrow i + 1
\]
end while
\[
P_0 \leftarrow B^\top S^{-1} B
\]
if QCQP algorithm then
\[
\hat{x} \leftarrow \text{solve the QCQP problem (43) given } P_0, \{ P_i, r_i \}_{i=1}^5
\]
else if SDP algorithm then
\[
\hat{X} \leftarrow \text{solve the SDP problem (48) given } P_0, \{ P_i, r_i \}_{i=1}^5
\]
\[
[U, S, V] \leftarrow \text{svdskech}(\hat{X})
\]
if rank(\( \hat{X} \)) == 1 then
\[
\hat{x} \leftarrow U \text{ else } \hat{x} \leftarrow \sqrt{S_{11}} U_{1:9,1}
\]
end if
\[
\text{if } \hat{x}_9 < 0 \text{ then } \hat{x} \leftarrow -\hat{x}
\]
\[
\tilde{t} \leftarrow \hat{x}_{1:3}, \tilde{\theta} \leftarrow \text{atan2}(\hat{x}_5, \hat{x}_4)
\]
\[
\Theta \leftarrow [\tilde{t}^\top, \tilde{\theta}]^\top
\]

\[
(t_x^2 + t_y^2 + t_z^2) + (t_x) \leftrightarrow x_1^2 + x_2^2 + x_3^2 = d_0^2
\]
\[
\implies x^\top P_5 x = r_5
\] (44)

where
\[
P_1 = \text{sparse}([4, 5], [4, 5], [1, 1], 9, 9), r_1 = 1
\]
\[
P_2 = \text{sparse}([1, 2, 9], [4, 5, 6], [1, 1, -1], 9, 9), r_2 = 0
\]
\[
P_3 = \text{sparse}([2, 1, 9], [4, 5, 7], [1, -1, -1], 9, 9), r_3 = 0
\]
\[
P_4 = \text{sparse}([1, 2, 3, 8], [1, 2, 3, 9], [1, 1, -1, 9, 9]), r_4 = 0
\]
\[
P_5 = \text{sparse}([1, 2, 3], [1, 2, 3, 9], [1, 1, 1, 9, 9]), r_5 = d_0^2
\] (45)

Here, we use the MATLAB sparse\(^1\) function to succinctly create a sparse matrix that contains mostly zeros. We note that the constraint \( i = 5 \) is included only if \( d_0 \), i.e., the UWB measurement that “connects” the frames’ origin, is available; otherwise, it is removed. This design choice makes our system more general, since, in certain scenarios, \( d_0 \) might not exist. For example, if one robot starts moving while the other robot is not ready for operation, or if the starting points are not within LOS.

C. SDP Relaxation

1) Problem Reformulation: We follow the steps outlined in [39] to obtain the SDP relaxation formulation of the QCQP problem (43). First, observe that
\[
x^\top P_1 x = \text{Tr}(x^\top P_1 x) = \text{Tr}(P_1 xx^\top).
\] (46)

By using a new variable \( X = xx^\top \), an equivalent formulation of (43) can be written as
\[
\min_x \text{Tr}(P_0 X)
\]
s.t. \( \text{Tr}(P_i X) = r_i, \quad i = 1, \ldots, 5 \)
\[
\mathbf{X} \succeq 0, \text{ rank}(\mathbf{X}) = 1
\] (47)

where \( \mathbf{X} \succeq 0 \) indicates that \( \mathbf{X} \) is positive semidefinite. By dropping the nonconvex rank constraint, we obtain the SDP relaxation problem as
\[
\min_x \text{Tr}(P_0 X)
\]
s.t. \( \text{Tr}(P_i X) = r_i, \quad i = 1, \ldots, 5 \)
\[
\mathbf{X} \succeq 0
\] (48)

The advantage of the SDP problem (48) over the original NP-hard problems (43) and (47) is that it can be solved in polynomial time. In the noiseless or low-noise cases, the SDP relaxation can be expected to be tight, i.e., solving either the relaxed and original problems is equivalent [39]. A problem-specific explanation that is also applicable to our case can be found in [26, Lemma 1].

2) Recover the Original Solution: Let \( \hat{x} \) and \( \hat{X} \) be the solution of (43) and (48), respectively. If \( \text{rank}(\hat{X}) = 1 \), then the low-rank decomposition \( \hat{X} = \hat{x}(\hat{x})^\top \) will provide the feasible and optimal solution \( \hat{x} \). If \( \text{rank}(\hat{X}) > 1 \), the rank-one decomposition process in [39] can be used to extract \( \hat{x} \). Using SVD to decompose \( \hat{X} \) as
\[
\hat{X} = \sum_{i=1}^{n} \lambda_i q_i q_i^\top, \quad \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n > 0
\] (49)

where \( \lambda_i \) and \( q_i \) are the eigenvalues and respective eigenvectors, the best rank-one approximation is
\[
\hat{x}_{1} = \lambda_1 q_1 q_1^\top.
\] (50)

Finally, since \( x_9 \) should be positive from the way \( x \) is constructed [see (39)], the sign of the final solution is flipped if \( \hat{x}_9 \) is negative. Our methods are summarized in Algorithm 1.

D. Uncertainty Estimation

Based on the theoretical analysis in Section IV, our method can detect whether the current configuration is singular and calculate the uncertainty of the estimate for each parameter. These capabilities make our system more favorable for practical applications. In addition, we also take into account other real-life issues such as accounting for spatial–temporal offsets between the sensors as well as rejecting UWB outliers, which we refer the reader to our previous work [40] for more details.

\(^1\)Online. Available: https://www.mathworks.com/help/matlab/ref/sparse.html
1) Singular Configuration Detection: Singular configurations refer to the cases where multiple solutions exist for at least one of the parameters given the available measurements [31]. In the physical sense, this happens if the target robot’s trajectory can be flipped, rotated, or a combination of both in the world frame and the distance measurements will be the same [8]. This observation corroborates our interpretation of the intuitive unobservable cases, as discussed previously in Section IV-C. In previous works listed in Table I, neither the detection method for singular configurations nor measures of the uncertainty of the estimates were provided.

