2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Robot-to-Robot Relative Pose Estimation based on Semidefinite Relaxation Optimization

Ming Li[1,2], Guanqi Liang[1,2], Haobo Luo[1,2], Huihuan Qian[1,2], and Tin Lun Lam[1,2,†]

Abstract— **In this paper, the 2D robot-to-robot relative pose (position and orientation) estimation problem based on egomotion and noisy distance measurements is considered. We address this problem using an optimization-based method, which does not require complicated numerical analysis while yields no inferior relative localization (RL) results compared to existing approaches. In particular, we start from a state-of-the-art method named square distances weighted least square (SD-WLS), and reformulate it as a non-convex quadratically constrained quadratic programming (QCQP) problem. To handle its non-convex nature, a semidefinite programming (SDP) relaxation optimization-based method is proposed, and we prove that the relaxation is tight when measurements are free from noise or just corrupted by small noise. Further, to obtain the optimal solution of the relative pose estimation problem in the sense of maximum likelihood estimation (MLE), a theoretically optimal WLS method is developed to refine the estimate from the SDP optimization. Comprehensive simulations and well-designed experiments are presented for validating the tightness of the SDP relaxation, and the effectiveness of the proposed algorithm is highlighted by comparing it to the existing approaches.**

I. Introduction

Relative localization (RL), which refers to determine relative configurations of mobile agents with respect to (w.r.t.) other agents or landmark, is critically important in multi-robot systems since it is the pre-requisite for robot teaming and swarming [1]–[3]. Many applications, such as target tracking [4], localization [5] and mapping [6] are often faced with RL problem. In the case of static multi-robot systems, a variety of algorithms based on convex optimization [7], sum of squares relaxation [8], multidimensional scaling [9] and flying anchor nodes approach, were employed to address the RL problem, where these methods can precisely locate the position of sensor nodes, meantime, fast and robust. However, w.r.t. mobile multi-robot systems, except for position, the relative orientation also has to be determined, and this is emphasized by plenty of literature [10]–[12]. Among the vast investigations, however, the relative pose estimation problem is usually supposed to be solved, and only a few works describe how to locate the relative pose in multi-robot systems.

Manually measuring the relative pose between robots is simple and straightforward, but tedious, with low accuracy and not applicable to large robot teams. Alternatively, utilizing external references to achieve RL, such as global positioning system (GPS), compass, maps of the environment or image matching, is regarded as an efficient method and indeed widely used in practice. Unfortunately, this type of approach is not available in some scenarios like underwater, underground, outer space, or indoors. Localization based purely on proprioceptive sensors (e.g., from odometry), also known as dead reckoning (DR), suffers from unbounded error accumulation over time. This phenomenon arises because DR continuously integrates noise and biases affecting the sensor measurements. Another convenient method is using the inter-robot distance and bearing measurements [13]. However, due to power and processing constraints, only the distance or bearing measurements is available in a multi-robot system. In this regard, RL can be achieved by combining the information from egomotion (obtained from DR) with bearing [14] or distance [15] measurements. Actually, the latter case (i.e., combine the egomotion information with distance-only measurements) is exactly what we investigated in this paper.

Generally, the relative pose estimation using egomotion and distance-only measurements can be cast as a nonlinear optimization problem and solved by the iterative method. For example, the two-step-based method [15], [16], where the deterministic algorithm is first used to compute a coarse initial solution, and then the iterative weight least squares (WLS) is employed to iteratively refine this initial estimate using additional range measurements. However, the quality of the RL result is highly dependent on the accuracy of initial estimate, which hinders the application of this approach to noise-corrupted scenarios. Alternatively, a fundamentally different way is presented in [17]. Instead of using the standard numerical optimization method, the WLS cost function, which obtains from maximum likelihood estimation (MLE) with the assumption of Gaussian noise, is transformed into polynomial form, and then the recent techniques like polynomial system solving and polynomial optimization method are employed to find all stationary points of the cost function. However, this method scales quite poorly since the complexity grows exponentially with the increase of measurements. To this end, a much preferable approach from the point of scalability is to solve the WLS based on noisy measurements of the squared distance. Thus a relaxed WLS formation basing on square distance measurements (SD-WLS) is developed, and the hybrid algebra-numeric technique basing on eigenvalue decomposition of a complex matrix is employed to compute the global optimum of the SD-WLS efficiently.

In this paper, we propose to provide an alternative solution for the SD-WLS, which avoids using the slightly complicated

This paper is partially supported by funding 2019-INT008 from the Shenzhen Institute of Artificial Intelligence and Robotics for Society.
(†*Corresponding author: Tin Lun Lam, email: tllam@cuhk.edu.cn*).
[1] The Chinese University of Hong Kong, Shenzhen, China.
[2] Shenzhen Institute of Artificial Intelligence and Robotics for Society.

hybrid algebra-numeric technique and yields an entirely equivalent RL result. In particular, we reformulate the SD-WLS as a non-convex QCQPs problem, and an efficient SDP relaxation optimization method is derived to address this problem. Meantime, we prove that when the in- and ex-sensors' observations are noise free or corrupted by small enough noise, the SDP relaxation can obtain the optimum of the QCQP. We also propose a theoretically optimal WLS method to refine the estimate from the SDP optimization and get an MLE of the relative pose, which is proved to be advantageous when the measurement noise is sufficiently large. Comprehensive simulations and well-designed experiments are presented for validating the tightness of the SDP relaxation, and the effectiveness (e.g., estimation accuracy and convergence speed) of the proposed algorithm is highlighted by comparing to the existing approaches.

II. Problem Formulation

Consider two robots $\mathcal{R}_1$ and $\mathcal{R}_2$ moving randomly in a 2D space, where their initial poses are indicated by the frames of reference $\mathcal{F}_1$ and $\mathcal{F}_2$, respectively. The two robots acquire N robot-to-robot distance measurements $d_l, l = 1, ..., N$ while moving in a 2D plane[1]. As shown in Fig. 1, an illustration of the considered 2D RL setup is given. Here, our goal is to determine the 3 degree-of-freedom (3DOF) robot-to-robot transformation from displacement estimates and distance measurements. To achieve this, we make some assumptions for each robot as follows:

- Each robot is equipped with proprioceptive motion sensors such that it can measure changes in its own pose;
- Mutual distance (include the identity) of the measured robot can be obtained from each robot with the exteroceptive sensors;
- Each robot has its own communication capability, which means that each robot is able to broadcast information to its neighboring robots.

At each time instant l, the measurement of $\mathcal{R}_1$ and $\mathcal{R}_2$, i.e., z_l, is corrupted by Gaussian noise v_k with zero mean and covariance σ_l^2, and produced according to

$$z_l = \sqrt{\mathbf{w}_l^T \mathbf{w}_l} + v_l, \tag{1}$$

where $\mathbf{w}_l$ is defined by

$$\mathbf{w}_l := \mathcal{T}(\mathbf{q}_{2,l}) - \mathbf{q}_{1,l} = \mathbf{p} + \mathbf{C}\mathbf{q}_{2,l} - \mathbf{q}_{1,l}, \tag{2}$$

$\mathcal{T}(\cdot)$ denotes the transform funcation, $\mathbf{q}_{1,l}$ and $\mathbf{q}_{2,l}$ denote the displacements of $\mathcal{R}_1$ and $\mathcal{R}_2$ in their respective frames, N is the moving step, $\mathbf{p}$ and $\mathbf{C}$ are the initial position and rotation matrix, respectively, and they are defined by

$$\mathbf{p} := d_0 \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}, \quad \mathbf{C} := \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix}, \tag{3}$$

[1]It is reasonable to assume that only robot $\mathcal{R}_1$ or $\mathcal{R}_2$ collects range measurement at each location. However, if both robots record a measurement, i.e., two measurements are obtained, a more accurate estimate of their distance can be given with the combination of the two measurements.

