# Robot-to-Robot Relative Pose Estimation from Range Measurements

Xun S. Zhou and Stergios I. Roumeliotis

***Abstract*—In this paper, we address the problem of determining the 2D relative pose of pairs of communicating robots from (i) robot-to-robot distance measurements and (ii) displacement estimates expressed in each robot's reference frame. Specifically, we prove that for nonsingular configurations the minimum number of distance measurements required for determining all 6 possible solutions for the 3 degrees-of-freedom robot-to-robot transformation is 3. Additionally, we show that given 4 distance measurements the maximum number of solutions is 4, while 5 distance measurements are sufficient for uniquely determining the robot-to-robot transformation. Furthermore, we present an efficient algorithm for computing the unique solution in closed form and describe an iterative least-squares process for improving its accuracy. Finally, we derive necessary and sufficient observability conditions based on Lie derivatives, and evaluate the performance of the proposed estimation algorithms both in simulation and experimentally.**

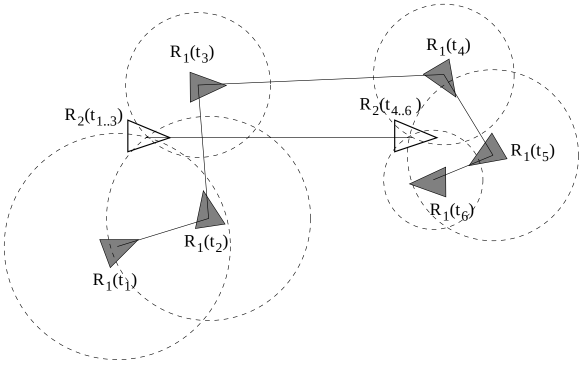


Fig. 1. A simplistic approach to determine an initial estimate for the robot-to-robot relative transformation using 6 distance measurements. The dark (light) triangles denote the location of robot $R_1$ ($R_2$), and $t_i, t_{i\dots j}, i, j \in \{1, \dots, 6\}$ indicate the time step(s) that a robot remains at a certain location.

## I. Introduction and Related Work

In order to solve distributed estimation problems such as cooperative localization, mapping, and tracking, robots must first determine their relative position and orientation (pose). This extrinsic calibration process is necessary for coordinating a robot team and registering measurements to the same frame of reference. Since the accuracy of the relative (robot-to-robot) transformation can significantly affect the quality of a sensor fusion task (e.g., tracking a target using observations from multiple sensors), it needs to be computed precisely. Mobile robots that move on a plane and use distance *and* bearing sensors (e.g., laser scanners or stereo cameras) can uniquely determine their relative pose by concurrently processing one distance and two relative bearing measurements recorded in a single location [1]. However, due to power and processing constraints, robots often have to rely on exteroceptive sensors that provide only range measurements. In these cases, computing the unknown robot-to-robot transformation requires developing algorithms for processing multiple distance measurements collected at numerous locations.

Due to their low cost and power consumption, range sensors have recently become widely used for solving localization problems in sensor networks [2] and distributed groups of mobile robots [3]. Range measurements can be obtained using Radio-Frequency (RF) or ultrasound signals. Many RF devices are cheap and already integrated into communication equipment; hence no additional cost is required for range sensing. One of the most common techniques for measuring range is based on the Received Signal Strength Indicator (RSSI). The main advantage of RSSI is that it is available in practically all receivers. An example of such a distance-measuring device is the RADAR system employed for wireless network localization in [2]. Alternatively, distances can be measured precisely by estimating the time-of-flight of an acoustic signal transmitted from one sensor or beacon to another [4], [5]. This type of system comprises a transmitter and a receiver. The transmitter sends out simultaneously a RF synchronization message and a distinct sound. The receiver then measures the time difference in the reception of the two signals and computes the distance using a known model for the speed of sound in the air.

Most current research on applications of range sensing has focused on designing algorithms that process distance measurements to determine only the *position* of each node in a *static* network of sensors [6], or the position and orientation of a mobile robot when *static* beacons are deployed within an area of interest [3]. In the case of networks of sensors, a variety of algorithms based on convex optimization [7] and Multi Dimensional Scaling (MDS) [8], have been employed to localize the sensor nodes. Additionally, distributed approaches that reduce the communication requirements and better balance the computational load among sensors have also received significant attention in the related literature (e.g., [9], [10]). In all these cases, the objective is to determine *only* the position of the sensor nodes with respect to anchor nodes that can globally localize via GPS measurements. Similarly, in the case of mobile robots the emphasis is on using distance measurements to localize robots with respect to *static* beacons [3], but not on computing the *relative pose* of the robots. Note, however, that

Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55455. Email: zhou|stergios@cs.umn.edu. This work was supported by the University of Minnesota (DTC) and the National Science Foundation (EIA-0324864, IIS-0643680, IIS-0811946).

determining the robot-to-robot transformation is a prerequisite for efficiently coordinating the motion of teams of robots [11], localizing cooperatively [12], [13], and in general, expressing measurements in a common frame of reference [1].

The problem we are interested in is that of directly computing the 3 degrees-of-freedom (d.o.f.) robot-to-robot transformation from *distance measurements* and *displacement estimates*, the latter expressed in the reference frame of the corresponding robot. Specifically, we consider pairs[1] of communicating robots equipped with odometric sensors for tracking their motion and range sensors for measuring the distance to each other. In this case, if no prior information about their relative pose is available, a human operator will need to manually measure the transformation between the two robots before they can be deployed to perform their assigned task (e.g., cooperative localization [13]). However, this tedious process limits the accuracy of the robot-to-robot transformation and increases the cost of deploying large teams of robots due to the time and effort required.

A straightforward approach to automating this *extrinsic calibration* process is for the robots to move randomly, collect distance measurements, and then compute their relative transformation by employing nonlinear least squares. However, any iterative process applied to minimize the nonlinear, in the unknown variables, cost function relies on the existence of an *accurate initial estimate* in order to converge to the global minimum. Additionally, since the necessary number of range measurements is not known a priori, a conservative strategy would require the robots to spend excessive time and energy measuring their distance numerous times at different locations. Instead it would be beneficial for the robots to follow a two-step process: (i) Employ a non-iterative algorithm to process the minimum number of distance measurements required to compute an initial estimate of their relative transformation, and (ii) Apply nonlinear least squares to iteratively refine this initial estimate using additional range measurements. This second step can be repeated until the user-specified level of accuracy is reached.

A simplistic method to compute an initial estimate for the 3 d.o.f. transformation in closed form would require the robots to follow a sequence of *coordinated motions* and measure distances to each other at certain locations and time instants. Specifically, as shown in Fig. 1, if robot $R_2$ remains static while $R_1$ measures its distance to $R_2$ at 3 different locations (time instants $t_1$, $t_2$, and $t_3$), the position of $R_2$ with respect to $R_1$ can be uniquely determined. In order to also compute their relative orientation, robot $R_2$ will need to move to a new location and remain static again until robot $R_1$ records another 3 distance measurements (time instants $t_4$, $t_5$, and $t_6$) and triangulates the new relative position of robot $R_2$. Using these 2 inferred relative position measurements and knowing the direction of motion of $R_2$ (computed from its own odometry), the relative orientation between the 2 robots can be uniquely determined. The main drawback of this approach is that it requires tight coordination between the robots for performing the sequence of necessary motions and recording the distance measurements at the appropriate locations. Additionally, this initial calibration phase delays the onset of the actual robot task which can be detrimental in time-critical situations involving large robot teams.

In this paper, we address this problem by deriving non-iterative algorithms for computing an initial estimate of the 3 d.o.f. robot-to-robot transformation without restricting their motion [14]. Specifically, we prove that when 2 robots move randomly and collect 3 distance measurements at different locations, the maximum number of solutions, generically, is 6 (cf. Lemma 1, Section II). When 4 range measurements are available, we show that, generically, there exist no more than 4 solutions (cf. Lemma 2, Section III). Furthermore, in Section IV (cf. Lemma 3) we prove that for nonsingular configurations the relative pose of the robots can be uniquely determined given 5 distance measurements (instead of 6 based on the simplistic method outlined in Fig. 1). Efficient algorithms for computing *all* solutions for the cases described above are presented. Additionally, we provide a novel linear algorithm for determining the unique solution (i.e., when 5 range measurements are available) that minimizes the numerical error in the computed transformation (cf. Section IV-B). In Section V, we describe the nonlinear least-squares algorithm that uses all range measurements available to iteratively refine the initial estimate for the unknown robot-to-robot transformation. In Section VI, we analyze the system observability and provide necessary and sufficient conditions for observability based on the control inputs applied to the two robots. In Section VII, we present simulation and experimental results that verify the validity of our theoretical analysis. Concluding remarks and future research directions are provided in Section VIII.

## II. Determining the Relative Pose from 3 Distance Measurements: At Most 6 Solutions

In order to improve the clarity of presentation, we first introduce the notation used throughout this paper:

| | |
|---|---|
| ${}^{i}\mathbf{p}_j$ | Position of frame $\{j\}$, expressed in frame $\{i\}$. |
| ${}^{i}\phi_j$ | Angle between the $x$ axes of frames $\{i\}$ and $\{j\}$ (i.e., their relative orientation). |
| ${}^{i}_{j}\mathbf{C}$ | Rotational matrix that projects vectors expressed in frame $\{j\}$ to frame $\{i\}$. |
| $d_{ij}$ | Distance between the origins of frames $\{i\}$ and $\{j\}$. |
| $s\alpha$ | Short for $\sin(\alpha)$. |
| $c\alpha$ | Short for $\cos(\alpha)$. |

Consider two robots $R_1$ and $R_2$ whose initial poses (position and orientation) are indicated by the frames of reference $\{1\}$ and $\{2\}$ respectively (cf. Fig. 2). The two robots move randomly through a sequence of odd poses $\{1\}, \{3\}, \ldots, \{2n-1\}$ for $R_1$, and even poses $\{2\}, \{4\}, \ldots, \{2n\}$ for $R_2$ and measure their distance $d_{ij}$, $i \in \{1, \ldots, 2n-1\}$, $j \in \{2, \ldots, 2n\}$ at each of these locations. Without loss of generality, we assume that only one of the robots records range measurements at each location. If both robots measure the same distance, the two measurements can be combined to provide a more accurate estimate of their distance.

Additionally, the robots are equipped with odometric sensors for estimating their poses with respect to their initial

[1] The extension to the problem of multiple robot teams is straightforward once a solution to the pair-wise problem is determined.

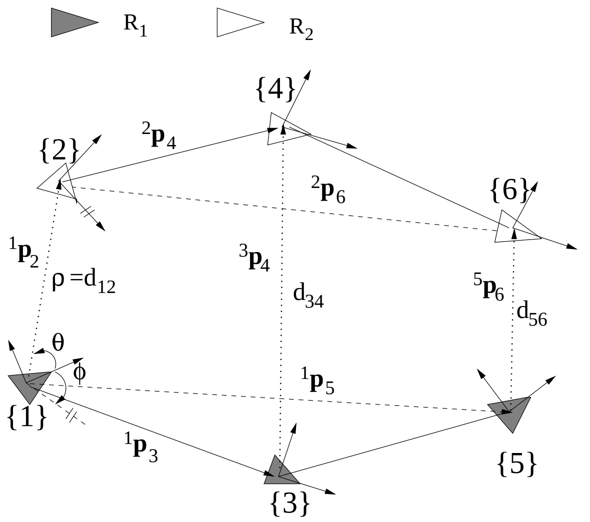


Fig. 2. The trajectories of robots $R_1$ and $R_2$. The odd (even) numbered frames of reference depict the consecutive poses of robot $R_1$ ($R_2$). $d_{ij}$, $i \in \{1,\dots,2n-1\}$, $j \in \{2,\dots,2n\}$ denotes the distance between the two robots when aligned to frames $\{i\}$ and $\{j\}$ respectively.

frames of reference. That is, robot $R_1$ estimates the position vectors ${}^1\mathbf{p}_3,\dots,{}^1\mathbf{p}_{2n-1}$, and the orientation angles ${}^1\phi_3,\dots,{}^1\phi_{2n-1}$ necessary for determining the rotational matrices ${}^1_3\mathbf{C},\dots,{}^1_{2n-1}\mathbf{C}$. Similarly, the quantities ${}^2\mathbf{p}_4,\dots,{}^2\mathbf{p}_{2n}$ and ${}^2\phi_4,\dots,{}^2\phi_{2n}$ (and hence ${}^2_4\mathbf{C},\dots,{}^2_{2n}\mathbf{C}$) are estimated by robot $R_2$ from its own odometry.[2] At this point, we should note that the particular kinematic model used by each robot and the motion measurements available to it, do not affect the solution of the relative pose problem. Actually, no motion measurements need to be exchanged between the robots. Instead, only the resulting position displacement estimates (i.e., ${}^1\mathbf{p}_{2\nu-1}$ and ${}^2\mathbf{p}_{2\nu}$, $\nu = 2,\dots,n$) must be communicated.