In the noiseless case, if one parameter is unobservable in a given configuration, the associated row and column in the FIM will become zeros (no information gain from the available measurements). As such, the singular configurations will manifest in the form of the FIM losing rank in noiseless cases, or the condition number of the FIM approaches infinity in noisy cases. Our detection scheme for singular configuration works as follows. First, we evaluate the FIM using the analytical formula in (16) at the latest estimates $\hat{\Theta}$. Then, we compute the condition number of the estimated FIM $\kappa(F)$. A configuration is deemed singular if $\kappa(F)$ is larger than a threshold, which can be determined empirically. In practice, before the first optimization, we check that the sample variance based on recent positions of the robots is higher than a threshold (empirically set as 0.05 m for the last 100 positions in our experiments); otherwise, the optimization process is skipped. This simple check ensures that 1) both the robots are moving and 2) there are motions on all axes, which is sufficient to avoid simple singular cases such as one robot is static or both the robots are moving in parallel. In addition, it was observed that if the poses only change marginally in $x$ and $y$ directions, such as when the quadrotors are hovering, the estimates do not improve regardless of how many new measurements are added. As such, we keep performing this check during the mission and skip any unnecessary updates.

Remark VI.1: In the noiseless case, the singular configurations will also manifest in the form of matrix $P_0$ in (43) losing rank, as the optimal solution of the QCQP problem is not unique. As a result, the singular cases can be detected by checking whether $P_0$ is rank deficient, which can avoid computing the FIM entirely. However, in our real-life experiments, the noise will generally make $P_0$ full rank even when the robots just move slightly. Hence, this method was not used.

2) Uncertainty of the Estimates: Let CRLB be the CRLB computed with the estimated $\hat{\Theta}$ instead of the true $\Theta$. Following [36], the 95% confidence interval for each parameter of $\Theta$ is

$$CI_i = [\hat{\Theta}_i - 1.96 \sqrt{\text{CRLB}_{i,i}}, \hat{\Theta}_i + 1.96 \sqrt{\text{CRLB}_{i,i}}]$$

(51)

for $i \in \{1, \ldots, 4\}$. In addition, CRLB, or generally the inverse of the observed FIM, can be used as the error covariance matrix of the maximum likelihood estimator [41].

In essence, at time $t_k$, our framework reports the current estimates $\hat{\Theta}$ and the corresponding uncertainty metrics, including the condition number $\kappa(F_k)$ and the standard error $\sigma_{\hat{\Theta}_i} := \sqrt{\text{CRLB}_{i,i}}$ for each parameter ($i \in \{1, \ldots, 4\}$). If $\kappa(F_k)$ and $\sigma_{\hat{\Theta}_i}$ are sufficiently large, the trajectory configuration can be recognized as singular, with $\Theta_i$ being the unobservable parameter. With previous methods, in particular the NLS method, the stopping criterion is typical when the gradient step is smaller than a threshold, which can hold true at the local minima of any parameter. In contrast, we monitor $\kappa(F_k)$ and $\sigma_{\hat{\Theta}_i}$ as the stopping condition and only stop the optimization when all the uncertainty estimates are lower than a threshold (or until a timeout). Hence, even though our method may not be able to resolve the degenerate cases completely, we can at least limit the number of ambiguous parameters to the minimum and improve the overall accuracy.

E. Extension to Multirobot Scenarios

While we only focus on two-robot scenarios in this article, it is important to discuss the scalability of our method in general multirobot cases. A simple approach of copying the same estimator to each pair of robots in the sensing graph is a valid solution and has been demonstrated in [8] and [42]. However, in such a scheme, the computation burden on each robot will scale with the number of connected neighbors. Furthermore, there are no embedded mechanisms to ensure the consistency of the estimates across the network, combine the estimates to improve the global results, or let the system run in an asynchronous and decentralized manner. Hence, using a distributed optimization method designed for multirobot systems [43] would be a better choice. In particular, we can alter our SDP algorithm into distributed formulation following existing techniques such as [44], where the network is divided into clusters, which can be solved in parallel using only local information. Hence, our approach is scalable and can be used in general cases. Alternatively, the asynchronous parallel alternating direct method of multipliers approach introduced in [45], which has been expanded for the range-based relative localization problem in [30], is a potential solution for our problem. However, how to determine an efficient sliding window size to reduce computation demand for each robot has not been well considered. Hence, tackling these open questions, reformulating our QCQP problem (43) into a distributed framework, leveraging its structure to improve the performance, and reducing the computation demand are worthy topics for future works.

With regard to communication, in [30], it has been shown that the average traffic between two agents in an odometry and range-based relative formation system would be less than 2 kBs/s (using ROS). This would take up much less bandwidth than SOTA vision-based systems [46], which can require MB/s per robot for keyframe data alone. As such, communication is not a critical issue with our method. However, the communication rate can be reduced even more by leveraging our findings. From the analysis in Section IV-C, we have shown that there are motions during which the estimation results for some of the parameters would not improve (for quadrotors: hovering, yawing while hovering, ascending/ descending, etc.). As such, when the robot is moving in such manners, it can stop sending data, and the performance would not be affected. Hence, with our method, each robot can
adding Gaussian noises with $\sigma_0^t = 0.5$ m and $\sigma_0^\theta = 10^5$ to the ground truth values.

5) All the methods are presented with the same processed input data (degenerate configurations checked, outlier rejection scheme applied, and spatial–temporal offsets compensated).

We evaluate the translation and heading parameters separately, with the estimated translation and heading error denoted as $e_t = \| t - \hat{t} \|$ and $e_\theta = | \theta - \hat{\theta} |$, respectively. The average errors over all runs $\bar{e}_t$ and $\bar{e}_\theta$, the root-mean-square errors $\text{RMSE}_t$ and $\text{RMSE}_\theta$, the MSEs against the CRLBs, and the solver time are the subjects of comparison.

B. Simulation

The simulations are designed to evaluate different factors in isolation. We focus on two essential factors that can affect the performance: trajectory configuration and signal noises. As previously mentioned, $d_0 = \| t \|$ is the true initial relative distance between the robots. Let $R_{\text{max}}$ be the maximum moving radius of the robots from their initial positions, i.e., all the possible trajectories are confined in a sphere centered at the local origin with a radius $R_{\text{max}}$ [see Fig. 1(a)]. The standard deviation of UWB and odometry data is denoted as $\sigma_r$ and $\sigma_o$, respectively.