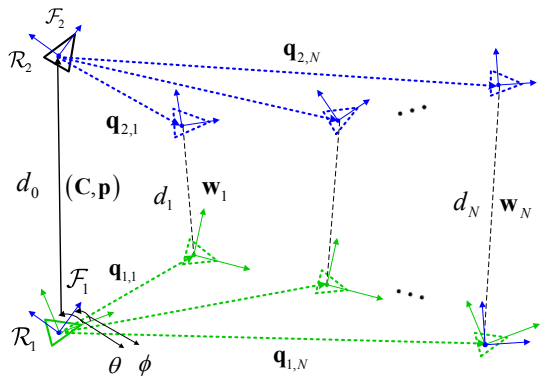

Fig. 1. The trajectories of robots $\mathcal{R}_1$ and $\mathcal{R}_2$: We denote the coordinates of $\mathcal{R}_1$ using green color, while $\mathcal{R}_2$ using blue, $\theta and \phi$ denote the relative direction and orientation of $\mathcal{R}_2$ with respect to $\mathcal{R}_1$, respectively.

where d_0 is a considered known distance measurement between $\mathcal{R}_1$ and $\mathcal{R}_2$ at initial stage, θ and ϕ are the relative angle and orientation of robot $\mathcal{R}_1$ and $\mathcal{R}_2$ at $l = 0$, respectively. According to the SD-WLS presented in [17], taking the square of (1), we have

$$z_l^2 = \mathbf{w}_l^T \mathbf{w}_l + 2v_l\sqrt{\mathbf{w}_l^T \mathbf{w}_l} + v_l^2. \tag{4}$$

Actually, the corresponding noise term $v_l^{'} := 2v_l\sqrt{\mathbf{w}_l^T \mathbf{w}_l} + v_l^2$ is not zero-mean Gaussian. However, as demonstrated in [18], the non-Gaussian random variable can be well approximated by a Gaussian density function with matching first and second order moments. Therefore, this equation can be rewritten as

$$z_l^{'} = \mathbf{w}_l^T \mathbf{w}_l + v_l^{'}, \tag{5}$$

where

$$\begin{aligned} &z_l^{'} \simeq z_l^2 - \bar{v}_l^{'}, \qquad \bar{v}_l^{'} := \mathbf{E}[v_l^{'}] = \mathbf{R}_{ll}, \\ &\mathbf{\Sigma}_{ll} := \mathbf{E}[(v_l^{'} - \bar{v}_l^{'})^2] = \mathbf{R}_{ll}(4d_l^2 + 2\mathbf{R}_{ll}) \\ &\mathbf{\Sigma}_{lk} := \mathbf{E}[(v_l^{'} - \bar{v}_l^{'})(v_k^{'} - \bar{v}_k^{'})] = \mathbf{R}_{ll}(4d_l^2 + 2\mathbf{R}_{ll}). \end{aligned} \tag{6}$$

Note that $\mathbf{E}[\cdot]$ denotes the expectation of a specific matrix, $\mathbf{R}_{ll}$ is one of the elements of $\mathbf{R}$ with entry $\{l, l\}$, and the details of computing $\mathbf{R}$ can be found in [17]. Since N distance measurements are recorded, thus N nonlinear equations equivalent to (6) can be obtained. We stack those equations, it is easy to have

$$\mathbf{z}^{'} = \mathbf{h}_{sd} + \mathbf{v}^{'}, \tag{7}$$

where

$$\begin{aligned} &\mathbf{z}^{'} = \left[z_1^{'} \cdots z_N^{'}\right]^T, \quad \mathbf{h}_{sd} = \left[\mathbf{w}_1^T \mathbf{w}_1 \cdots \mathbf{w}_N^T \mathbf{w}_N\right]^T, \\ &\mathbf{v}^{'} = \left[v_1^{'} \cdots v_N^{'}\right]^T \sim \mathcal{N}(\mathbf{v}^{'}; \mathbf{0}, \mathbf{R}^{'}), \end{aligned} \tag{8}$$

and $\mathbf{R}^{'}$ is approximated by $\mathbf{R}^{'} \simeq \mathbf{\Sigma}$ ($\mathbf{\Sigma}$ is calculated by (6)). Therefore, the SD-WLS function can be formulated as:

$$\begin{aligned} \min_{\theta,\phi} \quad & \frac{1}{2}(\mathbf{h}_{sd} - \mathbf{z}^{'})^T \mathbf{R}^{'-1}(\mathbf{h}_{sd} - \mathbf{z}^{'}), \\ \text{s.t.} \quad & \cos\phi^2 + \sin\phi^2 = 1. \end{aligned} \tag{9}$$

Remark 1. *As we see in (9), the optimization parameter for SD-WLS is θ and ϕ. Of course we can use $\mathbf{p}$ and ϕ instead, which is the same as paper [17]. Here, we replace $\mathbf{p}$ with θ just for simplicity, and it is reasonable because we assume that d_0 can be directly obtained from external sensors. Additionally, in 3D scenario, parameter ϕ should be replaced by a quaternion $\bar{q}$, which can fully construct the rotation matrix $\mathbf{C}$. Consistently, the reason why for the parameter ϕ should be optimized is due to that the quaternion in 2D can be specified by $\bar{q} = \begin{bmatrix} 0 & 0 & sin(\phi/2) & sin(\phi/2) \end{bmatrix}$.*

III. Semidefinite Optimization for Relative Pose Estimation

In this section, we prove that SD-WLS can be reformulated as an equivalent non-convex QCQP problem, and the SDP relaxation technique is used to handle its non-convex nature. Meantime, we prove that if observations are noise-free or corrupted by small enough noise ("small enough" is depended on the specific applications), the SDP relaxation can obtain the optimum of the QCQP, and a recovery strategy is provided to get the solution of the original problem.

A. SDP Relaxation

Proposition 1. *We assume that $\{\mathbf{q}_{1,l}, \mathbf{q}_{2,l}, d_l\}_{l=1}^N$ are obtained from in- and ex- sensors (d_0 is known), thus the SD-WLS can be equivalently formulated as*

$$\begin{aligned} f_{OPT}^* = \min_{\mathbf{x}} \quad & \mathbf{x}^T \mathbf{M}_0 \mathbf{x} \\ \text{s.t.} \quad & \mathbf{x}^T \mathbf{M}_l \mathbf{x} = b_l, \quad l = 1, ..., N \end{aligned} \tag{10}$$

which is essentially a non-convex QCQP. Besides, f_{OPT}^* denotes the optimal solution of (10), $\mathbf{M}_0$ can be obtained from $\{\mathbf{q}_{1,l}, \mathbf{q}_{2,l}, d_l\}_{l=1}^N$, and $\mathbf{M}_l, l = 1, ..., N$ are known variables.