Our goal is to use the odometry-based estimates and the $n$ distance measurements to determine the *maximum number of solutions* for the initial 3 d.o.f. robot-to-robot transformation, i.e., their relative position ${}^1\mathbf{p}_2$ and orientation ${}^1\phi_2 = \phi$, or equivalently (in polar coordinates) $\theta$ and $\phi$, with

$$ {}^1\mathbf{p}_2 = \rho \begin{bmatrix} c\theta \\ s\theta \end{bmatrix} \quad , \quad {}^1_2\mathbf{C} = \begin{bmatrix} c\phi & -s\phi \\ s\phi & c\phi \end{bmatrix} \tag{1} $$

Note that $\rho = d_{12}$ is measured and considered known hereafter.

We first address the case when $n = 3$ distance measurements ($d_{12}$, $d_{34}$, and $d_{56}$) are available. We proceed by substituting the geometric relations for the position vectors ${}^3\mathbf{p}_4$, ${}^5\mathbf{p}_6$ (cf. Fig. 2)

$$ {}^3\mathbf{p}_4 = {}^1_3\mathbf{C}^T({}^1\mathbf{p}_2 + {}^1_2\mathbf{C}\,{}^2\mathbf{p}_4 - {}^1\mathbf{p}_3) \tag{2} $$

$$ {}^5\mathbf{p}_6 = {}^1_5\mathbf{C}^T({}^1\mathbf{p}_2 + {}^1_2\mathbf{C}\,{}^2\mathbf{p}_6 - {}^1\mathbf{p}_5) \tag{3} $$

in the following expressions for the distance measurements $d_{34}$ and $d_{56}$, respectively:

$$ d_{34}^2 = {}^3\mathbf{p}_4^T\,{}^3\mathbf{p}_4 \quad , \quad d_{56}^2 = {}^5\mathbf{p}_6^T\,{}^5\mathbf{p}_6 \tag{4} $$

After rearranging terms and substituting $\rho^2 = d_{12}^2$ for ${}^1\mathbf{p}_2^T\,{}^1\mathbf{p}_2$, these can be written as:

$$ ({}^1\mathbf{p}_2 - {}^1\mathbf{p}_3)^T\,{}^1_2\mathbf{C}\,{}^2\mathbf{p}_4 - {}^1\mathbf{p}_2^T\,{}^1\mathbf{p}_3 = a_0 \tag{5} $$

$$ ({}^1\mathbf{p}_2 - {}^1\mathbf{p}_5)^T\,{}^1_2\mathbf{C}\,{}^2\mathbf{p}_6 - {}^1\mathbf{p}_2^T\,{}^1\mathbf{p}_5 = b_0 \tag{6} $$

where

$$ a_0 = 0.5(d_{34}^2 - \rho^2 - {}^2\mathbf{p}_4^T\,{}^2\mathbf{p}_4 - {}^1\mathbf{p}_3^T\,{}^1\mathbf{p}_3) $$

$$ b_0 = 0.5(d_{56}^2 - \rho^2 - {}^2\mathbf{p}_6^T\,{}^2\mathbf{p}_6 - {}^1\mathbf{p}_5^T\,{}^1\mathbf{p}_5) $$

Note that $a_0$ and $b_0$ on the right-hand side of (5) and (6) are known (measured or estimated), while the unknown variables $\theta$ and $\phi$, embedded in ${}^1\mathbf{p}_2$ and ${}^1_2\mathbf{C}$ (cf. (1)), only appear on the left-hand side expressions.

Equations (5) and (6) form a system of 2 nonlinear equations in the 2 unknowns $\theta$ and $\phi$. Applying standard numerical techniques, such as Newton-Raphson [15], for solving this system has three major drawbacks: (i) may take a large number of iterations, (ii) requires an accurate initial estimate, and (iii) provides no guarantee of finding all solutions. The first drawback is that iterative processes often require a large number of steps before converging to a solution. The second drawback is that in order for the algorithm to converge to the correct answer, *initial estimates* close to the true values of the unknown variables need to be specified. In practice, however, no such information is available; the only prior knowledge we have for $\theta$ and $\phi$ is that each lies within the interval $[0,\ 2\pi)$. Finally, if for example only $n = 3$ distance measurements are available, the total number of solutions that needs to be determined is 6 (cf. Lemma 1). To compute all possible roots, numerous initial estimates for the unknowns $\theta$ and $\phi$ must be drawn from the 2-dimensional region $[0,\ 2\pi) \times [0,\ 2\pi)$. Such procedure requires a large number of initializations of the iterative process while provides no guarantees that all 6 solutions will be computed.[3]

Instead we hereafter seek to find all possible solutions algebraically. Notice that (5) and (6) can be transformed to polynomial equations by treating $c\phi$, $s\phi$, $c\theta$, and $s\theta$ as 4 unknown variables. Together with the 2 trigonometric constraints $s\phi^2 + c\phi^2 = 1$ and $s\theta^2 + c\theta^2 = 1$, these form a system of 4 polynomial equations, namely $f_1^*$, $f_2^*$, $f_3^*$, and $f_4^*$, in 4 unknowns. In general, none of these polynomials lies in the ideal generated by the remaining ones (cf. [16] for the definitions of ideals and varieties), and thus the square system considered has a finite number of solutions. However, there exist degenerate cases where the system has infinite solutions. For example, when the two robots move on parallel straight lines with equal linear velocities, the bearing angle $\theta$ can take any value within the interval $[0,\ 2\pi)$. Such a degenerate case occurs because $f_4^*$ belongs to the ideal $< f_1^*,\ f_2^*,\ f_3^* >$, i.e., $< f_1^*,\ f_2^*,\ f_3^*,\ f_4^* >=< f_1^*,\ f_2^*,\ f_3^* >$. Based on Proposition 4 of Ch. 1, §4 in [16], if $< f_1^*,\ f_2^*,\ f_3^*,\ f_4^* >=< f_1^*,\ f_2^*,\ f_3^* >$, then the varieties of these two ideals are equal,

[2] As it will become evident, only the position vectors are required for estimating the unknown robot-to-robot transformation. The orientation angles at intermediate steps are used for expressing the next position vector in the original frame of reference of each robot.

[3] To further stress this point, consider the case where after numerous initializations of the Newton-Raphson algorithm, only 4 real roots have been determined. In this situation there is no clear way to unequivocally declare the remaining 2 roots complex and terminate the search, or continue the iterations with new initial values in search of the remaining roots.

i.e., $\mathbf{V}(f_1^*,\ f_2^*,\ f_3^*,\ f_4^*) = \mathbf{V}(f_1^*,\ f_2^*,\ f_3^*)$. Equivalently, the 4 unknowns are only constrained by 3 equations, and thus infinite solutions exist. Note though that when the two robots move randomly while satisfying the conditions of Lemma 4 (cf. Section VI-C), the corresponding continuous-time system is locally weakly observable, and hence within an open neighborhood around a solution of the robot-to-robot transformation, there exists no other point indistinguishable from it (cf. Definition 4 in Section VI-A). That is, generically, the discrete-time system will have a finite number of solutions and the variety $\mathbf{V}(f_1^*,\ f_2^*,\ f_3^*,\ f_4^*)$ has dimension zero. We hereafter concentrate on nonsingular configurations and prove the following lemma:

*Lemma 1:* Given 3 distance measurements between two robots at 3 different locations, the maximum number of solutions for the 3 d.o.f. robot-to-robot transformation, generically, is 6.

*Proof:* The following derivation is based on an elimination process for removing the quantities $c\theta$, $s\phi$, $c\phi$ from the expressions in (5) and (6), which results in a 6th order polynomial in the unknown variable $y := s\theta$. The key idea behind this approach is similar to Gaussian Elimination for linear systems of equations. Due to space limitations only the main steps of this process are shown while reassignment of variables is used to preserve the clarity of presentation.

By substituting the displacement estimates (known from odometry) for the two robots:

$$ {}^1\mathbf{p}_3 = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}, \quad {}^2\mathbf{p}_4 = \begin{bmatrix} a_3 \\ a_4 \end{bmatrix}, \quad {}^1\mathbf{p}_5 = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \quad {}^2\mathbf{p}_6 = \begin{bmatrix} b_3 \\ b_4 \end{bmatrix} $$

in (5) and (6), and defining as

$$ \begin{aligned} a_5 &:= a_2 a_4 + a_1 a_3 \\ a_6 &:= a_2 a_3 - a_1 a_4 \\ b_5 &:= b_2 b_4 + b_1 b_3 \\ b_6 &:= b_2 b_3 - b_1 b_4 \end{aligned} $$

we have:

$$ \underbrace{(\rho a_3 c\theta + \rho a_4 s\theta - a_5)}_{u_1} c\phi + \underbrace{(\rho a_3 s\theta - \rho a_4 c\theta - a_6)}_{v_1} s\phi = \underbrace{a_0 + \rho(a_1 c\theta + a_2 s\theta)}_{w_1} \quad (7) $$

$$ \underbrace{(\rho b_3 c\theta + \rho b_4 s\theta - b_5)}_{u_2} c\phi + \underbrace{(\rho b_3 s\theta - \rho b_4 c\theta - b_6)}_{v_2} s\phi = \underbrace{b_0 + \rho(b_1 c\theta + b_2 s\theta)}_{w_2} \quad (8) $$

Equations (7) and (8) can be written in a matrix form as:

$$ \begin{bmatrix} u_1 & v_1 \\ u_2 & v_2 \end{bmatrix} \begin{bmatrix} c\phi \\ s\phi \end{bmatrix} = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \quad (9) $$

Note that (9), is linear in the unknowns $c\phi$ and $s\phi$. Solving for these two variables we have:

$$ \begin{bmatrix} c\phi \\ s\phi \end{bmatrix} = \frac{1}{\det} \begin{bmatrix} v_2 w_1 - v_1 w_2 \\ u_1 w_2 - u_2 w_1 \end{bmatrix} \quad (10) $$

where, $\det = u_1 v_2 - u_2 v_1$. Substituting the above expressions for $c\phi$ and $s\phi$ in the trigonometric constraint $s\phi^2 + c\phi^2 = 1$ results in a single equation in the variables $c\theta$ and $s\theta$

$$ \begin{aligned} &(u_1 v_2 - u_2 v_1)^2 = (v_2 w_1 - v_1 w_2)^2 + (u_1 w_2 - u_2 w_1)^2 \Rightarrow \\ &(u_1 v_2 - u_2 v_1)^2 = \\ &(v_2^2 + u_2^2) w_1^2 + (v_1^2 + u_1^2) w_2^2 - 2(v_1 v_2 + u_1 u_2) w_1 w_2 \quad (11) \end{aligned} $$

As detailed in [17], the terms $v_2^2 + u_2^2$, $v_1^2 + u_1^2$, $v_1 v_2 + u_1 u_2$, and $u_1 v_2 - u_2 v_1$ are all *linear* in $c\theta$ and $s\theta$, while $w_1^2$, $w_2^2$, $w_1 w_2$ are all *quadratic* in the same quantities. Hence (11) is a 3rd order polynomial in $x := c\theta$ and $y := s\theta$, and can be written in the following simpler form:

$$ \begin{aligned} f_1 = &m_9 x^3 + m_8 x^2 y + m_7 x y^2 + m_6 x^2 + m_5 x y + m_4 x + \\ &m_3 y^3 + m_2 y^2 + m_1 y + m_0 = 0 \quad (12) \end{aligned} $$

where the constants $m_0, \ldots, m_9$ are functions of known quantities [17]. The final step in the elimination process is to invoke the trigonometric constraint

$$ f_2 = x^2 + y^2 - 1 = 0 \quad (13) $$

and eliminate $x$ from (12) by using the Sylvester Resultant [18]. Specifically, by multiplying (12) with $x$, and (13) with $x$ and $x^2$, and rewriting the resulting equations in a matrix form, we have:

$$ \begin{bmatrix} s_3 & s_2 & s_1 & s_0 & 0 \\ 0 & s_3 & s_2 & s_1 & s_0 \\ 1 & 0 & y^2-1 & 0 & 0 \\ 0 & 1 & 0 & y^2-1 & 0 \\ 0 & 0 & 1 & 0 & y^2-1 \end{bmatrix} \begin{bmatrix} x^4 \\ x^3 \\ x^2 \\ x \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (14) $$

with

$$ \begin{aligned} s_0 &:= m_3 y^3 + m_2 y^2 + m_1 y + m_0 \\ s_1 &:= m_7 y^2 + m_5 y + m_4 \\ s_2 &:= m_8 y + m_6 \quad , \quad s_3 := m_9 \end{aligned} $$

For the polynomials in (12) and (13) to have common roots, the determinant of the 6 × 6 Sylvester matrix on the left-hand side of (14) must be equal to zero. As shown in [17], the determinant is a 6th order polynomial in the single variable $y$:

$$ g_2(y) = \xi_6 y^6 + \xi_5 y^5 + \xi_4 y^4 + \xi_3 y^3 + \xi_2 y^2 + \xi_1 y + \xi_0 \quad (15) $$

where the constants $\xi_0, \ldots, \xi_6$ are functions of the known quantities $m_0, \ldots, m_9$. Therefore, the maximum number of solutions for $y$, including complex roots, is 6.