In previous works, the simulations are done for a specific scenario: the trajectories’ shape and size are fixed, and often only one value of $\Theta$ is tested. This might undermine the generalizability of the observations as well as the conclusions. In this article, we aim for more universal and comprehensive results. To this end, for each method, we perform 100 Monte Carlo simulations with the true relative translation $t$ uniformly sampled on a sphere centered at $0^3 \times 1$ with a radius $d_0$ and the true relative heading $\theta$ randomly sampled between $[-\pi, \pi]$. Then, in each simulation, the robots’ trajectories are generated randomly. Each trajectory consists of 20 poses with the distance to the origin no larger than $R_{\text{max}}$. Finally, the odometry data and UWB data are generated from the noise-free data by adding Gaussian noises. In this manner, the results should be indicative of all possible trajectories’ configuration, shape, size, and relative transform) that can be generated from a given condition specified by $d_0$ and $R_{\text{max}}$.

1) Effect of Trajectory Configuration: Fig. 5 shows the main results. Overall, our proposed QCQP method provides the best performance, especially in more challenging scenarios (large $d_0$ and small $R_{\text{max}}$). The second best results come from either our SDP method or the NLS method. Regardless of initial guess, our QCQP method is the best option in terms of accuracy. With a rough starting point, the NLS method would be the better option when computation resource is limited. However, as a good initial guess can be time consuming to obtain during real deployment (especially when the robots are far apart) or even unobtainable (when the robots do not start working at the same time, for example), our SDP method would be the more practical option.

Generally, it can be seen that both $d_0$ and $R_{\text{max}}$ affect the performance of all the methods. The main observations are: the larger the $d_0$, the larger the errors, while $R_{\text{max}}$ has the opposite effect; $d_0$ affects the translation error $\bar{e}_t$ more noticeably than...
Fig. 5. Estimation errors with varying $R_{\text{max}}$ and $d_0$. Top: translation error $e_t$. Bottom: heading error $e_{\theta}$. All simulations are run with $\sigma_o = 0.001$ m and $\sigma_r = 0.1$ m. As $d_0$ increases or $R_{\text{max}}$ decreases, the scale of the errors increases noticeably.

Fig. 6. Average translation $\bar{e}_t$ (m) and heading $\bar{e}_\theta$ (rad) errors with varying UWB noise ($\sigma_r$) and odometry noise ($\sigma_o$). All simulations are run with $d_0 = 50$ m and $R_{\text{max}} = 10$ m.

the heading error $e_{\theta}$, while it is the reverse for $R_{\text{max}}$; and the performance of the methods is more similar when $d_0$ is smaller or $R_{\text{max}}$ is larger.

In relation to our theoretical findings, the combination of $d_0$ and $R_{\text{max}}$ would limit the maximum relative angles between successive measurements (which then limits $\alpha$ and $\beta$), while $R_{\text{max}}$ would limit the maximum displacement of the target robot on the horizontal plane (which then limits $\rho$). As such, $d_0$ and $R_{\text{max}}$ together affect $T_t$ and, consequently, the relative translation $t$, while $R_{\text{max}}$ directly changes $\Phi_i$, and, consequently, the relative heading $\theta$. In principle, we have $\sin \alpha_{\text{max}} \simeq D/d_0$.

From a practical perspective, a situation with larger $d_0$ or smaller $R_{\text{max}}$ would be more difficult. Either increasing $d_0$ or decreasing $R_{\text{max}}$ would improve the relative translation estimates, while increasing $R_{\text{max}}$ for the target robot would improve the relative heading estimates.

2) Effect of Noise Levels: All the simulations in this evaluation are run with $d_0 = 50$ m and $R_{\text{max}} = 10$ m to imitate an outdoor scenario. The UWB noise $\sigma_r$ and odometry noise $\sigma_o$ vary from ground truth ($\sim 0.001$ m) to GPS ($\sim 1$–$10$ m) level of accuracy. Fig. 6 illustrates the general influence of $\sigma_r$ and $\sigma_o$ on the estimation errors. Fig. 7 shows the detailed results under extreme conditions.

Overall, from Fig. 6, it can be seen that in situations where measurements are very accurate ($\sigma_r < 0.1$ and $\sigma_o < 0.1$), all the methods perform similarly. However, when the noise levels are more realistic ($\sigma_r \geq 0.1$ or $\sigma_o \geq 0.1$), our SDP method clearly outperforms all the previous methods. Furthermore, $\sigma_r$ and $\sigma_o$ affect the estimation results in a similar manner but at different scales, with $\sigma_o$ having a stronger impact on the general accuracy.
It should be noted that the values of $\varepsilon_\theta$ are mostly the same between methods since, unlike $t$, the value of $\theta$ is limited to $[-\pi, \pi]$, and thus, $e_\theta$ is bounded.

Under extremely large noises (see Fig. 7), our methods still provide the best results. The proposed QCQP and SDP methods have mostly identical results, which indicate that the relaxation is tight under the tested conditions. Nonetheless, one can argue that the reported translation error is too large for practical purposes. As such, we acknowledge that there is still room for further improvement, in particular dealing with very inaccurate odometry data.

3) Drift-Correction Capability: While the odometry from SLAM methods can provide accurate short-term ego-motions, they suffer from long-term drift due to the accumulated errors, which is particularly prevalent for large-scale missions [7], [12]. This drift can be modeled as Gaussian noises that act on the RFT $\hat{L}_2^1\hat{T}$ [30]. As such, an RTE estimator can monitor and correct the drift by continuously estimating $\hat{L}_2^1\hat{T}_k$ in a sliding window fashion. Let $\sigma_t$ and $\sigma_\theta$ be the noise on the translation ($x$, $y$, $z$ axes) and heading $\theta$ parts of $\hat{L}_2^1\hat{T}$, respectively. The larger $\sigma_t$ and $\sigma_\theta$, the more significant the drift [see Fig. 8(a)].

We simulate a scenario with two robots exploring a large environment [see Fig. 8(a)] in a period of 10 min. The host robot is equipped with a highly accurate localization system, such as RTK-GPS or LiDAR-based SLAM. The target robot is equipped with a VIO system that will drift away from the ground truth as time goes on. All methods must 1) estimate the RFT $\hat{L}_2^1\hat{T}_0$ and 2) track the changes of $\hat{L}_2^1\hat{T}_k$, using only recent measurements. The host robot performs a sinuous trajectory to ensure the observability of the data within the sliding window, while the target robot performs a simple trajectory to scan the area. The sliding window contains the latest 50 measurements, which are collected in 5 s. All the measurements are corrupted by $\sigma_o = 0.001$ m and $\sigma_r = 0.1$ m. We tested the NLS with and without any good initial guesses, where $\Theta_0$ is generated as described in Section VI-A.