Proof. See Appendix A. □

As we see, the SD-WLS is formulated as a typical non-convex QCQPs problem in (10), which is a very general kind of problem that comprises many NP-hard problems. However, it is well-known that this kind of problems can be relaxed to a convex SDP, also known as Shor's relaxation [19]. Here, a crucial first step in deriving an SDP of problem (10) is to observe that

$$\mathbf{x}^T \mathbf{M}_l \mathbf{x} = \mathbf{Tr}(\mathbf{x}^T \mathbf{M}_l \mathbf{x}) = \mathbf{Tr}(\mathbf{M}_l \mathbf{X}), l = 0, ..., N \tag{11}$$

where $\mathbf{Tr}(\cdot)$ denotes the trace of a given matrix, and we define $\mathbf{X} = \mathbf{x}\mathbf{x}^T$. Note that $\mathbf{X} = \mathbf{x}\mathbf{x}^T$ is equivalent to $\mathbf{X}$ being a rank one symmetric positive matrix, then we obtain the following equivalent formulation of problem (10)

$$\begin{aligned} \min_{\mathbf{X},\mathbf{x}} \quad & \mathbf{Tr}(\mathbf{M}_0 \mathbf{X}) \\ \text{s.t.} \quad & \mathbf{Tr}(\mathbf{M}_l \mathbf{X}) = 0 \quad l = 1, \cdots N, \\ & \mathbf{X} \succeq 0, \qquad \mathrm{rank}(\mathbf{X}) = 1. \end{aligned} \tag{12}$$

Here, for symmetric matrix $\mathbf{X}$, $\mathbf{X} \succeq 0$ means that $\mathbf{X}$ is a positive semidefinite matrix. As we see, we obtain an additional property that the objective and constraints are affine in $\mathbf{X}$ and $\mathbf{x}$, except the last constraint $\mathrm{rank}(\mathbf{X}) = 1$, which is nonconvex. When we drop the rank constraint, we get a convex relaxation. By rewriting it using the Schur complement, we obtain an SDP relaxation:

$$\begin{aligned} f_{SDP}^* = \min_{\mathbf{X},\mathbf{x}} \quad & \mathbf{Tr}(\mathbf{M}_0 \mathbf{X}) \\ \text{s.t.} \quad & \mathbf{Tr}(\mathbf{M}_l \mathbf{X}) = b_l \quad l = 1, \cdots N, \\ & \mathbf{X} \succeq 0. \end{aligned} \tag{13}$$

where f_{SDP}^* denotes the solution of (13). Its dual problem is [20]

$$\begin{aligned} \max_{\boldsymbol{\nu}} \quad & \mathbf{b}^T \boldsymbol{\nu} \\ \text{s.t.} \quad & \mathbf{M}_0 - \sum_{l=1}^{N} \nu_l \mathbf{M}_l \succeq 0, \end{aligned} \tag{14}$$

where $b = [b_1, ..., b_N]$, and $\boldsymbol{\nu} = [\nu_1, ..., \nu_N]$ is the Lagrange multiplier. Globally optimal solution to the above equations (13 and 14) may be found by available numerical algorithms in polynomial time (often by interior-point methods), and many solvers can achieve this, such as SEDUMI and SDPT3.

B. Tightness of SDP Relaxation

In this subsection, we are going to prove that, if the in- and ex- sensors' observations are noise free or corrupted by small enough noise, the SDP relaxation is tight.

As a relaxation, the problem above fulfills

$$f_{SDP}^* \leq f_{OPT}^*. \tag{15}$$

Theoretically, if the SDP relaxation is tight ($f_{SDP}^* = f_{OPT}^*$) and (10) features a unique global minimum, we can easily obtain the guaranteed optimal solution $\mathbf{x}^*$ to the original problem (10) from the optimal solution $\mathbf{Z}^*$ to the SDP relaxation (13). As we proved in Lemma.1, the SDP relaxation is tight if the observations $\{d_l, \mathbf{q}_{1,l}, \mathbf{q}_{2,l}\}_{l=1}^N$ are noise free or corrupted by small enough noise.

Lemma 1. *If the observations $\{d_l, \mathbf{q}_{1,l}, \mathbf{q}_{2,l}\}_{l=1}^N$ are noise free or corrupted by small enough noise, the SDP relaxation (13) for the QCQP problem is tight, or in other words, the optimum of the original QCQP can be found by optimizing its SDP relaxing, i.e., $f_{SDP}^* = f_{OPT}^*$.*

Proof. See Appendix B. □

C. Recovery of Original Problem

Based on above, we have fully introduced our method to obtain the solution of QCQP via SDP relaxation when it is tight. However, after obtaining the solution $\mathbf{X}^*$ in (13), we still encounter another difficulty: It is a classical results related to SDP relaxation that whether the tight SDP solution $\mathbf{X}^*$ fulfills $\mathbf{rank}(\mathbf{X}^*) = 1$ and we may recover $\mathbf{x}^*$ from the low rank decomposition $\mathbf{X}^* = \mathbf{x}^*(\mathbf{x}^*)^T$. Here, [21] provides an efficient solution, where we can use SVD to decompose $\mathbf{X}^*$ and set all other singular value to be zero except for the largest one. Through this way, we will get a very accurate approximation whose rank is close to rank 1.

IV. Maximum Likelihood Estimation based on Weighted Least Square

In this section, we are going to use the estimates provided by the SDP relaxation outlined in Section III as an initial guess in a WLS algorithm to get the MLE of the relative pose. In the process of achieving this, the uncertainty in robot pose in model (1) is also considered, which makes our algorithm more practical in real applications.

First, the same as (7), we stack those equations for $l = 1, .., N$, then we have

$$\mathbf{z} = \mathbf{h}_d + \mathbf{v}, \tag{16}$$

where $\mathbf{z} = [z_1, ..., z_N]^T$, $\mathbf{h}_d = [\sqrt{\mathbf{w}_1^T\mathbf{w}_1}, ..., \sqrt{\mathbf{w}_N^T\mathbf{w}_N}]^T$. And $\mathbf{v} = [v_1, ..., v_N]^T$ is the measurement noise, which is modeled as zero-mean Gaussian with covariance $\mathbf{R} = \sigma_l^2\mathbf{I}$.

Then, we consider to incorporate the robot pose uncertainty $q_{1,l}$ and $q_{2,l}$ in model (17). Therefore, we linearize the measurement equations around the current estimate of the robot-to-robot pose, as well as the individual robot poses, and this corresponds to the linearization of the measurement model

$$\begin{aligned}\mathbf{z} &= \mathbf{h}_d + \mathbf{v} \\ &\simeq \begin{bmatrix} \mathbf{H}_{\mathbf{x}_\mathrm{s}} & \mathbf{H}_{\mathbf{x}_\mathrm{r}} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{x}}_\mathrm{s} \\ \tilde{\mathbf{x}}_\mathrm{r} \end{bmatrix} + \mathbf{H}_{\mathbf{q}_1}\tilde{\mathbf{q}}_1 + \mathbf{H}_{\mathbf{q}_2}\tilde{\mathbf{q}}_2 + \mathbf{v},\end{aligned} \tag{17}$$

where $\mathbf{x}_\mathrm{s}$ and $\mathbf{x}_\mathrm{r}$ is defined by the partial unknown variables of $\mathbf{x}$, which is given by

$$\mathbf{x}_\mathrm{s} = \begin{bmatrix} \cos\theta & \sin\theta \end{bmatrix}^T, \quad \mathbf{x}_\mathrm{r} = \begin{bmatrix} \cos\phi & \sin\phi \end{bmatrix}^T.$$