To prove our claim that there exist at most 6 solutions for $\theta$, we also need to show that for every solution of $y := c\theta$ (cf. (15)), only one solution for $x := s\theta$ can be found (cf. (12)). We present two possible ways for proving this:

By directly computing the Groebner basis [16] for this system of equations, it is easy to show that one base, $g_2$, is exactly the same as the polynomial in (15), while $g_1$ has the form:

$$ g_1(x, y) = x + k_5 y^5 + k_4 y^4 + k_3 y^3 + k_2 y^2 + k_1 y + k_0 $$

where the constants $k_0, \ldots, k_5$ are functions of known quantities [17]. Therefore, for every root $y_i^*$, $i = 1, \ldots, 6$ of

$g_2(y) = 0$, there exists *only one* solution $x_i^*$ of $g_1(x, y_i^*) = 0$ corresponding to it.

Alternatively, we can draw the same conclusion without explicitly computing the Groebner basis. Instead, we only need to show that the leading term of $g_1$, LT($g_1$), is linear in $x$. This can be easily seen by using the definition of a Groebner basis [16]: A set $\{g_1, \ldots, g_s\} \subset I$ is a Groebner basis of an ideal $I$ if the leading term of *any* element of $I$ is divisible by one of the LT($g_i$). Specifically, we can construct one element $f_3$ of the ideal $I =< f_1, f_2 >$ by setting

$$\begin{aligned} f_3 &= f_1 - (m_9 x + m_8 y + m_6) f_2 \\ &= (m_7 - m_9) x y^2 + \text{(lower order terms)} \end{aligned}$$

Since the leading term LT($f_3$) $= (m_7 - m_9)xy^2$ must be divisible by LT($g_1$), the degree of $x$ in LT($g_1$) has to be 1, or equivalently, $g_1$ is linear in $x$. Hence, the total number of distinct solutions for $(x,\ y)$ remains 6.

Finally, $\phi$ is uniquely determined by back-substitution of $x = c\theta$ and $y = s\theta$ in (10). ■

Although in general there exist 6, possibly complex, solutions for (15), the total number of *real roots* will depend on the robots' trajectories and cannot be determined a priori. A scenario with 6 real solutions is shown in Fig. 8.

*Corollary 1:* Given 3 distance measurements between two robots at 3 different locations, generically, there exist 2, 4, or 6 solutions for the 3 d.o.f. robot-to-robot transformation.

This is evident if one considers that $g_2(y)$ (cf. (15)) is a 6th order polynomial with real coefficients, and complex roots appear in conjugate pairs.

### A. Computing all possible solutions

There exist many methods to compute the roots of a single-variable polynomial. Our approach relies on the eigendecomposition of the 6×6 *companion matrix* [19] for (15):

$$\begin{bmatrix} 0 & & & & -\xi_0/\xi_6 \\ 1 & 0 & & & -\xi_1/\xi_6 \\ & 1 & 0 & & -\xi_2/\xi_6 \\ & & \ddots & & \vdots \\ & & & 1 & -\xi_5/\xi_6 \end{bmatrix}$$

While this method will determine all 6 roots of the polynomial (eigenvalues of the companion matrix), only the real ones are of practical interest since they have a geometric interpretation. Once $y$ is known, $x$ is determined by computing the null space of the matrix in (14).

## III. Determining the Relative Pose from 4 Distance Measurements: At Most 4 Solutions

Consider now the case where the robots $R_1$ and $R_2$ continue their paths shown in Fig. 2 and move to the new poses $\{7\}$ and $\{8\}$, respectively, where they record an additional distance measurement $d_{78}$. We will prove the following:

*Lemma 2:* Given 4 distance measurements between two robots at 4 different locations, the maximum number of solutions for the 3 d.o.f. robot-to-robot transformation, generically, is 4.

*Proof:* We proceed in a similar manner as for the case of 3 distance measurements. Specifically, the new position estimates for the two robots at the locations where they record their 4th distance measurement

$${}^1\mathbf{p}_7 = \begin{bmatrix} e_1 \\ e_2 \end{bmatrix}, \quad {}^2\mathbf{p}_8 = \begin{bmatrix} e_3 \\ e_4 \end{bmatrix}$$

are related through the following geometric constraint (analogous to (2)):

$${}^7\mathbf{p}_8 = {}^1_7\mathbf{C}^T({}^1\mathbf{p}_2 + {}^1_2\mathbf{C}\,{}^2\mathbf{p}_8 - {}^1\mathbf{p}_7) \tag{16}$$

Substituting in the expression for the new distance measurement, $d_{78}^2 = {}^7\mathbf{p}_8^T\,{}^7\mathbf{p}_8$, results in the following equation:

$$({}^1\mathbf{p}_2 - {}^1\mathbf{p}_7)^T\,{}^1_2\mathbf{C}\,{}^2\mathbf{p}_8 - {}^1\mathbf{p}_2^T\,{}^1\mathbf{p}_7 = e_0 \tag{17}$$

where

$$e_0 = 0.5(d_{78}^2 - \rho^2 - {}^2\mathbf{p}_8^T\,{}^2\mathbf{p}_8 - {}^1\mathbf{p}_7^T\,{}^1\mathbf{p}_7)$$

is known (i.e., it is computed based on measured and estimated quantities). Following the same algebraic process as in the previous section, (17) can be written as:

$$\underbrace{(\rho e_3 c\theta + \rho e_4 s\theta - e_5)}_{u_3} c\phi + \underbrace{(\rho e_3 s\theta - \rho e_4 c\theta - e_6)}_{v_3} s\phi = \underbrace{e_0 + \rho(e_1 c\theta + e_2 s\theta)}_{w_3} \tag{18}$$

where $e_5$ and $e_6$ are defined as before. Rewriting (7), (8), and (18) in a matrix form, we have:

$$\begin{bmatrix} u_1 & v_1 & -w_1 \\ u_2 & v_2 & -w_2 \\ u_3 & v_3 & -w_3 \end{bmatrix} \begin{bmatrix} c\phi \\ s\phi \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{19}$$

where the $u_i$'s, $v_i$'s, and $w_i$'s, $i = 1, 2, 3$, are functions of $s\theta$, $c\theta$, and known (measured or estimated) quantities. For the above system to have nonzero solutions, the determinant of the coefficient matrix must vanish, i.e.,

$$(u_1 v_2 - u_2 v_1) w_3 + (v_1 u_3 - v_3 u_1) w_2 + (u_2 v_3 - u_3 v_2) w_1 = 0$$

Note that the terms $u_1 v_2 - u_2 v_1$, $v_1 u_3 - v_3 u_1$, and $u_2 v_3 - u_3 v_2$ are again all *linear* in $x := c\theta$ and $y := s\theta$ and so are $w_1$, $w_2$, and $w_3$, which make the above polynomial *quadratic* in $x$ and $y$ [17]. Following the same elimination procedure as in Section II, we arrive at a 4th order polynomial in $y$

$$n_4 y^4 + n_3 y^3 + n_2 y^2 + n_1 y + n_0 = 0 \tag{20}$$

where $n_0, \ldots, n_4$ are known constants [17]. In this case, there exist at most 4 solutions for $y$ all of which can be found in closed form. Once $y$ is determined, back-substitution allows us to find $x$. Finally, $s\phi$ and $c\phi$ are retrieved by computing the null-space vector of the coefficient matrix in (19). ■

In this lemma, we have shown that the maximum number of possible solutions is 4. We should note, however, that in most cases in practice there exists only one real solution for the robot-to-robot transformation. Specifically, even if all 4 roots of (20) are reals in the interval $[-1,\ 1]$ (and so will be the corresponding values for $x$), back-substitution in (19) will not always give a real value for $\phi$. This is because the first two elements of the null-space vector of the coefficient matrix in (19) must also satisfy the trigonometric constraint $s\phi^2 + c\phi^2 = 1$. Actually, this is a general property of over-determined nonlinear systems, which we hereafter describe in the context of the problem at hand.

*Corollary 2:* Given 4 distance measurements between two robots at 4 different locations, with probability 1 (i.e., almost surely) there exists a unique solution for the 3 d.o.f. robot-to-robot transformation.

To verify this, consider the case where after processing the first 3 distance measurements, 6 solutions are found, denoted as $\mathbf{x}_i = \begin{bmatrix} {}^1\mathbf{p}_2^T & {}^1\phi_2 \end{bmatrix}^T$, $i = 1, \ldots, 6$. Without loss of generality, we first assume that $\mathbf{x}_1$ corresponds to the true robot-to-robot transformation. When the robots move to their new positions, denoted by frames $\{7\}$ and $\{8\}$ respectively, to record their 4th distance measurement, $d_{78}$, then $\mathbf{x}_1$ should also satisfy (17):

$$({}^1\mathbf{p}_2 - {}^1\mathbf{p}_7)^T\, {}^1_2\mathbf{C}\, {}^2\mathbf{p}_8 - {}^1\mathbf{p}_2^T\, {}^1\mathbf{p}_7 - e_0 = 0 \Leftrightarrow h(\mathbf{x}_1, \boldsymbol{\vartheta}) = 0 \quad (21)$$

where $\boldsymbol{\vartheta} = \begin{bmatrix} {}^1\mathbf{p}_7^T & {}^2\mathbf{p}_8^T & d_{78} \end{bmatrix}^T$. Given $\mathbf{x}_1$, we denote as $\mathcal{V}_1$ the set of all values of $\boldsymbol{\vartheta}$ that satisfy (21). Note that $\mathcal{V}_1$ is a 4 dimensional variety.

If we now assume that there exists a 2nd solution, for example $\mathbf{x}_2$, then it should also satisfy (17), i.e.,

$$h(\mathbf{x}_2, \boldsymbol{\vartheta}) = 0 \quad (22)$$

As before, given $\mathbf{x}_2$, we denote as $\mathcal{V}_2$ the set of all values of $\boldsymbol{\vartheta}$ that satisfy (22).

If both $\mathbf{x}_1$ and $\mathbf{x}_2$ are valid solutions, then $\boldsymbol{\vartheta}$ – which is the realization of the robots' displacement estimates and the distance measurement – must satisfy (21) *and* (22), and thus belongs to the set $\mathcal{V} = \mathcal{V}_1 \cap \mathcal{V}_2$. Note that since $\mathcal{V}$ is constrained by an additional equation (cf. (22)), $\mathcal{V} \subset \mathcal{V}_1$, and the dimension of $\mathcal{V}$ is smaller than that of $\mathcal{V}_1$ (Theorem 3, Ch. 9, §4 in [16]). Hence, the probability that the robots' trajectories are such that there exist two solutions is $|\mathcal{V}|/|\mathcal{V}_1|$, which is zero.[4] Following the same process, one can show that the probability of the event that 3 (or 4) solutions exist is also zero.

## IV. Determining the Relative Pose from 5 Distance Measurements: Unique Solution

We now treat the case where the robots $R_1$ and $R_2$ move again and arrive at the locations $\{9\}$ and $\{10\}$, respectively. At that point, they record their 5th distance measurement $d_{9,10}$ and also have available the estimates for their positions ${}^1\mathbf{p}_9$ and ${}^2\mathbf{p}_{10}$, respectively. We will first prove that in this case at most one solution exists (Section IV-A) generically and then propose an efficient and robust algorithm for computing its value (Section IV-B).

### A. Unique solution

*Lemma 3:* Given 5 distance measurements between 2 robots at 5 different locations, generically, there exists at most one solution for the 3 d.o.f. robot-to-robot transformation.

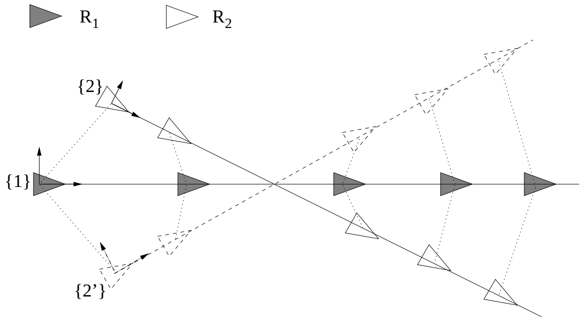


Fig. 3. An example of a degenerate case: When both robots move on straight lines there exist two solutions for the robot-to-robot transformation (i.e., two symmetric poses for $R_2$ denoted by $\{2\}$ and $\{2'\}$), regardless of how many distance measurements are available. The solid line represents the true trajectory of $R_2$, while the dashed line denotes the symmetric solution.