The global position of the target robot in the world frame, i.e., $\hat{L}_2^1\hat{p}_k := \hat{t}_k + C_k \vec{e}_L \hat{p}_k$, is the final output that we are mostly interested in. Table II shows the RMSE of the translation error of $\hat{L}_2^1\hat{T}_k$. The smaller the RMSE, the better the system in performing both finding and monitoring $\hat{L}_2^1\hat{T}_k$. Fig. 8(b) demonstrates the translation errors in the case with the most drift. It is noticeable that the algebraic and linear methods actually worsen the position estimates. The reason is that with only data in a short sliding window, the situation is similar to the “hard” cases described in previous section (small $R_{max}$, large $d_0$). Hence, these methods could not obtain a good estimate for $\hat{L}_2^1\hat{T}_k$ during the whole operation. The other optimization-based methods show more consistent and improved results in cases with larger drifts. Overall, it is obvious that when no initial guesses are provided, our QCQP and SDP are the best and second best methods with a significant margin compared to previous approaches. When a good initial guess is provided, the NLS method can consistently surpass our SDP method, but still falls short of our QCQP method, which showcases the advantages of our solution.

C. Real-Life Experiments

Fig. 9 shows the hardware platforms in our real-life experiments. Each quadrotor is equipped with a Humatics P440 UWB,7 an UP2 minicomputer,8 an Intel Realsense T265 VIO sensor.9 The UWB antenna positions in the body frames are $\vec{a}_i^p = [0.01, 0.1, -0.1]^T$ and $\vec{b}_i^p = [0.02, 0.15, -0.15]^T$. The UWB data is generated at 37 Hz, while the VIO data arrive

| $\sigma_\theta$ (degree) | Algebra [31] | Linear [25] | NLS w/o $\Theta_0$ | SDP [27] | Proposed SDP | Proposed QCQP | NLS w/ $\Theta_0$ |
|-----------------------|-------------|-------------|-----------------|----------|--------------|--------------|----------------|
| $\sigma_t = 0.01$ m   |             |             |                 |          |              |              |                |
| 0.01                  | 84.36       | 94.35       | 6.81            | 4.86     | 2.35         | 0.51         | 0.55           |
| 0.05                  | 88.06       | 108.39      | 7.59            | 4.91     | 2.37         | 0.42         | 0.35           |
| 0.1                   | 87.86       | 105.11      | 13.96           | 5.02     | 2.40         | 0.58         | 0.71           |
| $\sigma_t = 0.05$ m   |             |             |                 |          |              |              |                |
| 0.01                  | 82.07       | 153.26      | 7.05            | 4.94     | 2.37         | 0.59         | 0.63           |
| 0.05                  | 83.14       | 160.39      | 7.50            | 4.88     | 2.36         | 0.53         | 0.46           |
| 0.1                   | 81.16       | 173.13      | 7.06            | 4.97     | 2.39         | 0.43         | 0.30           |
| $\sigma_t = 0.1$ m    |             |             |                 |          |              |              |                |
| 0.01                  | 84.86       | 166.32      | 9.03            | 4.73     | 2.33         | 0.53         | 0.56           |
| 0.05                  | 85.01       | 170.38      | 9.04            | 4.82     | 2.38         | 0.54         | 0.60           |
| 0.1                   | 100.53      | 199.29      | 10.15           | 4.92     | 2.40         | 0.61         | 0.70           |

The first and second best results are ranked for each row. “w/ $\Theta_0$” and “w/o $\Theta_0$” represent with and without good initial guesses, respectively.

Fig. 8. (a) Trajectories and UWB measurements. QCQP-R,2 refers to the global trajectory of the target robot as computed by our QCQP method, with $\sigma_t = \sigma_\theta = 0.1$. Note that only a subset of all $d_k$ is illustrated to improve clarity. (b) Translation errors of different methods with $\sigma_t = \sigma_\theta = 0.1$.

Fig. 9. Quadrotor platforms used in our experiments.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
NGUYEN AND XIE: RELATIVE TRANSFORMATION ESTIMATION BASED ON FUSION OF ODOMETRY AND UWB RANGING DATA

TABLE III

| ID  | RMSE of Translation (t) | RMSE of Heading (θ) | RMSE of Translation (t) | RMSE of Heading (θ) |
|-----|-------------------------|---------------------|-------------------------|---------------------|
|     | Algebra [31]            | Linear [25]         | NLS [30]                | SDP [27]            |
| 01  | t(m) 0.759             | 0.469               | 0.335                   | 0.258               |
|     | θ(rad) 0.057           | 0.011               | 0.003                   | 0.024               |
| 02  | t(m) 0.156             | 0.255               | 0.072                   | 0.102               |
|     | θ(rad) 0.570           | 0.165               | 0.130                   | 0.124               |
| 03  | t(m) 0.315             | 0.187               | 0.174                   | 0.124               |
|     | θ(rad) 0.031           | 0.014               | 0.020                   | 0.014               |
| 04  | t(m) 1.265             | 0.115               | 0.195                   | 0.184               |
|     | θ(rad) 0.211           | 0.132               | 0.138                   | 0.119               |

Each result is averaged over three runs. The first and second best results are ranked for each row.

Fig. 10. Trajectories in real-life flight tests.

Fig. 11. Estimation results in one run for flight 02. (a) Estimation error. (b) MSE against CRLB.

Fig. 12. Uncertainty of the estimates in two real flight tests. (a) Flight 02. (b) Flight 05.

Overall, our QCQP method surpasses the other methods in most cases and also provides consistent results, with our SDP method being a close second.

D. Uncertainty Estimation

As stated in Section V-D1, the uncertainty of the configuration is quantified by the condition number of the estimated FIM, \( \kappa(F) \), whereas the uncertainty of each parameter is measured by the standard error \( \sigma_{\hat{\theta}} \). Figs. 12 and 13 demonstrate these values in real-life experiments and simulations, respectively. All simulations are done with \( \sigma_r = 0.1, \sigma_o = 0.001, R_{\text{max}} = 1, \) and \( d_0 = 3 \). All the results are obtained from our QCQP method.