Besides, $\mathbf{H}_{\mathbf{x}_\mathrm{s}}$, $\mathbf{H}_{\mathbf{x}_\mathrm{r}}$, $\mathbf{H}_{\mathbf{q}_1}$ and $\mathbf{H}_{\mathbf{q}_2}$ are Jacobians of the measurement function $\mathbf{h}_d$ with respect to the corresponding variables, which are obtained using the chain rule as follows:

$$\begin{aligned}\mathbf{H}_{\mathbf{x}_\mathrm{s}} &= \frac{\partial \mathbf{h}_d}{\mathbf{x}_\mathrm{s}} = d_0\mathbf{u}, \quad \mathbf{H}_{\mathbf{x}_\mathrm{r}} = \frac{\partial \mathbf{h}_d}{\mathbf{x}_\mathrm{r}} = \mathbf{u}\mathbf{q}_2\mathbf{C}_{\mathbf{x}_\mathrm{r}}, \\ \mathbf{H}_{\mathbf{q}_1} &= \frac{\partial \mathbf{h}_d}{\partial \mathbf{q}_1} = -\mathbf{u}, \quad \mathbf{H}_{\mathbf{q}_2} = \frac{\partial \mathbf{h}_d}{\partial \mathbf{q}_2} = \mathbf{u}\mathbf{C}.\end{aligned} \tag{18}$$

where $\mathbf{u} = [\mathbf{w}_1^T/d_1, ..., \mathbf{w}_N^T/d_N]^T$, and

$$\mathbf{C}_{\mathbf{x}_\mathrm{r}} = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 1 & 0 \end{bmatrix}^T.$$

In addition, $\mathbf{x}_\mathrm{s}$, $\mathbf{x}_\mathrm{r}$, $\mathbf{q}_1$ and $\mathbf{q}_2$ are given by the following equations:

$$\begin{aligned}\mathbf{x}_\mathrm{s} &= \hat{\mathbf{x}}_\mathrm{s} + \tilde{\mathbf{x}}_\mathrm{s}, \quad \mathbf{x}_\mathrm{r} = \hat{\mathbf{x}}_\mathrm{r} + \tilde{\mathbf{x}}_\mathrm{r} \\ \mathbf{q}_1 &= \hat{\mathbf{q}}_1 + \tilde{\mathbf{q}}_1, \quad \mathbf{q}_2 = \hat{\mathbf{q}}_2 + \tilde{\mathbf{q}}_2\end{aligned} \tag{19}$$

where we denote estimated quantities by "$\hat{}$", and errors by "$\tilde{}$". Besides, $\delta\phi$ denotes the small rotation of ϕ, and $\tilde{\mathbf{x}}_\mathrm{r} = \begin{bmatrix} 1 & \delta\phi \end{bmatrix}^T$. Then, we compute an augmented noise matrix

$$\mathbf{W} = \mathbf{R} + \mathbf{H}_{\mathbf{q}_1}\mathbf{P}_{\mathbf{q}_1}\mathbf{H}_{\mathbf{q}_1}^T + \mathbf{H}_{\mathbf{q}_2}\mathbf{P}_{\mathbf{q}_2}\mathbf{H}_{\mathbf{q}_2}^T \tag{20}$$

where $\mathbf{P}_{\mathbf{q}_1}$ and $\mathbf{P}_{\mathbf{q}_2}$ denote the covariance of $\mathbf{q}_1$ and $\mathbf{q}_2$, and use $\mathbf{W}$ as weighting matrix in the WLS cost function. Note that the DR errors for the same robot will be correlated, but those between two different robots will be independent.

We find the correction by solving the weighted normal equations

$$\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H} \begin{bmatrix} \tilde{\mathbf{x}}_\mathrm{s} \\ \tilde{\mathbf{x}}_\mathrm{r} \end{bmatrix} = \mathbf{H}^T\mathbf{W}^{-1}(\mathbf{z} - \hat{\mathbf{z}}), \tag{21}$$

where $\mathbf{H} = [\mathbf{H}_{\mathbf{x}_\mathrm{s}} \quad \mathbf{H}_{\mathbf{x}_\mathrm{r}}]$. We update the $\hat{\mathbf{x}}_\mathrm{s}$ and $\hat{\mathbf{x}}_\mathrm{r}$ with

$$\begin{aligned}\hat{\mathbf{x}}_\mathrm{s}^{j+1} &= \hat{\mathbf{x}}_\mathrm{s}^{j} + \tilde{\mathbf{x}}_\mathrm{s}, \\ \hat{\mathbf{x}}_\mathrm{r}^{j+1} &= \hat{\mathbf{x}}_\mathrm{r}^{j} + \tilde{\mathbf{x}}_\mathrm{r},\end{aligned} \tag{22}$$

and converge to an optimal solution after several iterates, and j is the index of iteration. Moreover, the final covariance of the estimate is given by

$$\mathbf{E}\left[\begin{bmatrix} \tilde{\mathbf{x}}_\mathrm{s} \\ \tilde{\mathbf{x}}_\mathrm{r} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{x}}_\mathrm{s} \\ \tilde{\mathbf{x}}_\mathrm{r} \end{bmatrix}^T \right] = (\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H})^{-1}. \tag{23}$$

Once the pose estimate and its covariance are computed, additional distance measurements can be processed in a recursive estimator, such as extended Kalman filter and particle filter.

V. Simulation and Experiment Results

In this section, comprehensive simulations and well-designed experiments (with real data processing results) are presented for validating the effectiveness of the proposed algorithm. In particular, measurements with almost negligible and sufficiently large noise are both discussed, and the performance (e.g., estimation accuracy) of the deterministic and SD-WLS methods are compared with the proposed SDP relaxation method. Moreover, the efficiency (such as the improvement in estimation, fast convergence speed) of the WLS method is highlighted by using the estimates of the discussed methods as initialization guess.

A. Simulations

For the results shown in the simulation part, the trajectories and distance measurements are generated as follows: (1) The two robots start at initial positions $10\mathrm{m}$ apart from each other and record their first distance measurement; (2) each robot moves randomly for approximately $2\mathrm{m}$; and (3) the robots record a distance measurement at their new positions, and repeat (2) and (3) for four times. The odometry measurement

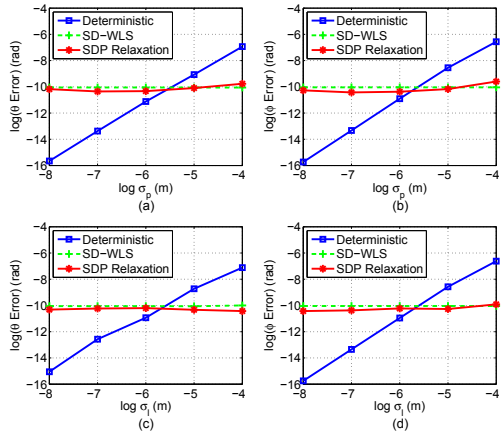

Fig. 2. Small noise case: A comparison of the three methods with different noise versus θ, ϕ error: (a) $\log(\sigma_p)$ versus $\log(\theta)$ error. (b) $\log(\sigma_p)$ versus $\log(\phi)$ error. (c) $\log(\sigma_l)$ versus $\log(\theta)$ error. (d) $\log(\sigma_l)$ versus $\log(\phi)$ error.