*Proof:* Following the same procedure as in Section II, we arrive at the following 4 equations (3 of these are the ones in (19) and the 4th one is computed in a similar manner using the latest distance measurement):

$$u_i c\phi + v_i s\phi = w_i \quad , \quad i = 1, \ldots, 4 \quad (23)$$

where the $u_i$'s, $v_i$'s, and $w_i$'s, are functions of $c\theta$, $s\theta$, and known quantities. Choosing all possible pairs of these 4 equations, we construct $\binom{4}{2} = 6$ systems of equations as the ones in (9). Employing the trigonometric constraint $x^2 + y^2 - 1 = 0$ and applying the same elimination process as in Section II, we derive 6 polynomial equations each of order 6, in the unknown variable $y := s\theta$ (cf. (15)):

$$\xi_{6,j} y^6 + \ldots + \xi_{1,j} y + \xi_{0,j} = 0 \quad (24)$$

where the $\xi_{i,j}$'s, $i = 0, \ldots, 6$, $j = 1, \ldots, 6$, are functions of measured and estimated quantities. Rewriting these polynomials in a matrix form, we have:

$$\begin{bmatrix} \xi_{6,1} & \cdots & \xi_{1,1} & \xi_{0,1} \\ \vdots & \ddots & \vdots & \vdots \\ \xi_{6,6} & \cdots & \xi_{1,6} & \xi_{0,6} \end{bmatrix} \begin{bmatrix} y^6 \\ \vdots \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \Longleftrightarrow \boldsymbol{\Xi}\mathbf{y} = \mathbf{0} \quad (25)$$

The null space of matrix $\boldsymbol{\Xi}$ in (25) has, generically, dimension 1. Thus, a unique solution for the vector $\mathbf{y}$, and hence for $y$, can be determined. Given $y := s\theta$, the unique values of the remaining unknowns, $s\theta$, $c\phi$, and $s\phi$, are computed via back-substitution as in the previous two cases. ■

Note that there exist singular cases where more than one solution exist and matrix $\boldsymbol{\Xi}$ will lose rank. For example, when both robots move on straight lines, two symmetric solutions exist (cf. Fig. 3). However, these events will not occur with probability 1 (cf. Corollary 2), and we can conclude that given 5 distance measurements, generically, there exists at most one solution for the 3 d.o.f. robot-to-robot transformation.

### B. Efficient Computation of the Unique Solution

The approach for computing the unique solution presented in the previous section, requires repetition of the elimination procedure of Section II 6 times. In addition to being time consuming, this method may result in incorrect values for

[4] $|\mathcal{V}|$ is a measure of the size of the set $\mathcal{V}$. If $\mathcal{V}$ is continuous, then $|\mathcal{V}|$ is its total volume.

the robot-to-robot transformation or even fail due to the accumulation of numerical errors. In this section, we present an alternative approach based on a *linear algorithm* for efficiently computing the unique solution given 5 distance measurements.

Note that this general method for solving systems of polynomial equations has been applied in the past for computing the pose of a camera given 4 bearing measurements to known landmarks [20], [21]. However, this is the first time that the distance-based robot-to-robot transformation problem is formulated so that it can be amenable to this form of solution.

As described in Sections II, III, and IV-A, for each of the last 4 distance measurements, $d_{34}, \ldots, d_{9,10}$, we can write an equation similar to (7), repeated below after rearranging terms and renaming the known quantities $\alpha_{i,j}$'s:

$$\begin{aligned}&\alpha_{7,j} c\phi + \alpha_{6,j} s\phi + \alpha_{5,j} c\theta + \alpha_{4,j} s\theta + \alpha_{3,j} c(\theta - \phi)\\ &+ \alpha_{2,j} s(\theta - \phi) + \alpha_{1,j} = 0 \quad , \quad j =, 1 \ldots, 4\end{aligned}$$

The unknowns in these 4 equations are $c\phi$, $s\phi$, $c\theta$, $s\theta$, $c(\theta-\phi)$, $s(\theta - \phi)$. Rewriting them in a matrix form, we have

$$\begin{bmatrix} \alpha_{7,1} & \cdots & \alpha_{1,1} \\ \vdots & \ddots & \vdots \\ \alpha_{7,4} & \cdots & \alpha_{1,4} \end{bmatrix} \begin{bmatrix} c\phi \\ s\phi \\ c\theta \\ s\theta \\ c(\theta-\phi) \\ s(\theta-\phi) \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \Longleftrightarrow \mathbf{A}\mathbf{x} = \mathbf{0}$$

where $\mathbf{A}$ is the $4 \times 7$ coefficient matrix (known), and $\mathbf{x}$ is the unknown vector we want to solve for. Once we have computed the three vectors $\mathbf{r}$, $\mathbf{s}$, and $\mathbf{t}$ that span the null space of $\mathbf{A}$, $\mathbf{x}$ can be written as:

$$\mathbf{x} = \lambda_1 \mathbf{r} + \lambda_2 \mathbf{s} + \lambda_3 \mathbf{t} \tag{26}$$

for some scalars $\lambda_1$, $\lambda_2$, $\lambda_3$. To determine their values, we use the trigonometric identities

$$\begin{aligned}&c^2\phi + s^2\phi = 1,\ c^2\theta + s^2\theta = 1,\ c^2(\theta-\phi) + s^2(\theta - \phi) = 1\\ &c\theta c\phi + s\theta s\phi = c(\theta - \phi),\ s\theta c\phi - c\theta s\phi = s(\theta - \phi)\end{aligned} \tag{27}$$

and the constraint

$$\lambda_1 r_7 + \lambda_2 s_7 + \lambda_3 t_7 = 1 \tag{28}$$

where $r_7$, $s_7$, and $t_7$ denote the 7th scalar elements of vectors $\mathbf{r}$, $\mathbf{s}$, and $\mathbf{t}$, respectively. Substituting the corresponding elements of $\mathbf{x}$ from (26) in the constraints (27), and eliminating $\lambda_3$ using (28), we obtain the following system of equations:

$$\begin{bmatrix} \beta_{1,1} & \cdots & \beta_{1,5} \\ \vdots & \ddots & \vdots \\ \beta_{5,1} & \cdots & \beta_{5,5} \end{bmatrix} \begin{bmatrix} \lambda_1^2 \\ \lambda_2^2 \\ \lambda_1\lambda_2 \\ \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} \gamma_1 \\ \vdots \\ \gamma_5 \end{bmatrix}$$

where $\beta_{i,j}$ and $\gamma_i$, $i$, $j = 1, \ldots, 5$ are functions of known quantities [17]. This system can be solved to uniquely determine the unknown vector $[\lambda_1^2\ \lambda_2^2\ \lambda_1\lambda_2\ \lambda_1\ \lambda_2]^T$. Given the values of $\lambda_1$ and $\lambda_2$, $\lambda_3$ is computed from (28) and subsequently $\mathbf{x}$ is uniquely determined from (26). Finally, the unknown robot-to-robot transformation is retrieved from the first 4 elements of $\mathbf{x}$.

## V. Weighted Least Squares Refinement

When 5 or more distance measurements are available to the robots, their relative pose can be computed with higher accuracy based on the following two-step process: (i) Compute the initial estimate for the 3 d.o.f. transformation from 5 distance measurements (cf. Section IV-B), (ii) Use the estimate from the previous step to initialize an iterative weighted least squares (WLS) algorithm that processes all distance measurements available. We hereafter describe the second step of this process.

Assume that the robots have recorded $n$ distance measurements, which are used to form a system of $n-1$ nonlinear equations equivalent to (5). Rearranging terms, these can be written in a compact form as

$$\mathbf{h}(\mathbf{x}, \boldsymbol{\vartheta}) = \mathbf{0} \tag{29}$$

where $\mathbf{x} := [\theta, \phi]^T$ denotes the vector of the unknown variables, and $\boldsymbol{\vartheta} := [\ {}^1\mathbf{p}^T\ {}^2\mathbf{p}^T\ \mathbf{z}^T\ ]^T$ is the vector comprising the following known quantities:

$$\begin{aligned} {}^1\mathbf{p} &:= [\ {}^1p_3^T\ \ldots\ {}^1p_{2n-1}^T\ ]^T \\ {}^2\mathbf{p} &:= [\ {}^2p_4^T\ \ldots\ {}^2p_{2n}^T\ ]^T \\ \mathbf{z} &:= [\ d_{12}\ \ldots\ d_{(2n-1)(2n)}\ ]^T \end{aligned}$$

Since ${}^1\mathbf{p}$, ${}^2\mathbf{p}$, and $\mathbf{z}$ are estimated or measured independently, the covariance matrix $\mathbf{P}_{\boldsymbol{\vartheta}\boldsymbol{\vartheta}}$ of $\boldsymbol{\vartheta}$ has a block diagonal structure:

$$\mathbf{P}_{\boldsymbol{\vartheta}\boldsymbol{\vartheta}} = \begin{bmatrix} \mathbf{P}_{11} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{22} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{R} \end{bmatrix}$$

where[5] $\mathbf{P}_{11} = E[{}^1\tilde{\mathbf{p}}{}^1\tilde{\mathbf{p}}^T]$ and $\mathbf{P}_{22} = E[{}^2\tilde{\mathbf{p}}{}^2\tilde{\mathbf{p}}^T]$ are the covariance matrices for the robot-position vectors ${}^1\mathbf{p}$ and ${}^2\mathbf{p}$, respectively, and $\mathbf{R} = \text{diag}(\sigma_{d_{ij}}^2)$ is the distance-measurements' noise covariance matrix, with $\sigma_{d_{ij}}$ denoting its standard deviation.

Given the estimate $\hat{\boldsymbol{\vartheta}}$ of $\boldsymbol{\vartheta}$, from the robots' odometry and the recorded distance measurements, and the initial estimate $\hat{\mathbf{x}}^{(1)}$ of $\mathbf{x}$, determined from the algebraic method of Section IV-B, the WLS algorithm computes the new estimate for $\mathbf{x}$ through the following iterative process:

$$\hat{\mathbf{x}}^{(\kappa+1)} = \hat{\mathbf{x}}^{(\kappa)} - \mathbf{P}_{\mathbf{xx}}\mathbf{H}_{\mathbf{x}}^T(\mathbf{H}_{\boldsymbol{\vartheta}}\mathbf{P}_{\boldsymbol{\vartheta}\boldsymbol{\vartheta}}\mathbf{H}_{\boldsymbol{\vartheta}}^T)^{-1}\mathbf{h}(\hat{\mathbf{x}}^{(\kappa)}, \hat{\boldsymbol{\vartheta}})$$

where $\mathbf{P}_{\mathbf{xx}} = [\mathbf{H}_{\mathbf{x}}^T(\mathbf{H}_{\boldsymbol{\vartheta}}\mathbf{P}_{\boldsymbol{\vartheta}\boldsymbol{\vartheta}}\mathbf{H}_{\boldsymbol{\vartheta}}^T)^{-1}\mathbf{H}_{\mathbf{x}}]^{-1}$ is the covariance of the estimates, and

$$\mathbf{H}_{\mathbf{x}} = \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}^{(\kappa)}, \boldsymbol{\vartheta}=\hat{\boldsymbol{\vartheta}}} \quad , \quad \mathbf{H}_{\boldsymbol{\vartheta}} = \left.\frac{\partial \mathbf{h}}{\partial \boldsymbol{\vartheta}}\right|_{\mathbf{x}=\hat{\mathbf{x}}^{(\kappa)}, \boldsymbol{\vartheta}=\hat{\boldsymbol{\vartheta}}}$$

are the Jacobians of the nonlinear function $\mathbf{h}$ [17].

Note that the iterative WLS process will converge if the robot-to-robot transformation is observable. In practice, we can

[5] We denote the error as $\tilde{\mathbf{p}} = \mathbf{p} - \hat{\mathbf{p}}$, where $\mathbf{p}$ and $\hat{\mathbf{p}}$ are the true and estimated vectors, respectively. Note also that the covariance matrices $\mathbf{P}_{11}$ and $\mathbf{P}_{22}$ are computed using state augmentation [22]. At every new robot position a copy of the current robot pose is added to the state vector. The covariance matrix is also appropriately augmented to include the covariance of the duplicate robot pose and its correlations with the rest of the poses in the state vector. Afterwards, only the copied state is propagated using odometric measurements. The interested reader is referred to [23] for details.

detect singular cases by checking the rank of $\mathbf{H}_{\mathbf{x}}$; if it is not of rank 2, the robots will need to move to new locations and acquire additional range measurements. In order to avoid such singular configurations, in the following section we present necessary and sufficient conditions on the robots' motion for the system to be observable.

## VI. Observability Analysis

The system describing the 3 d.o.f. robot-to-robot transformation given odometric and distance measurements is nonlinear. Therefore, tests designed for *linear time-invariant* systems (e.g., the Gramian matrix rank [24] or the Popov-Belevitch-Hautus (PBH) test [25]) cannot be used for examining its observability. Instead, we hereafter employ the observability rank criterion based on Lie derivatives [26] to determine the conditions under which our system is locally weakly observable. Recently, Mariottini *et al.* [11] have employed this criterion to investigate the observability of 2D leader-follower formations using only *bearing* measurements. In [12], Martinelli and Siegwart have also used this test to determine *only sufficient conditions* for the observability of cooperative localization for pairs of mobile robots navigating in 2D.