In typical observable situations [see Fig. 12(a) and 12(b)], the estimated uncertainties follow the trend of the actual errors: as more measurements are incorporated, the errors as well as the uncertainties reduce. In simulation, the robots’ movements cover all directions and the uncertainty on \( x, y, \) and \( z \) axes behave similarly. In real flights, the motion on the \( z \)-axis is much more limited than the other axes due to safety concerns and the platforms’ capability. As such, the rate of improvement for \( \sigma_z \) is noticeably much slower. Note that for multiple-unmanned-aerial-vehicle (UAV) scenarios, the experiments usually start with degenerate configurations: first, the two UAVs take off (i.e., the robots move in parallel in the \( z \)-axis); then, at least one of them will hover for a short period of time to establish communication or collect data (i.e., one robot is static). Hence, by monitoring \( \kappa(F) \), we can detect whether the configuration is degenerate. As shown in Fig. 12(a), the value of \( \kappa(F) \) stays high until \( t = 20 \) s, which corresponds to when \( R_2 \) starts moving. Afterward, a noticeable drop in \( \kappa(F) \) can be seen, which indicates the configuration is no longer unobservable.

On the other hand, Fig. 13(a)–(d) shows the cases where the configuration remains unobservable throughout the experiment. First, notice that the value of the condition number \( \kappa(F) \) is substantially larger and tends to only increase. Second, the particular parameters that are unobservable (shown in Fig. 3) will have considerably larger standard errors and do not improve over time, while the others behave similar to the observable cases. Hence, these configurations can be recognized as singular with the unobservable DoFs identified.

E. Computational Demands

Fig. 14(a) and 14(b) demonstrates the solver time in all the simulations of Fig. 5 and real-life experiments in Table III.
Fig. 13. Estimation errors (top row) and uncertainty metrics (bottom row) in simulations. We can see that: 1) the estimated uncertainty metrics are clearly much larger in unobservable [(a)–(d)] than observable cases (see Fig. 12); and 2) if the standard error \( \sigma_i \) or estimation error \( e_i \) do not improve over time, the parameter \( \Theta_i \) is likely to be unobservable in that configuration. (a) Parallel motion. (b) Planar motion. (c) Static target. (d) Static host.

Fig. 14. Comparison of solver time (ms). (a) Solver time of all simulations in Fig. 4. (b) Average solver time in real-life experiments in Table III.

respectively. Overall, most methods can run in real time. The linear method is consistently the fastest, closely matched by the algebraic method. Our QCQP method is the slowest in simulation. Since the QCQP method’s solver time often correlates with the hardness of the problem (takes longer with smaller \( R_{\text{max}}/d_0 \) or larger noise) and the simulation covers all cases from easy to hard, the variation is much larger than the other methods. In real-life experiments, our SDP method runs faster than our QCQP method with an almost 50% improvement, while the previous SDP method is the slowest and not real time. The reason might be that the QCQP solver library directly supports our problem formulation, while we needed to adapt the public SDP solver library for the formulation of [27]. Hence, the optimization of the library would be the main reason for the improved speed of our SDP method. However, as the scale of our real-life experiments is limited, these results are not as indicative as the simulation.

VII. CONCLUSION

In this article, we studied the four-DoF RTE problem using the local odometry and interrobot UWB range measurements. The theoretical analysis of the problem was put forth, including the CRLB, the FIM, and the determinant of the FIM. Based on these findings, insights for the geometric interpretation of information gains for each parameter, methods to detect singular configurations and measure the uncertainty of the estimates were provided. To solve the problem, optimization-based solutions were introduced, which consisted of a QCQP approach and the corresponding SDP relaxation. Our system outperformed previous methods in both simulations and real-life experiments, especially in challenging scenarios, and was more robust to large UWB noise. While both the proposed approaches can run in real time on minicomputers, the QCQP method generally provides the most accurate results but takes longer time than the SDP counterpart. Finding the full unobservable conditions, the optimal trajectory configuration using \( \det(F) \) as well as extending the system to the general case with \( N \) robots are interesting topics for future works.

APPENDIX A

After computing the Jacobian, we have

\[
G_i = [\partial_z f_i, \partial_y f_i, \partial_z f_i, \partial_y f_i] = \frac{1}{\alpha_i} 
\begin{bmatrix}
g_{f_i}^x & g_{f_i}^y & g_{f_i}^z & g_{f_i}^\theta
\end{bmatrix} \tag{52}
\]
where
\[
\begin{align*}
   d_i &= \| \mathbf{t} + \mathbf{C}_{a_2}^\theta \mathbf{p}_i - \mathbf{C}_{a_1}^\theta \mathbf{p}_i \| = \sqrt{(g_i^x)^2 + (g_i^y)^2 + (g_i^z)^2} \\
   g_i^x &= g_i^x(t^x, \theta) = t^x + a_i^x \cos \theta - a_i^y \sin \theta - \varphi_i^x \\
   g_i^y &= g_i^y(t^y, \theta) = t^y + a_i^x \sin \theta + a_i^y \cos \theta - \varphi_i^y \\
   g_i^z &= g_i^z(t^z, \theta) = t^z + a_i^x \cos \theta - a_i^y \sin \theta.
\end{align*}
\]

Notice that \(g_i^x, g_i^y, g_i^z\), respectively, correspond to the displacement in \(x, y,\) and \(z\) axes between the positions of the two robots at time \(t_i\), while the true distance measurement \(d_i\) would be the same as \(\| \mathbf{a}_2^\theta \mathbf{p}_i \|\). Overall, we can write
\[
[\partial_x f_i, \partial_y f_i, \partial_z f_i] = \frac{1}{d_i} \begin{bmatrix} g_i^x / d_i \\ g_i^y / d_i \\ g_i^z / d_i \end{bmatrix} = \frac{a_2^\theta \mathbf{p}_i}{\| a_2^\theta \mathbf{p}_i \|} = \mathbf{u}_i
\]

which leads to
\[
\Phi_i = \rho_i \sin \gamma_i. \tag{57}
\]

**APPENDIX B**

**A. 3-D RTE With a Common Heading Reference**

In this case, the state vector is \(\Theta := [t^x, t^y, t^z]^{T}\) and the Jacobian is reduced to \(G_i = [\partial_x f_i, \partial_y f_i, \partial_z f_i]\), and we still have the FIM as \(\mathbf{F} = \sigma_r^{-2}J^{T}J\), where the \(i\)th row of \(J\) is \(G_i\). Let \(S_i = \{1 \leq i < j < l \leq k\}\); the Cauchy–Binet formula gives
\[
\det(F) = \frac{1}{\sigma_r^2} \sum_{S_i} \left( \det \left( \begin{bmatrix} \mathbf{u}_i^T \\ \mathbf{u}_j^T \\ \mathbf{u}_l^T \end{bmatrix} \right) \right)^2
\]

\[
= \frac{1}{\sigma_r^2} \sum_{S_i} \left( \det(\mathbf{u}_i \mathbf{u}_j \mathbf{u}_l) \right)^2. \tag{58}
\]