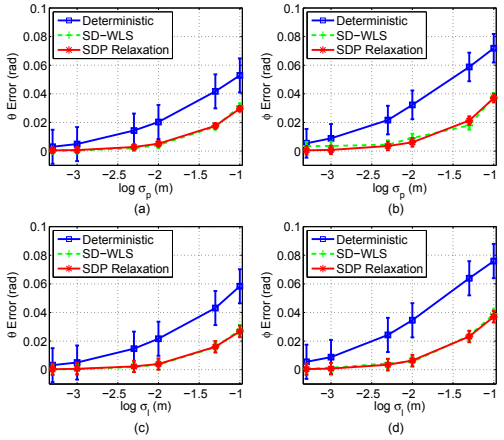

Fig. 3. Large noise case: A comparison of the three methods with different noise versus θ, ϕ error: (a) $\log(\sigma_p)$ versus θ error. (b) $\log(\sigma_p)$ versus ϕ error. (c) $\log(\sigma_l)$ versus θ error. (d) $\log(\sigma_l)$ versus ϕ error.

is modeled based on (19), and its noise follows a zero-mean Gaussian distribution, with covariance $\mathbf{Q} = \sigma_p^2\mathbf{I}$. The distance measurement is generated by (17), and the covariance of its noise is assumed to be $\mathbf{R} = \sigma_l^2\mathbf{I}$. Note that both σ_l and σ_p are used as parameters to examine the accuracy of the presented algorithms, and we conducted 1000 trails per setting.

Firstly, we are devoted to examine the performance of the different algorithms when the measurement noise is relatively small. As shown in Fig. 2, a comparison of the deterministic, SD-WLS and the proposed SDP relaxation is provided, where σ_p and σ_l vary from 10^{-8}m to 10^{-4}m. In Fig. 2, we have taken a combination of the parameters σ_p, σ_l with the estimation results θ and ϕ in (a)-(d), respectively. Since these parameters are small, they are denoted by a logarithmic form. As we see, the deterministic algorithm performs a little bit better than the SD-WLS and the SDP relaxation methods when the measurement noise is within

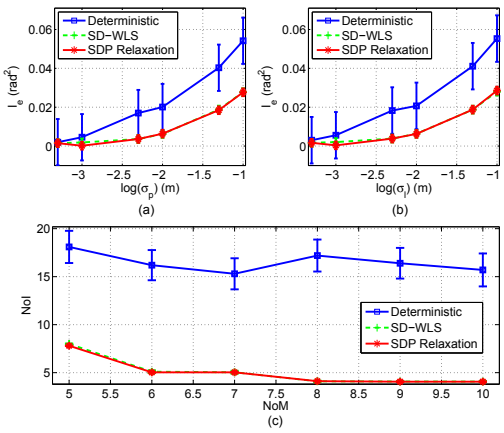

Fig. 4. The improvement of WLS in terms of estimation accuracy and convergence speed: (a) σ_p versus I_e. (b) σ_l versus I_e. (c) NoM versus NoI.

the scale of $[10^{-8}, 2\cdot 10^{-6}]$m. This is possibly because of the suboptimal characteristic of the SD-WLS and the SDP relaxation for the assumption on Guassian noise in (4). However, since the error is extremely approaching zero, we ignore the gap between the three methods here. Then, once the noise deviation exceeds $2\cdot 10^{-6}$m, we will find that the SD-WLS and the SDP relaxation methods consistently outperform the deterministic algorithm, and the error for the deterministic algorithm grows exponentially with the increase of the noise standard deviation, which increases much faster than the (linear growing) SD-WLS and SDP relaxation algorithm. Another thing has to be mentioned is that the estimation error of θ and ϕ for both SD-WLS and SDP relaxation methods achieve approximately the same estimation accuracy[2] and close to zero, which verifies our conclusion in Lemma. 1.

Next, we aim to find out whether the our algorithm still performs well when the measurement noise is large. To this end, we provide another simulation in Fig. 3, where the measurement noise deviation is changed from 10^{-4}m to 10^{-1}m. Besides, we didn't use the logarithmic form of the estimation error on parameter θ, ϕ since they are large enough, while other settings are kept the same with the previous small noise case. As we see in Fig. 3, even though the measurement noise deviation is set to be sufficiently large, like 10^{-1}m, the SDP relaxation method still can produce a good performance (the θ and ϕ both less than 0.04rad) on the estimation of θ, ϕ (The estimation variance is also plotted), which to some extend support a guess that the SDP relaxation is still tight when the measurement noise is large.

By using the outputs of the deterministic, SD-WLS, and SDP relaxation algorithms as initialization points, the performance of the WLS algorithm is investigated. The results show that the WLS indeed has an improvement on the

[2]Start from here, we will no longer give a specific assertion if the SD-WLS and SDP relaxation methods have the same performance on other indicators since they are always consistent.

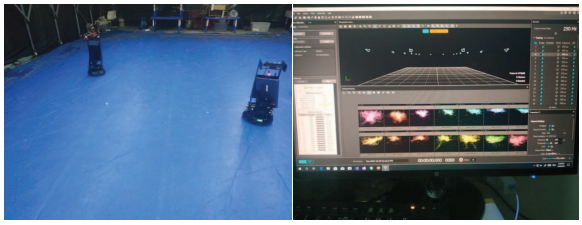

Fig. 5. (a) Two robots moving in $4\mathrm{m} \times 6\mathrm{m}$ wide arena, and each robot travels with an average velocity of around 0.1m/s. (b) The computer interaction platform of the motion capture system.

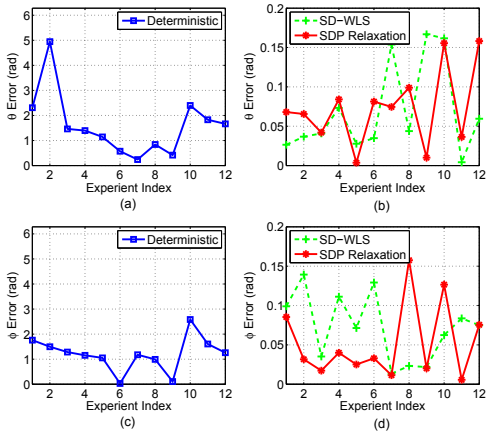

Fig. 6. A comparison of the three methods in terms of estimation accuracy based on real data: (a) Experiment Index versus θ error w.r.t. deterministic method. (b) Experiment Index versus θ error w.r.t. SD-WLS and SDP relaxation. (c) Experiment Index versus ϕ error w.r.t. deterministic method. (d) Experiment Index versus ϕ error w.r.t. SD-WLS and SDP relaxation.

initialization points of the deterministic, SD-WLS and SDP relaxation algorithms. Meantime, the SD-WLS and SDP relaxation algorithms have faster convergence speed. Here, we define $\tilde{\mathbf{x}}_{\mathrm{WLS}}$ as the error of the estimation after the WLS refinement is finalized with the discussed algorithms, thus the improvement on the error is defined by

$$\mathrm{I}_e = \|\tilde{\mathbf{x}}_{\mathrm{WLS}} - \mathbf{x}\|_2. \tag{24}$$