In this paper for the first time, we study the observability of the nonlinear system describing the time evolution of the robot-to-robot transformation given distance measurements and determine the *necessary and sufficient* observability conditions on the motion of the two robots. Specifically, after a brief review of the key concepts of observability (Section VI-A), the Lie derivatives and the observability rank condition (Section VI-B), in Section VI-C we prove that the sufficient conditions for the robot-to-robot transformation to be locally weakly observable are: (i)-(ii) both robots have nonzero linear velocities, (iii) at least one of them has nonzero rotational velocity. Additionally, we prove that conditions (i)-(ii) are also necessary.

### A. Observability of Nonlinear Systems

Consider the state-space representation of the following nonlinear system [27]:

$$\Sigma \left\{ \begin{array}{l} \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u}), \qquad \mathbf{x}(t_0) = \mathbf{x}_0 \\ \mathbf{y} = \mathbf{h}(\mathbf{x}) \end{array} \right. \tag{30}$$

where $\mathbf{x} \in M$ (a $C^\infty$ connected manifold of dimension $n$) is the state vector, $\mathbf{u} = [u_1 \dots u_l]^T \in \mathbb{R}^l$ is the vector of control inputs, and $\mathbf{y} = [y_1 \dots y_m]^T \in \mathbb{R}^m$ is the measurement vector, with $y_k = h_k(\mathbf{x})$, $k = 1, \dots, m$.

*Definition 1:* Two initial states $\mathbf{x}_0$ and $\mathbf{x}_1$ are *indistinguishable* if given the same input $\mathbf{u}(t)$, the system $\Sigma$ produces the same output $\mathbf{y}(t)$ for both initial states $\mathbf{x}_0$ and $\mathbf{x}_1$. The system is termed *observable* if for all $\mathbf{x} \in M$, the only state indistinguishable from $\mathbf{x}$ is $\mathbf{x}$ itself.

Notice that observability is a global concept. However, it might be necessary for the state to travel a considerable distance or for a long time to distinguish two points of $M$. For this reason, the following local concept is introduced [27]:

*Definition 2:* $\Sigma$ is *locally observable* at $\mathbf{x}_0 \in M$ if for every open neighborhood $U$ of $\mathbf{x}_0$, the set of points indistinguishable from $\mathbf{x}_0$ by trajectories in $U$ only consists of $\mathbf{x}_0$ itself. $\Sigma$ is *locally observable* if it is locally observable for every $\mathbf{x} \in M$.

In practice, it may be sufficient to distinguish $\mathbf{x}_0$ only from its neighbors (e.g., when some prior knowledge of $\mathbf{x}_0$ is available). However, it is possible that $\mathbf{x}_0$ is indistinguishable from states that are far away. This leads to the concept of weak observability [27].

*Definition 3:* $\Sigma$ is *weakly observable* at $\mathbf{x}_0$ if there exists an open neighborhood $U$ of $\mathbf{x}_0$ such that the only point in $U$ which is indistinguishable from $\mathbf{x}_0$ is $\mathbf{x}_0$ itself. The system $\Sigma$ is *weakly observable* if it is weakly observable at every $\mathbf{x} \in M$.

Notice again that it might be necessary to travel far away from $U$ to distinguish two points in $U$. For this reason, the following local concept is introduced [27].

*Definition 4:* $\Sigma$ is *locally weakly observable* at $\mathbf{x}_0$ if there exists an open neighborhood $U$ of $\mathbf{x}_0$ such that for every open neighborhood $V$ of $\mathbf{x}_0$ contained in $U$, the set of points indistinguishable from $\mathbf{x}_0$ in $U$ by trajectories in $V$ is $\mathbf{x}_0$ itself. The system $\Sigma$ is *locally weakly observable* if it is locally weakly observable for every $\mathbf{x} \in M$.

The advantage of local weak observability over other concepts is that it has a simple algebraic test which will be described in Section VI-B. Furthermore, and in the context of the robots' relative pose estimation problem, if the system is locally weakly observable and 3 or more distance measurements are available, then generically given an initial estimate that is close to the true solution, the iterative algorithm described in Section V (or higher-order variants of this) will converge to it. Additionally, if a unique solution exists (cf. Lemma 3, Section IV), the required initial estimate can be computed in closed form (Section IV-B), and the system is locally observable.

### B. Lie Derivatives and Nonlinear Observability

We consider the special case of system (30) where the process function, $\mathbf{f}$, can be separated into a summation of independent functions, each one excited by a different component of the control input vector, i.e.,

$$\left\{ \begin{array}{l} \dot{\mathbf{x}} = \mathbf{f}_0(\mathbf{x}) + \mathbf{f}_1(\mathbf{x})u_1 + \cdots + \mathbf{f}_l(\mathbf{x})u_l \\ \mathbf{y} = \mathbf{h}(\mathbf{x}) \end{array} \right. \tag{31}$$

where $\mathbf{f}_0$ is the zero-input function of the process model.

The zeroth-order Lie derivative of any (scalar) function is the function itself, i.e., $\mathfrak{L}^0 h_k(\mathbf{x}) = h_k(\mathbf{x})$. The first-order Lie derivative of function $h_k(\mathbf{x})$ with respect to $\mathbf{f}_i$ is defined as:

$$\begin{aligned} \mathfrak{L}^1_{\mathbf{f}_i} h_k(\mathbf{x}) &= \frac{\partial h_k(\mathbf{x})}{\partial x_1} f_{i1}(\mathbf{x}) + \cdots + \frac{\partial h_k(\mathbf{x})}{\partial x_n} f_{in}(\mathbf{x}) \\ &= \nabla h_k(\mathbf{x}) \cdot \mathbf{f}_i(\mathbf{x}) \end{aligned} \tag{32}$$

where $\mathbf{f}_i(\mathbf{x}) = [f_{i1}(\mathbf{x}) \ , \dots , \ f_{in}(\mathbf{x})]^T$, $\nabla$ represents the gradient operator, and '$\cdot$' denotes the vector inner product. Considering that $\mathfrak{L}^1_{\mathbf{f}_i} h_k(\mathbf{x})$ is a scalar function itself, the second-order Lie derivative of $h_k(\mathbf{x})$ with respect to $\mathbf{f}_i$ is:

$$\mathfrak{L}^2_{\mathbf{f}_i} h_k(\mathbf{x}) = \mathfrak{L}^1_{\mathbf{f}_i} \left( \mathfrak{L}^1_{\mathbf{f}_i} h_k(\mathbf{x}) \right) = \nabla \mathfrak{L}^1_{\mathbf{f}_i} h_k(\mathbf{x}) \cdot \mathbf{f}_i(\mathbf{x}) \tag{33}$$

Higher-order Lie derivatives are computed similarly. Additionally, it is possible to define mixed Lie derivatives, i.e.,

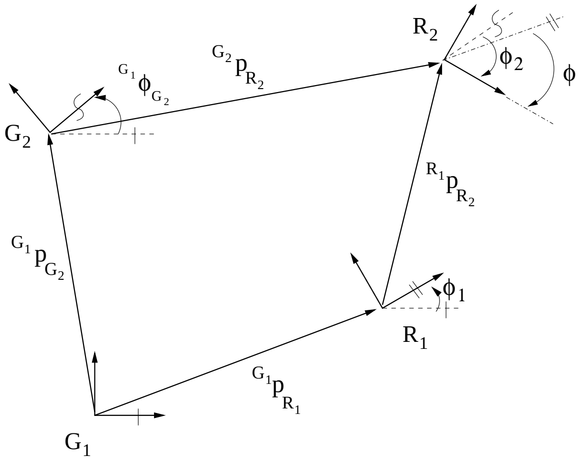


Fig. 4. Geometric relation between the two robots. The initial (global) frames of the two robots are denoted by $\{G_1\}$ and $\{G_2\}$, while their current poses are represented by the frames $\{R_1\}$ and $\{R_2\}$, respectively.

with respect to different functions of the process model. For example the second-order Lie derivative of $h_k$ with respect to $\mathbf{f}_j$ and $\mathbf{f}_i$, given its first derivative with respect to $\mathbf{f}_i$, is:

$$\mathfrak{L}^2_{\mathbf{f}_j\mathbf{f}_i} h_k(\mathbf{x}) = \mathfrak{L}^1_{\mathbf{f}_j}\left(\mathfrak{L}^1_{\mathbf{f}_i} h_k(\mathbf{x})\right) = \nabla \mathfrak{L}^1_{\mathbf{f}_i} h_k(\mathbf{x}) \cdot \mathbf{f}_j(\mathbf{x}) \tag{34}$$

Based on the preceding expressions for the Lie derivatives, the observability matrix is defined as the matrix with rows:

$$\mathcal{O} \triangleq \{\nabla \mathfrak{L}^{\ell}_{\mathbf{f}_i \cdots \mathbf{f}_j} h_k(\mathbf{x}) | i, j = 0 \ldots l; k = 1 \ldots m; \ell \in \mathbb{N}\} \tag{35}$$

The important role of this matrix in the observability analysis of a nonlinear system is captured by Theorems 3.1 and 3.11 in [26], repeated below:

*Theorem 1 (Observability Sufficient Condition):* If a system satisfies the observability rank condition then it is locally weakly observable.

*Definition 5 (Observability Rank Condition):* The observability rank condition is satisfied when the observability matrix (cf. (35)) is full rank.

*Theorem 2 (Observability Necessary Condition):* If a system is locally weakly observable, then the observability rank condition is satisfied generically.

In this case, "generically" means that the observability matrix is full rank everywhere except possibly within a subset of the domain of $\mathbf{x}$ [27]. Hence, if the observability matrix is not of sufficient rank for all values of $\mathbf{x}$, the system is *not* locally weakly observable [28].

### *C. Observability of the 2D Robot-to-Robot Transformation*

In this section, we first derive the continuous-time system model describing the relative position and orientation of the two robots and then analyze its observability, when range measurements are available.

Consider two robots that start from initial poses denoted by the (global) frames of reference $\{G_1\}$ and $\{G_2\}$, respectively. After some time, the robots move and their new poses are now depicted in Fig. 4 by the frames $\{R_1\}$ and $\{R_2\}$.

The 2×2 rotational matrices describing vector transformations between these frames satisfy the following relation:

$${}^{G_1}_{G_2}\mathbf{C}({}^{G_1}\phi_{G_2}) = {}^{G_1}_{R_1}\mathbf{C}(\phi_1){}^{R_1}_{R_2}\mathbf{C}(\phi){}^{R_2}_{G_2}\mathbf{C}(-\phi_2)$$

or, equivalently

$${}^{G_1}\phi_{G_2} = \phi_1 + \phi - \phi_2 \tag{36}$$

where $\phi$ denotes the relative orientation of the two robots.

Additionally, their positions satisfy the following geometric constraint (cf. Fig. 4):

$${}^{G_1}\mathbf{p}_{G_2} = {}^{G_1}\mathbf{p}_{R_1} + {}^{G_1}_{R_1}\mathbf{C}(\phi_1){}^{R_1}\mathbf{p}_{R_2} - {}^{G_1}_{G_2}\mathbf{C}({}^{G_1}\phi_{G_2}){}^{G_2}\mathbf{p}_{R_2} \tag{37}$$

where ${}^{R_1}\mathbf{p}_{R_2}$ is the robots' relative position vector.

Differentiating (36) and (37) with respect to time and noting that ${}^{G_1}\dot{\phi}_{G_2} = 0$ and ${}^{G_1}\dot{\mathbf{p}}_{G_2} = \mathbf{0}_{2\times 1}$, we have:

$$\dot{\mathbf{p}} = -\mathbf{e}_1 v_1 + \mathbf{C}(\phi)\mathbf{e}_1 v_2 - \mathbf{J}\mathbf{p}\omega_1 \tag{38}$$

$$\dot{\phi} = -\omega_1 + \omega_2 \tag{39}$$

with

$$\mathbf{p} := {}^{R_1}\mathbf{p}_{R_2} = \begin{bmatrix} p_x \\ p_y \end{bmatrix}, \mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \mathbf{J} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

where $v_i$ and $\omega_i = \dot{\phi}_i$, $i = 1, 2$, are the linear and rotational velocity, respectively, of robot $R_i$.

Furthermore, we rearrange the nonlinear kinematic equations (cf. (38) and (39)) in the following convenient form for computing the Lie derivatives:

$$\underbrace{\begin{bmatrix} \dot{p}_x \\ \dot{p}_y \\ \dot{\phi} \end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{f}_1} v_1 + \underbrace{\begin{bmatrix} c\phi \\ s\phi \\ 0 \end{bmatrix}}_{\mathbf{f}_2} v_2 + \underbrace{\begin{bmatrix} p_y \\ -p_x \\ -1 \end{bmatrix}}_{\mathbf{f}_3} \omega_1 + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}_{\mathbf{f}_4} \omega_2 \tag{40}$$

Finally in order to preserve the clarity of presentation, the measurement function is chosen to be the squared distance between the two robots divided by two, $d^2/2$, instead of the distance $d$, i.e.,

$$h(\mathbf{x}) = \frac{d^2}{2} = \frac{1}{2}\mathbf{p}^T\mathbf{p} \tag{41}$$

Note that $d$ and $d^2/2$ are both strictly positive, there is a one-to-one correspondence between them, and provide the same information for the spatial relation of the two robots.[6]

We hereafter compute the necessary Lie derivatives of $h$ and their gradients.