Since \(\det([\mathbf{a} \mathbf{b} \mathbf{c}]) = (\mathbf{a} \times \mathbf{b}) \cdot \mathbf{c}\), we have
\[
\det(F) = \frac{1}{\sigma_r^2} \sum_{S_i} \left( \det(\mathbf{u}_i \mathbf{u}_j \mathbf{u}_l) \right)^2
\]

\[
= \frac{1}{\sigma_r^2} \sum_{S_i} \mathbf{u}_i^2 \mathbf{u}_j^2 \mathbf{u}_l^2 \sin^2 \alpha \sin^2 \beta
\]

\[
= \frac{1}{\sigma_r^2} \sum_{S_i} \sin^2 \alpha \sin^2 \beta \tag{59}
\]

where \(\alpha = \angle(\mathbf{u}_i, \mathbf{u}_j)\) and \(\beta = \frac{\pi}{2} - \angle(\mathbf{u}_i \times \mathbf{u}_j, \mathbf{u}_l)\).

**B. 2-D RTE Without a Common Heading Reference**

In this case, the state vector is \(\Theta := [t^x, t^y]^{T}\). To simplify the analysis, we still use the same 3-D representation for the local odometry vectors and the unit relative position vector \(\mathbf{u}_i\) in \(\mathbb{R}^3\) but with zero \(z\) elements. The target rotation matrix \(\mathbf{C} \in SO(3)\) is the same. The length of the projection of \(\mathbf{C}_{a_2}^\theta \mathbf{p}_i\) on the \(xy\) plane of \(\{\mathcal{L}_2\}\) is \(\rho_i = \| \mathbf{C}_{a_2}^\theta \mathbf{p}_i \|\). The range measurement model is
\[
\begin{align*}
   f_i &= \left\| \mathbf{t}^x + a_i^x \cos \theta - a_i^y \sin \theta - \varphi_i^x \right\| \\
   &\quad \left\| \mathbf{t}^y + a_i^x \sin \theta + a_i^y \cos \theta - \varphi_i^y \right\| \\
   &\quad \left\| \mathbf{t}^z + a_i^x \cos \theta - a_i^y \sin \theta \right\|
\end{align*}
\]

and the Jacobians are \(G_i = [\partial_x f_i, \partial_y f_i, \partial_z f_i] = \frac{1}{\sigma_i} [g_i^x, g_i^y, g_i^z]\), where
\[
\begin{align*}
   d_i &= \sqrt{(g_i^x)^2 + (g_i^y)^2} \\
   g_i^x &= t^x + a_i^x \cos \theta - a_i^y \sin \theta - \varphi_i^x \\
   g_i^y &= t^y + a_i^x \sin \theta + a_i^y \cos \theta - \varphi_i^y \\
   g_i^z &= g_i^z(t^z, \theta) = t^z + a_i^x \cos \theta - a_i^y \sin \theta.
\end{align*}
\]

Let \(\gamma_i = \frac{\pi}{2} - \angle(\mathbf{u}_i \times \mathbf{C}_{a_2}^\theta \mathbf{p}_i, \mathbf{u}_i)\) and \(\rho_i\) be the length of the projection of \(\mathbf{C}_{a_2}^\theta \mathbf{p}_i\) on the \(xy\) plane of \(\{\mathcal{L}_2\}\); then, we have
\[
\begin{align*}
   \sin \gamma_i &= \cos \angle(\mathbf{u}_i \times \mathbf{C}_{a_2}^\theta \mathbf{p}_i, \mathbf{u}_i) \\
   \rho_i &= \| \mathbf{C}_{a_2}^\theta \mathbf{p}_i \| \sin \angle(\mathbf{u}_i \times \mathbf{C}_{a_2}^\theta \mathbf{p}_i, \mathbf{u}_i).
\end{align*}
\]
\[
\frac{1}{\sigma_t^2} \sum_{S_2} \left( \sum_{i=1}^{3} (-1)^{i+3} \frac{\partial \phi_{f_1}}{\partial x_{f_p}} \cdot \frac{\partial \phi_{f_1}}{\partial y_{f_q}} \right)^2
\]

(62)

where \( S_2 = \{ij, jk, ki\} | 1 \leq j < k < i \leq k \), \( l = j, \), \( p, q \in \{j, k, i\} \), \( l, p < q \). From (61), we can write \( \partial \phi_{f_1} \) as

\[
\partial \phi_{f_1} = \frac{1}{d_l} \left[ g_l^{\theta_i} (-\sigma_l \sin \theta - \sigma_l \cos \theta) \right. \\
+ \left. g_l^{\sigma_i} (\sigma_l \cos \theta - \sigma_l \sin \theta) \right]
\]

(63)

where \( u_l = \left[ g_l^{\theta_i}, g_l^{\sigma_i}, 0 \right]^T \). Since vector \( C_{a_j}^2 p_l \) resides on the \( xy \) plane, the angle between \( u_x \) and \( C_{a_j}^2 p_l \) is always \( \pi/2 \), i.e., \( \angle (u_x, C_{a_j}^2 p_l) = \pi/2 \). Also, notice that \( \| C_{a_j}^2 p_l \| = \| f_l \| = \rho_l \). Let \( \gamma_l = \frac{\pi}{2} - \angle (u_x, C_{a_j}^2 p_l, u_l) \), \( \partial \phi_{f_1} \) in (63) can be simplified as

\[
\partial \phi_{f_1} = \| u_x \| \| C_{a_j}^2 p_l \| \| u_l \| \sin \angle (u_x, C_{a_j}^2 p_l) \sin \gamma_l
\]

(64)

\[= \rho_l \sin \gamma_l.\]