In Fig. 4(a) and Fig. 4(b), we find that the WLS provides an appealing error refinement, and the refinement performance becomes better with the increase of the measurement noise. Especially, there is a huge reduction in the error (around 50%) when measurement noise is large, which means that the WLS has a good capacity to deal with the large noise case. Additionally, the convergence speed (indicated by the number of iteration (NoI)) is also compared. As we know, 5 distance measurements are capable of getting a unique solution of the initial pose in a 2D plane. Therefore, in Fig. 4(c), we consider that the number of measurements (NoM) increases from $N = 5$ to $N = 10$, and the corresponding iteration time of the algorithms are presented. As we see, since the SD-WLS and SDP relaxation algorithms have much better initialization value, thus the convergence speed

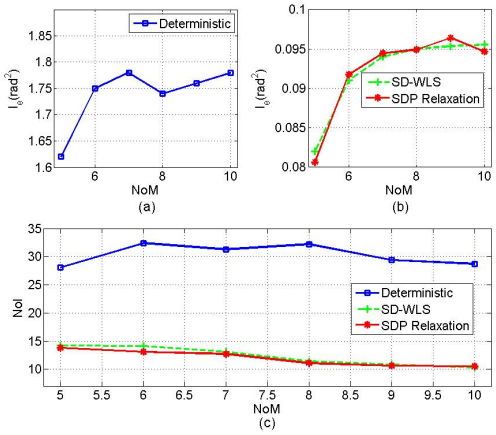

Fig. 7. The improvement of WLS with real data in terms of estimation accuracy and convergence speed: (a) σ_p versus I_e. (b) σ_l versus I_e. (c) NoM versus NoI.

is around 2.5 times faster than the deterministic method. Besides, for SD-WLS and SDP relaxation initialization, there is a small improvement of the convergency speed with the increase of the NoM.

B. Real-Data Experiments

In addition to simulations, we have tested the SDP relaxation-based algorithm in real-world experiments, and the performance is also compared with the deterministic and SD-WLS algorithms, where the experiment results strongly support the conclusion of the simulations. In the experiment part, DR and distance measurements were recorded, and the data is further processed in Matlab2013a on a Intel core i7 with a 3.6-GHz processer.

The 2D experiment is carried out using two-wheeled robots, where IMU sensor, wheel encoder, and UWB module (DWM1000) are installed on the bottom of each robot, where DR and ranging information can be obtained from these sensors. Here, the UWB module with up to 30 meters' ranging capability and programmed to send the ranging measurements with 1Hz frequency (For simplicity, though the DR data has a rate of $50\mathrm{Hz}$, we record the data at every second.). The STM32F103 discovery board serves as host controllers to the DWM1000. As shown in Fig. 5(a), the proposed SDP relaxation and WLS algorithms have been tested on two robots moving in $4\mathrm{m}\times6\mathrm{m}$ wide arena, and each robot travels with an average velocity of around 0.1m/s. The ground truth is established by a calibrated motion capture system, and a computer interaction platform of the motion capture system (with 16 calibrated cameras) is provided in Fig. 5(b). The distance and angle error is limited less than $0.05\mathrm{mm}$ and 0.01°, respectively.

To show the effectiveness of our initialization strategy, we fix $\phi = 45.23^\circ$ first. Note that this is achieved by manually adjusting the two robots, and the specific orientation value is obtained from the motion capture system. Then, we conducted 12 experiments with a set of predefined θ, which is shown in TABLE I. Then, similarly, we fix $\theta = 44.99^\circ$, and

TABLE I
EXPERIMENT INDICES AND CORRESPONDING PARAMETERS

Index	1	2	3	4	5	6	7	8	9	10	11	12
θ	$-150.66°$	$-120.76°$	$-88.64°$	$-60.25°$	$-32.76°$	$0.002°$	$30.23°$	$60.16°$	$90.1869°$	$120.64°$	$149.92°$	$179.95°$
Index	1	2	3	4	5	6	7	8	9	10	11	12
ϕ	$-147.26°$	$-119.25°$	$-90.22°$	$-60.16°$	$-31.11°$	$3.68°$	$32.56°$	$59.26°$	$89.12°$	$120.61°$	$150.23°$	$179.41°$

tried another 12 different experiments, and different values of the parameter θ are also given in TABLE I. Accordingly, with real data, a comparison of the SDP relaxation algorithm with the deterministic and SD-WLS method is provided in Fig. 6. We have also taken a combination of the parameter σ_p and σ_l with the estimation error of θ and ϕ in (a)-(d). As we see, in a practical application, the deterministic method has an unbearable estimation error with θ or ϕ, which means that it nearly lost the capacity to obtain a correct estimation of parameters. In contrast, the SD-WLS and our SDP relaxation still have a small estimation error with θ and ϕ, where their efficiency is well exhibited. Further, similar to the simulation part, we are devoted to demonstrating that the WLS algorithm has an improvement on the initialization points of the deterministic, SD-WLS, and SDP relaxation algorithms. As shown in Fig. 7, instead of choosing different σ_p and σ_l, we consider that NoM increases from $N = 5$ to $N = 10$, and the estimation error of θ and ϕ are presented. We see that even with the real data, the WLS also provides an appealing refinement in Fig. 7(a) and Fig. 7(b), and the error refinement performance becomes better with the increase of the NoM. Also, even though more iteration time is required for all discussed algorithms in a practical application, Fig. 7(c) consistently shows that the SD-WLS and SDP relaxation algorithms indeed have a faster convergence speed (also around 2.5 times) since they have a better initialization value, and the increase of the NoM shows only a little influence on convergency speed of WLS algorithms, where these conclusions are in accord with the conclusion in the simulation part.

VI. CONCLUSIONS AND FUTURE WORK

The 2D robot-to-robot relative pose estimation problem is considered in this paper, and an SDP optimization method is intensively investigated. Specifically, we prove that a state-of-the-art method SD-WLS can be reformulated as a non-convex QCQP problem without any loss. To handle its non-convex nature, an SDP relaxation optimization-based method is proposed, and we prove the relaxation is theoretically tight when the measurements are free from noise or just corrupted by small noise. We also developed a WLS method to refine the estimate from the SDP optimization, which is theoretically optimal in the sense of MLE. Simulations and real data processing results reveal that the SDP relaxation is tight with small noise and performs well even when the measurement noise is sufficiently large. Besides, the efficiency of the WLS algorithm is highlighted. The future work will focus on extending the SDP relaxation method to a 3D case and test it on a multi-UAV system and our Freeform Robot (FreeBOT) [22].