- *Zeroth-order Lie derivative* ($\mathfrak{L}^0 h$)

$$\mathfrak{L}^0 h = h = \frac{1}{2}\mathbf{p}^T\mathbf{p}$$

with gradient:

$$\nabla \mathfrak{L}^0 h = \begin{bmatrix} \mathbf{p}^T & 0 \end{bmatrix} = \begin{bmatrix} p_x & p_y & 0 \end{bmatrix}$$

[6] In practice, if $d^2/2$ is used as the measurement instead of $d$, it will change the probability density function of the noise in the observations and may affect the selection of the appropriate estimator for determining the unknown transformation. This choice, however, has no impact on the observability which is a property of the system and is independent of the observation noise.

- *First-order Lie derivatives* ($\mathfrak{L}^1_{\mathbf{f}_1}h$, $\mathfrak{L}^1_{\mathbf{f}_2}h$)

$$\begin{aligned}\mathfrak{L}^1_{\mathbf{f}_1}h &= \nabla\mathfrak{L}^0 h\cdot\mathbf{f}_1 = -p_x\\ \mathfrak{L}^1_{\mathbf{f}_2}h &= \nabla\mathfrak{L}^0 h\cdot\mathbf{f}_2 = p_x c\phi + p_y s\phi\end{aligned}$$

with gradients:

$$\begin{aligned}\nabla\mathfrak{L}^1_{\mathbf{f}_1}h &= \begin{bmatrix}-1 & 0 & 0\end{bmatrix}\\ \nabla\mathfrak{L}^1_{\mathbf{f}_2}h &= \begin{bmatrix}c\phi & s\phi & -p_x s\phi + p_y c\phi\end{bmatrix}\end{aligned}$$

- *Second-order Lie derivatives* ($\mathfrak{L}^2_{\mathbf{f}_1\mathbf{f}_2}h$, $\mathfrak{L}^2_{\mathbf{f}_3\mathbf{f}_1}h$, $\mathfrak{L}^2_{\mathbf{f}_4\mathbf{f}_2}h$)

$$\begin{aligned}\mathfrak{L}^2_{\mathbf{f}_1\mathbf{f}_2}h &= (\nabla\mathfrak{L}^1_{\mathbf{f}_2}h)\cdot\mathbf{f}_1 = -c\phi\\ \mathfrak{L}^2_{\mathbf{f}_3\mathbf{f}_1}h &= (\nabla\mathfrak{L}^1_{\mathbf{f}_1}h)\cdot\mathbf{f}_3 = -p_y\\ \mathfrak{L}^2_{\mathbf{f}_4\mathbf{f}_2}h &= (\nabla\mathfrak{L}^1_{\mathbf{f}_2}h)\cdot\mathbf{f}_4 = -p_x s\phi + p_y c\phi\end{aligned}$$

with gradients:

$$\begin{aligned}\nabla\mathfrak{L}^2_{\mathbf{f}_1\mathbf{f}_2}h &= \begin{bmatrix}0 & 0 & s\phi\end{bmatrix}\\ \nabla\mathfrak{L}^2_{\mathbf{f}_3\mathbf{f}_1}h &= \begin{bmatrix}0 & -1 & 0\end{bmatrix}\\ \nabla\mathfrak{L}^2_{\mathbf{f}_4\mathbf{f}_2}h &= \begin{bmatrix}-s\phi & c\phi & -p_x c\phi - p_y s\phi\end{bmatrix}\end{aligned}$$

- *Third-order Lie derivatives* ($\mathfrak{L}^3_{\mathbf{f}_3\mathbf{f}_1\mathbf{f}_2}h$, $\mathfrak{L}^3_{\mathbf{f}_4\mathbf{f}_1\mathbf{f}_2}h$, $\mathfrak{L}^3_{\mathbf{f}_4\mathbf{f}_4\mathbf{f}_2}h$)

$$\begin{aligned}\mathfrak{L}^3_{\mathbf{f}_3\mathbf{f}_1\mathbf{f}_2}h &= (\nabla\mathfrak{L}^2_{\mathbf{f}_1\mathbf{f}_2}h)\cdot\mathbf{f}_3 = -s\phi\\ \mathfrak{L}^3_{\mathbf{f}_4\mathbf{f}_1\mathbf{f}_2}h &= (\nabla\mathfrak{L}^2_{\mathbf{f}_1\mathbf{f}_2}h)\cdot\mathbf{f}_4 = s\phi\\ \mathfrak{L}^3_{\mathbf{f}_4\mathbf{f}_4\mathbf{f}_2}h &= (\nabla\mathfrak{L}^2_{\mathbf{f_4f_2}}h)\cdot\mathbf{f}_4 = -p_x c\phi - p_y s\phi\end{aligned}$$

with gradients:

$$\begin{aligned}\nabla\mathfrak{L}^3_{\mathbf{f}_3\mathbf{f}_1\mathbf{f}_2}h &= \begin{bmatrix}0 & 0 & -c\phi\end{bmatrix}\\ \nabla\mathfrak{L}^3_{\mathbf{f}_4\mathbf{f}_1\mathbf{f}_2}h &= \begin{bmatrix}0 & 0 & c\phi\end{bmatrix}\\ \nabla\mathfrak{L}^3_{\mathbf{f}_4\mathbf{f}_4\mathbf{f}_2}h &= \begin{bmatrix}-c\phi & -s\phi & p_x s\phi - p_y c\phi\end{bmatrix}\end{aligned}$$

Any other higher-order Lie derivatives are repetitions of $\pm c\phi$, $\pm s\phi$, $\pm(p_x c\phi + p_y s\phi)$, and $\pm(p_x s\phi - p_y c\phi)$ [17].

At this point, we present the main results of this section regarding the observability of the system under consideration:

*Lemma 4 (Sufficient Conditions):* The system (40)-(41) is locally weakly observable, if the following conditions are satisfied:

(i) $v_1 \neq 0$,
(ii) $v_2 \neq 0$,
(iii) $\omega_1 \neq 0$ or $\omega_2 \neq 0$,

*Proof:* When conditions (i)-(iii) are satisfied, and particularly $\omega_1 \neq 0$, the observability matrix

$$\mathcal{O} = \begin{bmatrix}\nabla\mathfrak{L}^1_{\mathbf{f}_1}h\\ \nabla\mathfrak{L}^2_{\mathbf{f}_3\mathbf{f}_1}h\\ \nabla\mathfrak{L}^2_{\mathbf{f}_1\mathbf{f}_2}h\\ \nabla\mathfrak{L}^3_{\mathbf{f}_3\mathbf{f}_1\mathbf{f}_2}h\end{bmatrix} = \begin{bmatrix}-1 & 0 & 0\\ 0 & -1 & 0\\ 0 & 0 & s\phi\\ 0 & 0 & -c\phi\end{bmatrix}$$

is full rank, because $\mathcal{O}^T\mathcal{O} = \mathbf{I}_3$. Alternatively, if $\omega_1 = 0$ and $\omega_2 \neq 0$, we replace the rows that involve $\mathbf{f}_3$ with those corresponding to $\mathbf{f}_4$. Then the observability matrix $\mathcal{O}$ is

$$\mathcal{O} = \begin{bmatrix}\nabla\mathfrak{L}^1_{\mathbf{f}_2}h\\ \nabla\mathfrak{L}^2_{\mathbf{f}_4\mathbf{f}_2}h\\ \nabla\mathfrak{L}^2_{\mathbf{f}_1\mathbf{f}_2}h\\ \nabla\mathfrak{L}^3_{\mathbf{f}_4\mathbf{f}_1\mathbf{f}_2}h\end{bmatrix} = \begin{bmatrix}c\phi & s\phi & -p_x s\phi + p_y c\phi\\ -s\phi & c\phi & -p_x c\phi - p_y s\phi\\ 0 & 0 & s\phi\\ 0 & 0 & c\phi\end{bmatrix}$$

which is also full rank, since $\det(\mathcal{O}^T\mathcal{O}) = 1$. Therefore, in both cases, the observability rank condition is satisfied and the system is locally weakly observable (cf. Theorem 1). ■

*Lemma 5 (Necessary Conditions):* The system (40)-(41) is *not* locally weakly observable, if the following conditions are *not* satisfied:

(i) $v_1 \neq 0$,
(ii) $v_2 \neq 0$.

*Proof:* We hereafter show that when any of the above two conditions is violated, the observability matrix is never full rank, and thus (cf. Theorem 2) the system is not locally weakly observable.

(i) If $v_1 = 0$, we can only select Lie derivatives that do not involve $\mathbf{f}_1$. In this case, the observability matrix is:

$$\mathcal{O}_1 = \begin{bmatrix}\nabla\mathfrak{L}^0 h\\ \nabla\mathfrak{L}^1_{\mathbf{f}_2}h\\ \nabla\mathfrak{L}^2_{\mathbf{f}_4\mathbf{f}_2}h\end{bmatrix}\begin{bmatrix}p_x & p_y & 0\\ c\phi & s\phi & -p_x s\phi + p_y c\phi\\ -s\phi & c\phi & -p_x c\phi - p_y s\phi\end{bmatrix}$$

which is singular since $\det(\mathcal{O}_1) = 0$.

(ii) If $v_2 = 0$, we can only choose Lie derivatives that do not involve $\mathbf{f}_2$. The observability matrix is now

$$\mathcal{O}_2 = \begin{bmatrix}\nabla\mathfrak{L}^0 h\\ \nabla\mathfrak{L}^1_{\mathbf{f}_1}h\\ \nabla\mathfrak{L}^2_{\mathbf{f}_3\mathbf{f}_1}h\end{bmatrix}\begin{bmatrix}p_x & p_y & 0\\ -1 & 0 & 0\\ 0 & -1 & 0\end{bmatrix}$$

which is also singular.

Hence, it is necessary that both linear velocities $v_1$ and $v_2$ are nonzero for the robot-to-robot relative pose to be locally weakly observable. ■

*Corollary 3:* Condition (iii) $\omega_1 \neq 0$ or $\omega_2 \neq 0$ is not necessary for the system (40)-(41) to be locally weakly observable.

*Proof:* If $\omega_1 = \omega_2 = 0$, then the relative orientation will remain constant, i.e., $\phi = \phi_0$, and the system equation becomes:

$$\begin{bmatrix}\dot{p}_x\\ \dot{p}_y\\ \dot{\phi}\end{bmatrix} = \underbrace{\begin{bmatrix}-1\\ 0\\ 0\end{bmatrix}}_{\mathbf{f}_1} v_1 + \underbrace{\begin{bmatrix}c\phi_0\\ s\phi_0\\ 0\end{bmatrix}}_{\mathbf{f}_2} v_2 \tag{42}$$

which is a special case of (40). The gradients of the nonzero Lie derivatives are:

$$\begin{aligned}\nabla\mathfrak{L}^0 h &= \begin{bmatrix}p_x & p_y & 0\end{bmatrix}\\ \nabla\mathfrak{L}^1_{\mathbf{f}_1}h &= \begin{bmatrix}-1 & 0 & 0\end{bmatrix}\\ \nabla\mathfrak{L}^1_{\mathbf{f}_2}h &= \begin{bmatrix}c\phi_0 & s\phi_0 & -p_x s\phi_0 + p_y c\phi_0\end{bmatrix}\\ \nabla\mathfrak{L}^2_{\mathbf{f}_1\mathbf{f}_2}h &= \begin{bmatrix}0 & 0 & s\phi_0\end{bmatrix}\end{aligned}$$

The observability matrix is now given by

$$\mathcal{O}_3 = \begin{bmatrix}-1 & 0 & 0\\ p_x & p_y & 0\\ 0 & 0 & s\phi_0\\ c\phi_0 & s\phi_0 & -p_x s\phi_0 + p_y c\phi_0\end{bmatrix}$$

which is full rank in general. Therefore the system is still locally weakly observable (cf. Theorem 1). Hence, (iii) $\omega_1 \neq 0$ or $\omega_2 \neq 0$ is a sufficient but *not* a necessary condition for the system to be locally weakly observable. ■

Finally, we note that Lemmas 4 and 5 provide sufficient and necessary conditions on the robots' motions for the 3 d.o.f. transformation to be *locally weakly* observable. If, additionally, 5 distance measurements are available (cf. Lemma 3), then generically the system is *locally* observable and the robot-to-robot transformation can be uniquely determined.

## VII. Simulation and Experimental Results

### A. Simulations

The purpose of our simulations is to verify the validity of the presented algorithms, and demonstrate the accuracy and robustness of Algorithm IV-B (cf. Section IV-B) against that of Algorithm IV-A (cf. Section IV-A) for computing the relative pose of two robots using 5 distance measurements.

For the results shown hereafter, the trajectories and distance measurements were generated as follows: (i) the two robots start at initial positions 10 m apart from each other and record their first distance measurement; (ii) each robot moves randomly for approximately 2 m; (iii) the robots record a distance measurement at their new positions. Steps (ii) and (iii) were repeated until 6 distance measurements were collected.