Next, we have

\[
\det\left( \begin{bmatrix} \frac{\partial f_p}{\partial x_{f_p}} & \frac{\partial f_q}{\partial y_{f_q}} \\ \frac{\partial f_q}{\partial x_{f_q}} & \frac{\partial f_p}{\partial y_{f_p}} \end{bmatrix} \right) = \det\left( \begin{bmatrix} \frac{\partial x_{f_p}}{\partial x_{f_p}} & \frac{\partial x_{f_q}}{\partial y_{f_q}} \\ \frac{\partial x_{f_q}}{\partial x_{f_q}} & \frac{\partial x_{f_p}}{\partial y_{f_p}} \end{bmatrix} \right)
\]

\[= \det\left( \begin{bmatrix} \frac{\partial x_{f_p}}{\partial x_{f_p}} & 0 \\ \frac{\partial x_{f_q}}{\partial x_{f_q}} & \frac{\partial x_{f_p}}{\partial y_{f_p}} \end{bmatrix} \right) \cdot \left( u_p \times u_q \right) \cdot u_{z}.
\]

(65)

Since both vectors \( u_p \) and \( u_q \) reside on the \( xy \) plane, their cross product will align with \( u_z \). The signed angle from \( u_p \) to \( u_q \) can be computed as

\[
\alpha_i = \arctan2 \left( [u_p \times u_q]_z, u_p \cdot u_q \right)
\]

(66)

where \([z]_z\) denotes the \( z \) element of the argument vector in \( \mathbb{R}^3 \). As such, \( u_p \times u_q = \| u_p \| \| u_q \| \sin \alpha_i u_z = \sin \alpha_i u_z \). Equation (65) can then be written as

\[
\det\left( \begin{bmatrix} \frac{\partial x_{f_p}}{\partial x_{f_p}} & \frac{\partial x_{f_q}}{\partial y_{f_q}} \\ \frac{\partial x_{f_q}}{\partial x_{f_q}} & \frac{\partial x_{f_p}}{\partial y_{f_p}} \end{bmatrix} \right) = \sin \alpha_i \| u_z \|^2 = \sin \alpha_i.
\]

(67)

Replacing (64) and (67) into (62), we have

\[
\det(F) = \frac{1}{\sigma_t^2} \sum_{S_2} \left( \sum_{i=1}^{3} (-1)^{i+1} \rho_i \sin \alpha_i \sin \gamma_i \right)^2.
\]

(68)

C. 2-D RTE With a Common Heading Reference

In this case, the state vector is \( \Theta := \left[ t^r, t^v \right]^T \). Built upon the notations and equations in Appendix B-A, the determinant of the FIM is

\[
\det(F) = \frac{1}{\sigma_t^2} \sum_{S_3} \left( \sum_{i=1}^{3} \rho_i \sin \alpha_i \sin \gamma_i \right)^2.
\]

(69)

where \( S_3 = \{(i, j) | 1 \leq i < j \leq k \} \) and \( \alpha = \angle (u_i, u_j) \). This completes the proof.

ACKNOWLEDGMENT

The authors would like to thank Dr. Cao Kun and Cao Muqing for the fruitful discussions regarding previous works.

REFERENCES

[1] J. P. Queralt et al., “Collaborative multi-robot search and rescue: Planning, coordination, perception, and active vision,” IEEE Trans. Autom., vol. 8, pp. 191617–191643, 2020.

[2] W. Shule, C. M. Almansa, J. P. Queralt, Z. Zou, and T. Westerlund, “UWB-based localization for multi-UAV systems and collaborative heterogeneous robot systems,” Procedia Comput. Sci., vol. 175, pp. 357–364, 2020.

[3] Y. Rizik, M. Awad, and E. W. Tuncel, “Cooperative heterogeneous multi-robot systems: A survey,” ACM Comput. Surv., vol. 52, no. 2, pp. 1–7, 2019.

[4] X. S. Zhou and S. L. Roumeliotis, “Determining 3-D relative transformations for any combination of range and bearing measurements,” IEEE Trans. Robot., vol. 29, no. 2, pp. 458–474, Apr. 2013.

[5] J. Knuth and P. Barthol, “Collaborative localization with heterogeneous inter-robot measurements by Riemannian optimization,” in Proc. IEEE Int. Conf. Robot. Autom., 2013, pp. 1534–1539.

[6] F. de Ponte Müller, “Survey on ranging sensors and cooperative techniques for relative positioning of vehicles,” Sensors, vol. 17, no. 2, Apr. 2017.

[7] S. Yuan, H. Wang, and L. Xie, “Survey on localization systems and algorithms for unmanned systems,” Unmanned Syst., vol. 9, no. 2, pp. 129–163, Feb. 2021.

[8] A. Cornejo and R. Nagpal, “Distributed range-based relative localization of robot swarms,” in Algorithmic Foundations of Robotics XI, New York, NY, USA: Springer, 2015, pp. 91–107.

[9] T.-M. Nguyen, Z. Qiu, T. H. Nguyen, M. Cao, and L. Xie, “Distance-based cooperative relative localization for leader-following control of MAVs,” IEEE Robot. Autom. Lett., vol. 4, no. 4, pp. 3641–3648, Oct. 2019.

[10] T.-M. Nguyen, A. H. Zaini, C. Wang, K. Guo, and L. Xie, “Robust target-relative localization with ultra-wideband ranging and communications,” in Proc. IEEE Int. Conf. Robot. Autom., 2018, pp. 2312–2319.

[11] X. Yu, Q. Li, J. P. Queralt, J. Heikkonen, and T. Westerlund, “Applications of UWB networks and positioning to autonomous robots and industrial systems,” in Proc. 10th Mediterranean Conf. Embedded Comput., 2021, pp. 1–6.

[12] T.-M. Nguyen, M. Cao, S. Yuan, Y. Lyu, T. H. Nguyen, and L. Xie, “VIRAL-Fusion: A visual-inertial-ranging-lidar sensor fusion approach,” IEEE Trans. Robot., vol. 38, no. 2, pp. 958–977, Apr. 2022.

[13] S. A. S. Mohamed, M. H. Haghbayan, T. Westerlund, J. Heikkonen, H. Tenhunen, and J. Plosila, “A survey on odometry for autonomous navigation systems,” IEEE Access, vol. 7, pp. 97466–97486, 2019.

[14] G. Huang, “Visual-inertial navigation: A concise review,” in Proc. Int. Conf. Robot. Autom., 2019, pp. 9572–9582.

[15] P. N. T. M. Nguyen, “Range-based adaptive navigation for autonomous micro aerial vehicles,” Ph.D. dissertation, Dept. Elect. Electron. Eng., Nanyang Technol. Univ., Singapore, 2020.