APPENDIX A

According to equation (2) and (5), it is easy to have

$$\mathbf{w}_l^T\mathbf{w}_l - z_l^{'} = \varepsilon_l + 2(\mathbf{p}-\mathbf{q}_{1,l})^T\mathbf{C}\mathbf{q}_{2,l} - 2\mathbf{q}_{1,l}^T\mathbf{p}, \quad (25)$$

where $\varepsilon_l := d_0^2 + \mathbf{q}_{1,l}^T\mathbf{q}_{1,l} + \mathbf{q}_{2,l}^T\mathbf{q}_{2,l} - d_l^2$. Note that ε_l is assumed to be known since d_0, $\mathbf{q}_{1,l}$ and $\mathbf{q}_{2,l}$ for $l = 1, ..., N$ can be obtained from in- and ex- sensors. Then we substitute (3) into (25), we will have

$$\begin{aligned}
&\varepsilon_l + 2(\mathbf{p}-\mathbf{q}_{1,l})^T\mathbf{C}\mathbf{q}_{2,l} - 2\mathbf{q}_{1,l}^T\mathbf{p} \\
&= \varepsilon_l + 2(d_0\begin{bmatrix}\mathrm{c}\theta\\ \mathrm{s}\theta\end{bmatrix} - \begin{bmatrix}x_{1,l}\\ y_{1,l}\end{bmatrix})^T\begin{bmatrix}\mathrm{c}\phi & -\mathrm{s}\phi\\ \mathrm{s}\phi & \mathrm{c}\phi\end{bmatrix}\begin{bmatrix}x_{2,l}\\ y_{2,l}\end{bmatrix} \\
&\quad - 2d_0\begin{bmatrix}x_{1,l}\\ y_{1,l}\end{bmatrix}^T\begin{bmatrix}\mathrm{c}\theta\\ \mathrm{s}\theta\end{bmatrix} \\
&= \varepsilon_l + 2d_0(x_{2,l}\mathrm{c}\theta\mathrm{c}\phi + x_{2,l}\mathrm{s}\theta\mathrm{s}\phi - y_{2,l}\mathrm{c}\theta\mathrm{s}\phi + y_{2,l}\mathrm{s}\theta\mathrm{c}\phi) \\
&\quad - 2(x_{1,l}x_{2,l}\mathrm{c}\phi + y_{1,l}x_{2,l}\mathrm{s}\phi - x_{1,l}y_{2,l}\mathrm{s}\phi + y_{1,l}y_{2,l}\mathrm{c}\phi) \\
&\quad - 2d_0(x_{1,l}\mathrm{c}\theta + y_{1,l}\mathrm{s}\theta) \\
&= -2d_0x_{1,l}\mathrm{c}\theta - 2d_0y_{1,l}\mathrm{s}\theta - 2(x_{1,l}x_{2,l} + y_{1,l}y_{2,l})\mathrm{c}\phi \\
&\quad + 2(x_{1,l}y_{2,l} - y_{1,l}x_{2,l})\mathrm{s}\phi + 2d_0x_{2,l}\mathrm{c}(\theta-\phi) \\
&\quad + 2d_0y_{2,l}\mathrm{s}(\theta-\phi) + \varepsilon_l
\end{aligned} \quad (26)$$

where $\mathrm{c}\theta = \cos(\theta)$, $\mathrm{s}\theta = \sin(\theta)$, $\mathrm{c}\phi = \cos(\phi)$ and $\mathrm{s}\phi = \sin(\phi)$. Next, in order to simplify (26), we define

$$\begin{aligned}
\boldsymbol{\chi}_l &= \begin{bmatrix} -2(x_{1,l}x_{2,l} + y_{1,l}y_{2,l}) & 2(x_{1,l}y_{2,l} - y_{1,l}x_{2,l})\end{bmatrix}^T, \\
\mathbf{x} &= \begin{bmatrix}\mathrm{c}\theta & \mathrm{s}\theta & \mathrm{c}\phi & \mathrm{s}\phi & \mathrm{c}(\theta-\phi) & \mathrm{s}(\theta-\phi) & 1\end{bmatrix}^T.
\end{aligned} \quad (27)$$

Substitute (27) into (26), thus

$$\mathbf{w}_l^T\mathbf{w}_l - z_l^{'} = \begin{bmatrix} -2d_0\mathbf{q}_{1,l}^T & \boldsymbol{\chi}_l & 2d_0\mathbf{q}_{2,l}^T & \varepsilon_l\end{bmatrix}\mathbf{x} \quad (28)$$

Next, the same as (7), we stack N equations of (28), hence

$$\mathbf{h}_{sd} - \mathbf{z}^{'} = \mathbf{A}\mathbf{x} \quad (29)$$

where

$$\mathbf{A} = \begin{bmatrix} -2d_0\mathbf{q}_{1,1}^T & \boldsymbol{\chi}_1^T & 2d_0\mathbf{q}_{2,1}^T & \varepsilon_1 \\ \vdots & \vdots & \vdots & \vdots \\ -2d_0\mathbf{q}_{1,N}^T & \boldsymbol{\chi}_N^T & 2d_0\mathbf{q}_{2,N}^T & \varepsilon_N\end{bmatrix}. \quad (30)$$

Therefore

$$\frac{1}{2}(\mathbf{h}_{sd} - \mathbf{z}^{'})^T\mathbf{R}^{'-1}(\mathbf{h}_{sd} - \mathbf{z}^{'}) = \frac{1}{2}\mathbf{x}^T\mathbf{A}^T\mathbf{R}^{'-1}\mathbf{A}\mathbf{x}. \quad (31)$$

Here, we define $\mathbf{M}_0 = \mathbf{A}^T\mathbf{R}'\mathbf{A}$. Additionally, since $\mathbf{x}$ introduces several new variables, so we have

$$\begin{aligned}
&c^2\phi + s^2\phi - 1 = 0 \Rightarrow x_1^2 + x_2^2 = 1,\\
&c^2\theta + s^2\theta - 1 = 0 \Rightarrow x_3^2 + x_4^2 = 1,\\
&c^2(\theta-\phi) + s^2(\theta-\phi) - 1 = 0 \Rightarrow x_5^2 + x_6^2 = 1,\\
&c\theta c\phi + s\theta\phi - c(\theta-\phi) = 0 \Rightarrow x_1x_3 + x_2x_4 - x_5 = 0,\\
&s\theta c\phi - c\theta s\phi - c(\theta-\phi) = 0 \Rightarrow x_1x_4 - x_2x_3 - x_6 = 0.
\end{aligned} \tag{32}$$

Then, (32) can be easily written as

$$\mathbf{x}^T\mathbf{M}_l\mathbf{x} = b_l \quad l = 1, 2, .., N, \tag{33}$$

where $\mathbf{M}_l$ is a typical sparse matrix. To write $\mathbf{M}_l$ in a compact form, we brought the notation style from matlab, where the notation $\mathrm{sparse}(i, j, s, m, n)$ is used to represent each $\mathbf{M}_l$, where i and j denote the row and column indexes of $\mathbf{M}_l$, respectively, and s represents the corresponding values of $\mathbf{M}_l$ at index (i, j). Besides, m and n denote the size of the matrix $\mathbf{M}_l$. In particular, they are

$$\begin{aligned}
\mathbf{M}_1 &= \mathrm{sparse}([1,2],[1,2],[1,1],7,7),\\
\mathbf{M}_2 &= \mathrm{sparse}([3,4],[3,4],[1,1],7,7),\\
\mathbf{M}_3 &= \mathrm{sparse}([5,6],[5,6],[1,1],7,7),\\
\mathbf{M}_4 &= \mathrm{sparse}([1,2,7],[3,4,5],[1,1,-1],7,7),\\
\mathbf{M}_5 &= \mathrm{sparse}([1,2,7],[4,3,6],[1,-1,-1],7,7),
\end{aligned} \tag{34}$$

and $\mathbf{b} = [b_1, ..., b_S]^T = [1, 1, 1, 0, 0]$. Therefore, based on (31)-(34), the SD-WLS in (9) can be equivalently formulated as the following

$$\begin{aligned}
\min_{\mathbf{x}} \quad & \mathbf{x}^T\mathbf{M}_0\mathbf{x}\\
\text{s.t.} \quad & \mathbf{x}^T\mathbf{M}_l\mathbf{x} = b_l, l = 1, .., S,
\end{aligned}$$

which is a typical QCQP problem.