Each robot is modeled as a differential two-wheel drive vehicle equipped with encoders measuring the left, $v_{\ell_i}$, and right, $v_{r_i}$, wheel velocities, $i = 1,\ 2$. The linear and rotational velocity of each robot is given by:

$$v_i = \frac{v_{\ell_i} + v_{r_i}}{2}, \quad \omega_i = \frac{v_{r_i} - v_{\ell_i}}{a}, \qquad i = 1,\ 2$$

where $a = 0.35$ m is the distance between the wheels.

The wheel velocities are commanded to yield constant linear velocity $v_1 = v_2 = 1$ m/s, and rotational velocity $\omega_i$, $i = 1,\ 2$, that varies between $\pm 2.8$ rad/s following a uniform distribution. The measured wheel velocities are corrupted with zero-mean Gaussian noise with standard deviations $\sigma_{v_{\ell_i}} = \kappa v_{\ell_i}$, and $\sigma_{v_{r_i}} = \kappa v_{r_i}$, $i = 1,\ 2$. The distance measurement noise is also assumed to be additive zero-mean Gaussian with covariance $\mathbf{R} = \sigma_d^2 \mathbf{I}_n$. Both $\sigma_d$ and $\kappa$ (velocity noise standard deviation as percentage of the actually velocity) were used as parameters in our simulations for examining the robustness and accuracy of the presented algorithms.

Fig. 5 shows the averaged results of 2000 trials for each noise level. In particular, Fig.s 5(a) and 5(c) show the relative bearing $\theta$ error as a function of the noise standard deviation in the odometry and the distance measurements. The corresponding figures for the relative orientation $\phi$ error are similar to the ones for bearing and are omitted due to space limitations. To further highlight the difference in performance of the two algorithms, we fixed the standard deviation of the noise in the distance (odometry) measurements, and plotted the estimation error as a function of the odometry (distance) noise standard deviation in Fig. 5(b) (Fig. 5(d)). Evidently, the linear Algorithm IV-B is consistently more accurate than Algorithm IV-A over a wide range of values for $\kappa$ and $\sigma_d$.

As explained in Section V, in order to improve the accuracy of the relative pose estimates, a weighted least squares (WLS) refinement step is also necessary. However, every iterative method requires an initial estimate, whose accuracy greatly impacts the quality of the solution. Imprecise initial estimates can lead to local minima and even divergence. Fig. 6 compares the WLS performance when using the resulting estimates from Algorithms IV-A and IV-B for initialization. Fig. 6(a) shows the estimation errors when the WLS process *converged* to the true solution and the estimates are *consistent*. As evident, the estimation errors are comparable regardless of which algorithm is used for initialization. However, the impact of the initialization algorithm on the WLS's performance is better appreciated when considering additional factors such as: (i) percentage of inconsistent estimates, (ii) percentage of divergence, and (iii) required number of iterations. Specifically, the percentage of inconsistent estimates[7] is significantly larger when Algorithm IV-A, instead of IV-B is used for WLS initialization (cf. Fig. 6(b)). Furthermore, as shown in Fig.s 6(c) and 6(d), initialization with Algorithm IV-B reduces the number of WLS iterations required to converge and improves the divergence rate (we declare that the WLS process has not converged if it has reached 1000 iterations). Finally, we note that the estimation accuracy is significantly improved when additional distance measurements are available (cf. Fig 6(a)).

### B. Experiments

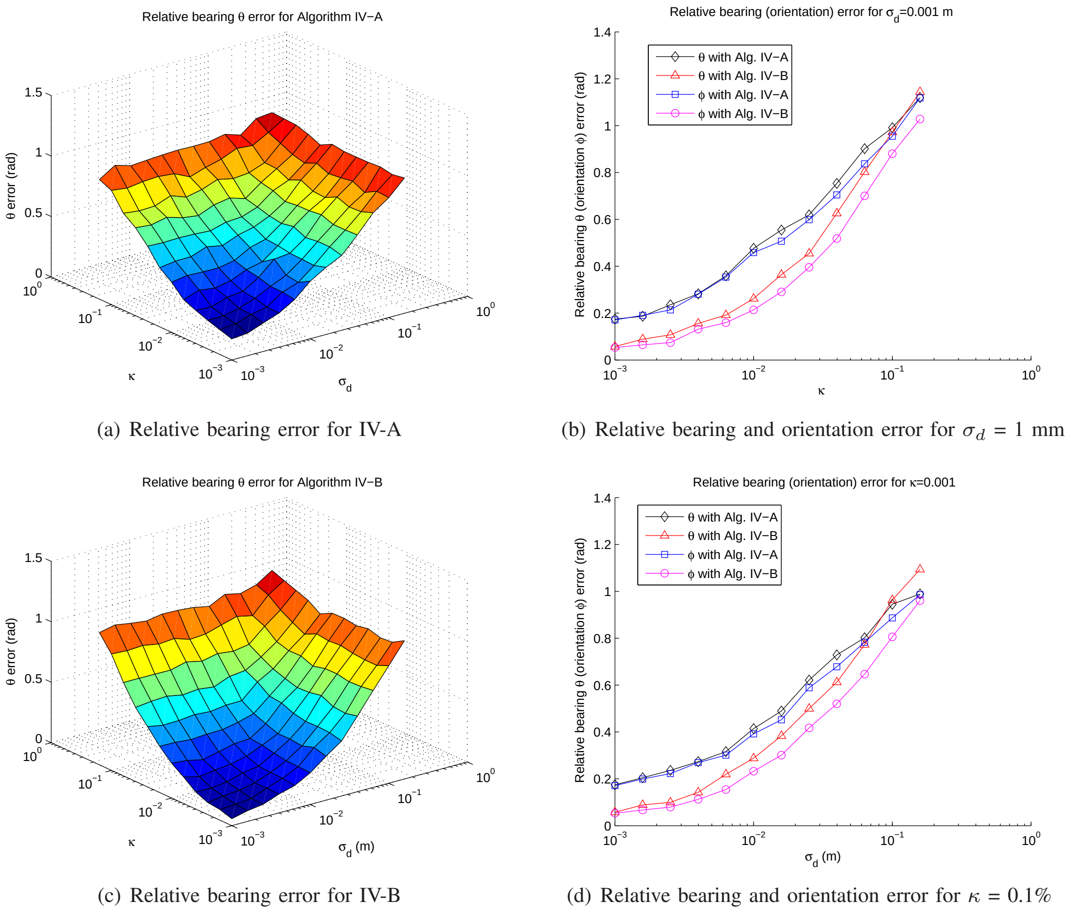
We hereafter experimentally validate the accuracy of the algorithms presented in Sections II-IV for determining the robot-to-robot transformation given 3, 4, or 5 range measurements.

For our experiments we deployed two Pioneer II robots within an area of 4 m × 5 m (cf. Fig. 7). The robots estimated their poses with respect to their initial locations using linear and rotational velocity measurements from their wheel-encoders. An overhead camera was used to provide ground truth for evaluating the errors in the computed estimates. Additionally, using the position measurements from the camera, we were able to compute the distances between the robots and control their accuracy by adding noise in these measurements. In this experiment, the standard deviation of the noise in the distance measurements was set to $\sigma_d = 0.01$ m.

Given 3 distance measurements, the 6 possible robot-to-robot transformations are shown in Fig. 8. By comparing each of them to the true robot trajectory (cf. Fig. 7), it is evident that Solution 4 (cf. Fig. 8(d)) corresponds to the true configuration. The same conclusion can be reached after processing 4 or 5 distance measurements (in this case, there exists only one real solution when considering 4 distance measurements).

The computed values for the two robot's relative bearing and orientation are shown in the first three columns of Table I for 3, 4, and 5 distance measurements respectively. Note that for the case of 3 distance measurements, only the solution closest to the true value (shown in the last column and computed using the camera) is included. These data show a slight decrease in accuracy when using additional distance measurements. This is to be expected since the distance measurements are used here to compute the *initial* estimate, at time $t = 0$, for the robot-to-robot transformation, i.e., when the two robots first detected each other. In this case, as the robots move

[7] An estimate is considered inconsistent if the estimation error is larger than $3\sigma$, where $\sigma$ is obtained from the diagonal elements of the covariance matrix $\mathbf{P_{xx}}$ computed using the iterative WLS algorithm (cf. Section V).



(a) Relative bearing error for IV-A

(b) Relative bearing and orientation error for $\sigma_d = 1$ mm

(c) Relative bearing error for IV-B

(d) Relative bearing and orientation error for $\kappa = 0.1\%$

Fig. 5. Simulation results, averaged over 2000 trials, for Algorithms IV-A and IV-B. (a), (c): relative bearing $\theta$ estimation error vs. standard deviation of the noise in the odometry and distance measurements. (b): relative bearing $\theta$ and relative orientation $\phi$ estimation errors vs. standard deviation of the odometry noise (for constant distance noise standard deviation). (d): relative bearing $\theta$ and relative orientation $\phi$ estimation errors vs. standard deviation of the distance noise (for constant odometry noise standard deviation).

and accumulate odometry error, the uncertainty in the initial transformation will increase. However, this would not be in general the case if instead we were to compute the robot-to-robot transformation at subsequent time instants. Actually, as it was shown in Section VI, the system describing the *current* robot-to-robot transformation is locally observable and hence its error will remain bounded.

Finally, we tested the accuracy of the WLS process initialized with the estimate computed by Algorithm IV-B given 5 distance measurements. As shown in the 4th column of Table I, the WLS estimates for the relative bearing and orientation are $\theta = -0.9319$ rad, and $\phi = 0.3328$ rad, both of which are of higher accuracy compared to the initial estimates.

TABLE I
RESULTS WITH 3, 4, 5 DISTANCE MEASUREMENTS

| No. meas. | 3 | 4 | 5 | 5 (WLS) | Cam |
|---|---|---|---|---|---|
| $\theta$ (rad) | -0.9490 | -0.9469 | -0.9501 | -0.9319 | -0.9160 |
| $\phi$ (rad) | 0.3312 | 0.3286 | 0.3353 | 0.3328 | 0.3280 |

## VIII. Conclusions and Future Work

In this paper, we presented efficient algorithms for solving the relative pose problem for pairs of robots moving on a plane using robot-to-robot distance measurements. Non-iterative algorithms for computing the initial estimate of the 3 d.o.f. transformation were presented for the cases when 3, 4, and 5 distance measurements were available. We have shown that for nonsingular configurations the maximum number of

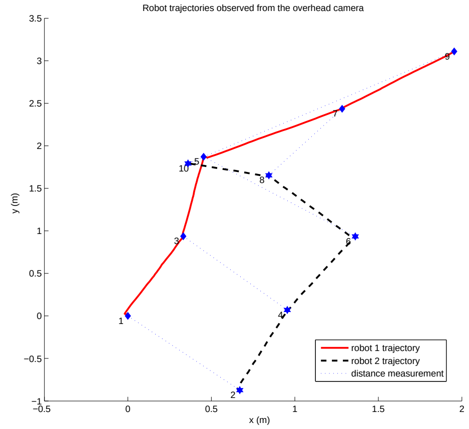
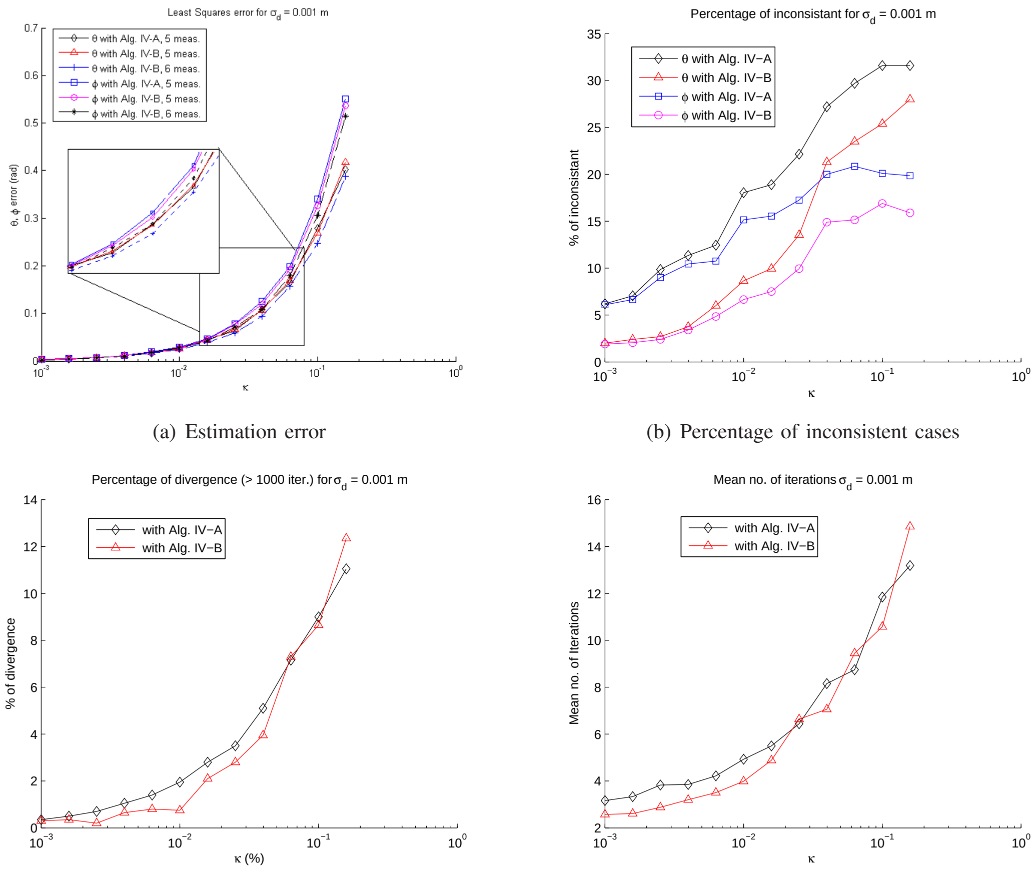


Fig. 7. The trajectories of the two robots and the locations where distance measurements were recorded.