[16] Y. Cao and G. Beltrame, “VIR-SLAM: Visual, inertial, and ranging SLAM for single and multi-robot systems,” Auton. Robots, vol. 45, no. 6, pp. 905–917, 2021.

[17] T. H. Nguyen, T.-M. Nguyen, and L. Xie, “Tightly-coupled ultra-wideband-aided monocular visual SLAM with degenerate anchor configurations,” Auton. Robots, vol. 44, no. 8, pp. 1519–1534, 2020.

[18] H. Xu, L. Wang, Y. Zhang, K. Qiu, and S. Shen, “Decentralized visual-inertial-UWB fusion for relative state estimation of aerial swarm,” in Proc. IEEE Int. Conf. Robot. Autom., 2020, pp. 8776–8782.
NGUYEN AND XIE: RELATIVE TRANSFORMATION ESTIMATION BASED ON FUSION OF ODOMETRY AND UWB RANGING DATA

[19] S. Güler, M. Abdelkader, and J. S. Shamma, “Peer-to-peer relative localization of aerial robots with ultrawideband sensors,” IEEE Trans. Control Syst. Technol., vol. 29, no. 5, pp. 1981–1996, Sep. 2021.

[20] X. Yu, Q. Li, J. P. Queralt, J. Heikonen, and T. Westerlund, “Cooperative UWB-based localization for outdoors positioning and navigation of UAVs aided by ground robots,” in Proc. IEEE Int. Conf. Auton. Syst., 2021, pp. 1–5.

[21] M. Shalaby, C. C. Cossette, J. R. Forbes, and J. Le Ny, “Relative position estimation in multi-agent systems using attitude-coupled range measurements,” IEEE Robot. Autom. Lett., vol. 6, no. 3, pp. 4955–4961, Jul. 2021.

[22] T.-M. Nguyen, Z. Qiu, T. H. Nguyen, M. Cao, and L. Xie, “Persistently excited adaptive relative localization and time-varying formation of robot swarms,” IEEE Trans. Robot., vol. 36, no. 2, pp. 553–560, Apr. 2020.

[23] K. Guo, X. Li, and L. Xie, “Ultra-wideband and odometry-based cooperative relative localization with application to multi-UAV formation control,” IEEE Trans. Cybern., vol. 50, no. 6, pp. 2590–2603, Jun. 2020.

[24] J. Knuth and P. Baroosh, “Distributed collaborative 3D pose estimation of robots from heterogeneous relative measurements: An optimization on manifold approach,” Robotica, vol. 33, no. 7, pp. 1507–1535, 2014.

[25] F. M. Martel, J. Siderenko, C. Bodenstein, M. Arens, and U. Hagentobler, “Unique 4-DOF relative pose estimation with six distances for UWB/V-SLAM-based devices,” Sensors, vol. 19, no. 19, Art. no. 4366.

[26] M. Li, G. Liang, H. Luo, H. Qian, and T. L. Lam, “Robot-to-robot relative pose estimation based on semidefinite relaxation optimization,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., 2020, pp. 4491–4498.

[27] B. Jiang, B. D. O. Anderson, and H. Hnam, “3-D relative localization of mobile systems using distance-only measurements via semidefinite optimization,” IEEE Trans. Aerosp. Electron. Syst., vol. 56, no. 3, pp. 1900–1916, Jun. 2020.

[28] N. Trawny and S. I. Roumeliotis, “On the global optimum of planar, range-based robot-to-robot relative pose estimation,” in Proc. IEEE Int. Conf. Robot. Autom., 2010, pp. 3200–3206.

[29] A. Shariati, K. Mohra, and C. J. Taylor, “Recovering relative orientation and scale from visual odometry and ranging radio measurements,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., 2016, pp. 3627–3633.

[30] T. Zaegler, M. Karrer, P. Schmuck, and M. Chli, “Distributed formation estimation via pairwise distance measurements,” IEEE Robot. Autom. Lett., vol. 6, no. 2, pp. 3017–3024, Apr. 2021.

[31] N. Trawny, X. Zhou, K. Zhou, and S. Roumeliotis, “Interrobot transformations in 3-D,” IEEE Trans. Robot., vol. 26, no. 2, pp. 226–243, Apr. 2010.

[32] N. Funabiki, B. Morrell, J. Nash, and A.-akbar Agha-mohammadi, “Range-aided pose-graph-based SLAM: Applications of deployable ranging beacons for unknown environment exploration,” IEEE Robot. Autom. Lett., vol. 6, no. 1, pp. 48–55, Jan. 2021.

[33] A. Martinelli, “Vision and IMU data fusion: Closed-form solutions for attitude, speed, absolute scale, and bias determination,” Robotics, vol. 28, no. 1, pp. 44–60, Feb. 2012.

[34] T. H. Nguyen, T.-M. Nguyen, and L. Xie, “Range-focused fusion of camera-IMU-UWB for accurate and drift-reduced localization,” IEEE Robot. Autom. Lett., vol. 6, no. 2, pp. 1678–1685, Apr. 2021.

[35] H. C. So, Handbook of Position Location: Theory, Practice, and Advances, vol. 27. Hoboken, NJ, USA: Wiley, 2011.

[36] A. Ly, M. Marsman, J. Verhagen, R. F. Grasmann, and E.-J. Wagemakers, “A tutorial on fisher information,” J. Math. Psychol., vol. 80, pp. 40–55, 2017.

[37] K. Guo, Z. Qiu, W. Meng, L. Xie, and R. Teo, “Ultra-wideband based cooperative relative localization algorithm and experiments for multiple unmanned aerial vehicles in GPS denied environments,” Int. J. Micro Air Veh., vol. 9, no. 3, pp. 169–186, 2017.

[38] S. van der Helm, M. Coppola, K. N. McGuire, and G. C. H. E. de Croon, “On-board range-based relative localization for micro air vehicles in indoor leader-follower flight,” Auton. Robots, vol. 44, no. 3/4, pp. 415–441, 2019.

[39] Z.-Q. Luo, W.-K. Ma, A. M.-C. So, Y. Ye, and S. Zhang, “Semidefinite relaxation of quadratic optimization problems,” IEEE Signal Process. Mag., vol. 27, no. 3, pp. 20–34, May 2010.

[40] T. H. Nguyen, T.-M. Nguyen, and L. Xie, “Flexible and resource-efficient multi-robot collaborative visual-inertial-range localization,” IEEE Robot. Autom. Lett., vol. 7, no. 2, pp. 928–935, Apr. 2022.