APPENDIX B

Let $\overline{\mathbf{x}}$ denote the ground truth of the original problem (10) and assume $\boldsymbol{\nu} = \mathbf{0}$ in (14). Then $\overline{\mathbf{x}}$ is optimal to (10) and (14), respectively. Obviously, $\overline{\mathbf{x}}$ and $\boldsymbol{\nu}$ satisfy the following three conditions:

- Primal feasibility. Substituting $\overline{\mathbf{x}}$ in the original problem (10), the constraints are satisfied since $\overline{\mathbf{x}}$ is the ground truth of the original problem, which means that $\mathbf{x}^T\mathbf{M}_l\mathbf{x} = b_l$ is tenable for all l.
- Dual feasibility. Since $\{d_l, \mathbf{q}_{1,l}, \mathbf{q}_{2,l}\}_l^N$ are noise free, thus $\nu_l = 0$ or $\nu_l \approx 0$ for $l = 1, ..., N$ in (14). Further, based on (30) and $\mathbf{M}_0 = \mathbf{A}^T\mathbf{R}'\mathbf{A}$, we may know that $M_{\boldsymbol{\nu}} = \mathbf{M}_0 = \overline{\mathbf{M}}_0 \succeq 0$.
- Lagrange multiplier. Since $\nu_l = 0$ or $\nu_l \approx 0$ for $l = 1, ..., N$ when measurements are noise free, hence $\mathbf{M}_{\boldsymbol{\nu}}\overline{\mathbf{x}} = \overline{\mathbf{M}}_0\overline{\mathbf{x}}$. Meantime, it is easy to find that $\overline{\mathbf{x}}^T\mathbf{M}_0\overline{\mathbf{x}} = 0$ in (10). We combine the two conditions, and using the Lemma 3. in [23], thus $\mathbf{M}_{\boldsymbol{\nu}}\overline{\mathbf{x}} = 0$.

As pointed out by the Lemma. 2.4 in [24], as long as the above three conditions are satisfied, then we can recover the minimizer of the QCQP from the SDP without any loss.

REFERENCES

[1] A. Franchi, G. Oriolo, and P. Stegagno. "Mutual localization in multi-robot systems using anonymous relative measurements". *Int. J. of Robot. Res.*, 32(11):1302–1322, 2013.

[2] K. X. Guo, Z. R. Qiu, W. Meng, L. H. Xie, and R. Teo. "Ultra-wideband based cooperative relative localization algorithm and experiments for multiple unmanned aerial vehicles in GPS denied environtments". *Int. J. Micro Air Veh.*, 9(3):169–186, 2017.

[3] H. B. Luo, M. Li, G. Q. Liang, H. H. Qian, and T. L. Lam. "An obstacle-crossing strategy based on the fast self-reconguration for modular sphere robots". In *IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Las Vegas, USA, 2020.

[4] M. Mazo, A. Speranzon, K. H. Johansson, and X. M. Hu. "Muti-robot traking of a moving object using directional sensors". In *IEEE. Int. Conf. Robot. Autom.*, pages 1103–1108, New Orleans, LA, 2004.

[5] S. I. Roumeliotis and G. A. Bekey. "Distributed multirobot localization". *IEEE Trans. Robot. Autom.*, 18(5):781–795, 2002.

[6] A. Howard, L. E. Parker, and G. S. Sukhatme. "Experiments with a large heterogeneous mobile robot team: Exploration, mapping, deployment and detection". *Int. J. Robot. Res.*, 25(5-6):431–447, 2006.

[7] L. Doherty, K.S. J. Pister, and L. E. Ghaoui. "Convex position estimation in wireless sensor networks". In *Proc. Infocom Conf.*, pages 1655–1663, Anchorage, AK, 2001.

[8] J. Nie. "Sum of squares method for sensor network localization". *Computational Optimization and Applications*, 43(2):1573–2894, 2009.

[9] Y. Shang, W. Ruml, Y. Zhang, and M. P. J. Fromherz. "Localization from mere connectivity". In *Proceedings of ACM MobileHoc*, pages 201–212, Annapolis, MD, 2003.

[10] G. L. Marriottini, G. Papps, D. Prattichizzo, and K. Daniilidis. "Vision-based localization of leader-follower formations". In *Proc. 44th IEEE. Conf. Decision Control, Eur. Control Conf.*, pages 635–640, Seville, Spain, 2005.

[11] A. Martinelli and R. Siegwart. "Observability analysis for mobile robot localization". In *IEEE/RSJ Int. Conf. Intell. Robots Syst.*, pages 1471–1476, Edmonton, AB, Canada, 2005.

[12] Z. M. Han, K. X. Guo, and Z. Y. Lin. "Integrated relative localization and leader-follower formation control". *IEEE Trans. Autom. Control*, 64(1):20–34, 2018.

[13] X. S. Zhou and S. I. Roumeliotis. "Multi-robot SLAM with unknown initial correspondence: The robot rendezvous case". In *IEEE/RSJ Int. Conf. Intell. Robots Syst.*, pages 9–15, Beijing, China, 2006.

[14] X. S. Zhou and S. I. Roumeliotis. "Observability analysis for mobile robot localization". In *IEEE/RSJ Int. Conf. Intell. Robots Syst.*, pages 1471–1476, Edmonton, Canada, 2005.

[15] X. Zhou and S. Roumeliotis. "Robot-to-robot relative pose estimation from range measurements". *IEEE. Trans. Robot.*, 24(6):1379–1393, 2008.

[16] N. Trawny, X. S. Zhou, K. X. Zhou, and S. I. Roumeliotis. "3D relative pose estimation from distance-only measurements". In *IEEE/RSJ Int. Conf. Intell. Robots Syst.*, pages 1071–1078, San Diego, CA, 2007.

[17] N. Trawny and S. I. Roumeliotis. "On the global optimum of planar, range-based robot-to-robot relative pose estimation". In *IEEE. Int. Conf. Robot. Autom.*, pages 3200–3206, Anchorage, Alask, USA, 2010.

[18] N. Trawny. *"Cooperative localization: On motion-induced initialization and joint state estimation under communication constraint"*. PhD thesis, University of Minnesota, 2010.

[19] J. Park and S. Boyd. "General heuristics for nonconvex quadratically constrained quadratic programming". *arXiv preprint arXiv:1703.07870*, 2017.

[20] Y. C. Ding. "On efficient semidefinite relaxations for quadratically constrained quadratic programming". Master's thesis, University of Waterloo, 2007.

[21] Z. Q. Luo, W. K. Ma, A. M. S. So, Y. Y. Ye, and S. Z. Zhong. "Semidefinite relaxation of quadratic optimization problem". *IEEE Signal Process. Mag.*, 27(3):20–34, 2010.

[22] G. Q. Liang, H. B. Luo, M. Li, H. H. Qian, and T. L. Lam. "Freebot: A freeform modular self-reconfigurable robot with arbitrary connection point - design and implementation". In *IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Las Vegas, USA, 2020.

[23] J. Zhao. "An efficient solution to non-minimal case essential matrix estimation". *arXiv preprint arXiv:1903.09067*, 2019.

[24] D. Cifuentes, S. Agarwal, P. A. Parrilo, and A. R. R. Thomas. "On the local stability of semidefinite relaxations". *arXiv preprint arXiv:1710.04287v2*, 2018.