(a) Estimation error

(b) Percentage of inconsistent cases

(c) Percentage of diverged cases

(d) Mean number of iterations

Fig. 6. Performance of the Weighted Least Squares refinement step when using Algorithms IV-A and IV-B for initialization. (a) Final $\theta$ and $\phi$ estimation errors for the cases that converge to the true solution (consistent). (b) Percentage of cases which converge to local minima (inconsistent). (c) Percentage of cases where the maximum number of iterations (1000) is reached. (d) Average number of iterations required to convergence.

solutions for the above cases are 6, 4, and 1, respectively. In addition, we presented a novel linear algorithm for computing the unique solution that is robust to numerical errors. A key advantage of our approach is that it does not require any robot coordination or specific motion strategies, thus reducing the time and effort required. Furthermore, a weighted least squares process was presented that uses the result of the non-iterative algorithm as the initial relative pose estimate and iteratively improves its accuracy. Finally, we presented necessary and sufficient observability conditions for the motion of the robots based on Lie derivatives.

Currently we investigate optimal motion strategies for minimizing the uncertainty in the relative pose estimates and seek to extend the results of this work to motion in 3D.

## References

[1] X. S. Zhou and S. I. Roumeliotis, “Multi-robot SLAM with unknown initial correspondence: The robot rendezvous case,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China, Oct. 9 - 15 2006, pp. 1785–1792.

[2] P. Bahl and V. Padmanabhan, “Radar: An in-building RF-based user location and tracking system,” in *Proceedings of the 19th Annual Joint Conference of IEEE Computer and Communications Society*, Tel Aviv, Israel, Mar. 26-30 2000, pp. 775–784.

[3] J. Djugash, S. Singh, and P. I. Corke, “Further results with localization and mapping using range from radio,” in *Proceedings of the International Conference on Field and Service Robotics*, Jul. 29 - 31 2005.

[4] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, “The cricket location-support system,” in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, Boston, MA, Aug. 6 - 11 2000, pp. 32–43.

[5] L. Girod and D. Estrin, “Robust range estimation using acoustic and multimodal sensing,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Maui, HI, Oct. 29 - Nov. 3 2001, pp. 1312–1320.

[6] J. Aspnes, T. Eren, D. K. Goldenberg, A. S. Morse, W. Whiteley, Y. R. Yang, B. D. O. Anderson, and P. N. Belhumeur, “A theory of network localization,” *IEEE Transactions on Mobile Computing*, vol. 5, no. 12, pp. 1663–1678, Dec. 2006.

[7] L. Doherty, K. S. J. Pister, and L. E. Ghaoui, “Convex position estimation in wireless sensor networks,” in *Proceedings of the 20th Annual Joint Conference of IEEE Computer and Communications Society*, Anchorage, AK, Apr. 22 - 26 2001, pp. 1655–1663.

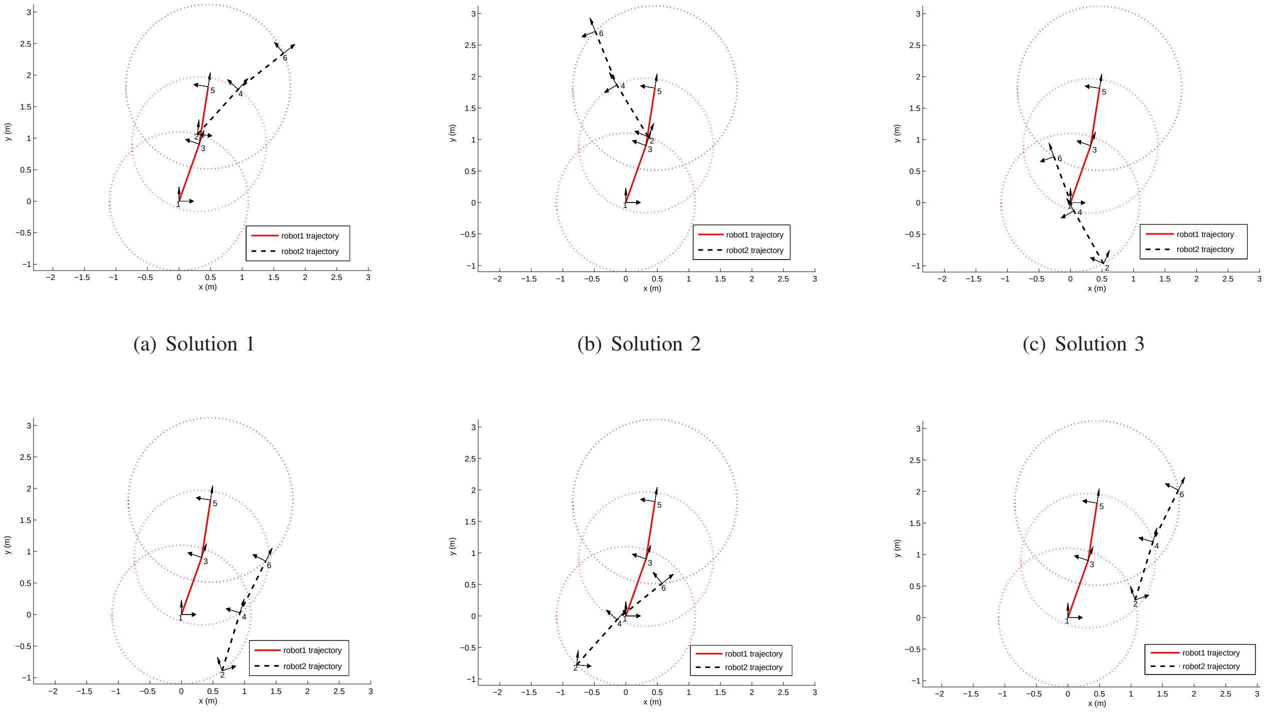
[8] Y. Shang, W. Ruml, Y. Zhang, and M. P. J. Fromherz, “Localization from mere connectivity,” in *Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, Annapolis, MD, June 1 - 3 2003, pp. 201–212.

[9] A. Savvides, H. Park, and M. B. Srivastava, “The bits and flops of the n-hop multilateration primitive for node localization problems,” in *Proceedings of the International Workshop on Sensor Networks and Applications*, Atlanta, GA, Sep. 28 2002, pp. 112–121.

[10] J. A. Costa, N. Patwari, and A. O. Hero, “Distributed weighted-multidimensional scaling for node localization in sensor networks,” *ACM Transactions on Sensor Networks*, vol. 2, no. 1, pp. 39–64, Feb. 2006.

[11] G. L. Mariottini, G. Pappas, D. Prattichizzo, and K. Daniilidis, “Vision-based localization of leader-follower formations,” in *Proceedings of the 44th IEEE Conference on Decision and Control, and European Control Conference*, Seville, Spain, Dec. 12 - 15 2005, pp. 635–640.

[12] A. Martinelli and R. Siegwart, “Observability analysis for mobile robot



(a) Solution 1

(b) Solution 2

(c) Solution 3

(d) Solution 4

(e) Solution 5

(f) Solution 6

Fig. 8. An experiment with 6 real solutions. Solution 4 is the true robot configuration. The distance measurements are depicted by circles centered at robot $R_1$ with radii equal to the distance to robot $R_2$ at each location.

localization," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Edmonton, Canada, Aug. 2 -6 2005, pp. 1471–1476.

[13] S. I. Roumeliotis and G. A. Bekey, "Distributed multi-robot localization," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 781–795, Oct. 2002.

[14] X. S. Zhou and S. I. Roumeliotis, "Determining the robot-to-robot relative pose using range-only measurements," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Rome, Italy, Apr. 10 - 14 2007, pp. 4025–4031.

[15] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, 2nd ed. Cambridge University Press, 1992.

[16] D. Cox, J. Little, and D. O'Shea, *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*, 2nd ed. Springer, 1996.

[17] X. S. Zhou and S. I. Roumeliotis, "Determining the robot-to-robot relative pose using range-only measurements," University of Minnesota, Minneapolis, MN, Tech. Rep., May 2006. [Online]. Available: www.cs.umn.edu/˜zhou/paper/distOnly.pdf

[18] A. G. Akritas, "Sylvester's form of resultant and the matrix-triangularization subresultant prs method," in *Proceedings of the Conference on Computer Aided Proofs in Analysis*, Cincinnati, OH, Mar. 1989, pp. 5–11.

[19] A. Edelman and H. Murakami, "Polynomial roots from companion matrix eigenvalues," *Mathematics of Computation*, vol. 64, no. 210, pp. 763–776, 1995.

[20] L. Quan and Z. Lan, "Linear n-point camera pose determination," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 8, pp. 774–780, 1999.

[21] A. Ansar and K. Daniilidis, "Linear pose estimation from points or lines," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 578–589, May 2003.

[22] S. I. Roumeliotis and J. W. Burdick, "Stochastic cloning: A generalized framework for processing relative state measurements," in *Proccedings of the IEEE International Conference on Robotics and Automation*, Washington D.C., May 11-15 2002, pp. 1788–1795.

[23] A. I. Mourikis, S. I. Roumeliotis, and J. W. Burdick, "SC-KF mobile robot localization: A stochastic cloning-kalman filter for processing relative-state measurements," *IEEE Transactions on Robotics*, vol. 23, no. 4, pp. 717–730, Aug. 2007.

[24] P. S. Maybeck, *Stochastic Models, Estimation, and Control*. New York: Academic Press, 1979, vol. 1.

[25] W. J. Rugh, *Linear System Theory*, 2nd ed. Prentice Hall, 1996.

[26] R. Hermann and A. Krener, "Nonlinear controllability and observability," *IEEE Transactions on Automatic Control*, vol. 22, no. 5, pp. 728–740, Oct. 1977.

[27] J. L. Casti, "Recent developments and future perspectives in nonlinear system theory," *SIAM Review*, vol. 24, no. 3, pp. 301–331, Jul. 1982.

[28] W. Respondek, "Geometry of static and dynamic feedback," Lecture Notes at the Summer School on Mathematical Control Theory, Trieste, Italy, Sep. 2001.

PLACE
PHOTO
HERE

**Xun S. Zhou** received his B.Sc. in Physics with honors from the Zhongshan University, China in 2000, and the M.Sc. in Computer Science from the Bradley University, IL in 2003. He is currently a Ph.D student in the Department of Computer Science and Engineering (CSE) at the University of Minnesota, MN. His research interests lie in the areas of single- and multi-robot systems, localization and mapping, and multi-sensor extrinsic calibration. He is the recipient of the 2007 Excellence in Research Award from the CSE Department of the University of Minnesota, and in 2006 he was the Finalist for the Best Paper Award of the IEEE/RSJ International Conference on Intelligent Robots and Systems.

PLACE
PHOTO
HERE

**Stergios I. Roumeliotis** received his Diploma in Electrical Engineering from the National Technical University of Athens, Greece, in 1995, and the M.S. and Ph.D. degrees in Electrical Engineering from the University of Southern California, CA in 1999 and 2000 respectively. From 2000 to 2002 he was a Postdoctoral Fellow at the California Institute of Technology, CA. Between 2002 and 2008 he was an Assistant Professor, and currently he is an Associate Professor in the Department of Computer Science and Engineering at the University of Minnesota, MN. His research interests include inertial navigation of aerial and ground autonomous vehicles, distributed estimation under communication and processing constraints, and active sensing for networks of mobile sensors.

S.I. Roumeliotis is the recipient of the NSF CAREER award (2006), the McKnight Land-Grant Professorship award (2006-08), the ICRA Best Reviewer Award (2006), and he is the co-recipient of the One NASA Peer award (2006), and the One NASA Center Best award (2006). Papers he has co-authored have received the Robotics Society of Japan Best Journal Paper award (2007), the ICASSP Best Student Paper award (2006), the NASA Tech Briefs award (2004), and one of them was the Finalist for the IROS Best Paper Award (2006). S.I. Roumeliotis is currently serving as Associate Editor for the IEEE Transactions on Robotics.