A Semidefinite Relaxation Method for Source Localization Using TDOA and FDOA Measurements

Gang Wang, Youming Li, and Nirwan Ansari, *Fellow, IEEE*

***Abstract*—Localization by a sensor network has been extensively studied. In this paper, we address the source localization problem by using time-difference-of-arrival (TDOA) and frequency-difference-of-arrival (FDOA) measurements. Owing to the nonconvex nature of the maximum-likelihood (ML) estimation problem, it is difficult to obtain its globally optimal solution without a good initial estimate. Thus, we reformulate the localization problem as a weighted least squares (WLS) problem and perform semidefinite relaxation (SDR) to obtain a convex semidefinite programming (SDP) problem. Although SDP is a relaxation of the original WLS problem, it facilitates accurate estimate without postprocessing. Moreover, this method is extended to solve the localization problem when there are errors in sensor positions and velocities. Simulation results show that the proposed method achieves a significant performance improvement over existing methods.**

***Index Terms*—Frequency difference of arrival (FDOA), localization, semidefinite programming (SDP), sensor network, time difference of arrival (TDOA).**

I. Introduction

THE WIRELESS sensor network has been a hot research topic in recent years [1]–[6]. In particular, localization by a sensor network has attracted much attention since it has found wide applications in many fields, such as surveillance, navigation, target tracking, and others [6]. Among all localization problems, passively locating a source is extremely important in military applications. To passively locate a source with high accuracy, time-difference-of-arrival (TDOA) measurements can be utilized. If there is relative motion between the sensors and the source, frequency-difference-of-arrival (FDOA) measurements can also be utilized to further improve localization performance. Furthermore, TDOAs can only be used to locate a stationary source, i.e., to determine the position of the source. If the source is moving, both TDOAs and FDOAs are to be employed to determine both the position and the velocity of the source. In practice, TDOAs can be measured by mutual correlation of the received signals at different sensors. Hence, TDOAs are particularly useful for localization of high-bandwidth sources. On the other hand, FDOAs are actually the differences in received Doppler frequency offsets; thus, FDOAs are more suitable for localization of low-bandwidth sources. By combining TDOA and FDOA measurements, the source can be accurately located in a wide spectrum of bandwidth.

Unlike TDOA-based localization, which has been extensively studied [7]–[14], the study of TDOA/FDOA-based localization has seldom been reported in the literature. The challenges of locating a source using TDOA/FDOA measurements lie in the high nonlinearity and nonconvexity of the maximum-likelihood (ML) problem and the coupling of the to-be-estimated parameters (i.e., the source position and velocity) in the measurement models. Recently, some effort has been devoted to solve this difficult problem. The traditional solution to this nonlinear ML problem is to iteratively linearize it using Taylor series expansion [15]. However, this method needs an initial estimate, and it cannot guarantee convergence to the global solution of the ML problem. To circumvent this drawback, some closed-form solution methods have been proposed [16]–[19]. Ho *et al.* [16] proposed the well-known two-step weighted least squares (WLS) method, which linearizes the nonlinear measurement equations by introducing two nuisance parameters and solves the subsequent linear equations in the WLS sense. Extensions to this method have been reported in [17] and [18]. In particular, the sensor position and velocity errors are taken into account in [17], and the multiple-source localization problem is studied in [18]. Wei *et al.* [19] proposed a multidimensional scaling (MDS) method, in which the classical MDS framework is extended to be amenable to this particular localization problem. Closed-form solution methods do not have the divergence problem and are very computationally efficient. Moreover, their performance can attain the Cramer–Rao lower bound (CRLB) at sufficiently low noise levels. Recently, a quadratic constraint solution (QCS) method [20] and a semidefinite relaxation (SDR) method [21] have been proposed. The former finds two Lagrange multipliers by Newton's method and uses them to obtain the source position and velocity estimate, whereas the latter relaxes the ML problem to obtain a convex semidefinite programming (SDP) problem and further refines the SDP solution using local search algorithms. Both methods show superior performance over the two-step WLS method. However, current methods usually either cannot obtain

Manuscript received May 2, 2012; revised July 27, 2012; accepted October 3, 2012. Date of publication October 16, 2012; date of current version February 12, 2013. This work was supported in part by the National Natural Science Foundation of China under Grant 61201099 and Grant 61071119, by the Program for the Ningbo Municipal Technology Innovation Team under Grant 2011B81002, and by the K. C. Wong Magna Fund in Ningbo University. The review of this paper was coordinated by Prof. X. Wang.

G. Wang and Y. Li are with the College of Information Sciences and Engineering, Ningbo University, Ningbo 315211, China (e-mail: wanggang@nbu.edu.cn; liyouming@nbu.edu.cn).

N. Ansari is with the Advanced Networking Laboratory, Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102 USA (e-mail: Nirwan.Ansari@njit.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2012.2225074

good performance at high noise levels or have local convergence problems, which may degrade their performance.

Recently, convex relaxation techniques have been applied to many applications in communications and signal processing, such as source and sensor network localization [12], [13], [21]–[23], multiple-input–multiple-output (MIMO) detection [24], [25], joint source and relay power allocation for MIMO relay systems [26], [27], and transmit beamforming [28]. In this paper, the single-source localization problem using line-of-sight TDOA and FDOA measurements is addressed, and the SDR technique is applied to solve this problem. Unlike the SDR method in [21], in which SDR is performed to the original ML formulation, we reformulate this localization problem based on the WLS criterion and solve this WLS problem in two steps. We first approximate the weighting matrix by using an initial estimate of the source position and velocity in the first step and then apply SDR to the approximate WLS problem in the second step. The contributions of this paper include the following.

1) The WLS problem is shown to closely approximate the original ML problem.
2) The proposed SDP solution does not require postprocessing and avoids local convergence.
3) The proposed SDR method has lower complexity than the existing SDR method [21].

The accuracy of the SDP solution is guaranteed due to the facts that 1) the WLS problem is not sensitive to the approximation of the weighting matrix (see the simulation results in Section VI), and 2) SDR is possible to find the solution of the approximate WLS problem, i.e., SDR is tight (see the discussions at the end of Section III-B). Therefore, postprocessing techniques are *not* needed to refine the SDP solution, and local convergence is thus avoided. This is extremely important since local convergence may result in significant performance degradation. Furthermore, the proposed SDR method is extended to the localization case in the presence of errors in sensor positions and velocities.

The remainder of this paper is organized as follows. In Section II, the TDOA and FDOA measurement models are given. Subsequently, the SDR methods for solving the localization problems without and with errors in sensor positions and velocities are presented in Sections III and IV, respectively. Complexity analysis is given in Section V, and simulation results are illustrated in Section VI. Finally, conclusions are drawn in Section VII.

The following notations are adopted throughout this paper. Boldface lowercase letters and boldface uppercase letters denote the vectors and matrices, respectively. $\boldsymbol{a}(i)$ denotes the ith element of vector $\boldsymbol{a}$, and $\boldsymbol{a}(i:j)$ denotes the subvector of $\boldsymbol{a}$ composed by the ith to jth elements of $\boldsymbol{a}$. $\boldsymbol{A}(i,:)$ denotes the ith row of the matrix $\boldsymbol{A}$, $\boldsymbol{A}(i,j)$ denotes the (i,j)th element of $\boldsymbol{A}$, and $\boldsymbol{A}(i:j,k:l)$ denotes the submatrix of $\boldsymbol{A}$ formed by rows i to j and columns k to l. $\mathbf{1}_k$ and $\boldsymbol{I}_k$ denote the $k\times 1$ all-one column vector and the $k\times k$ identity matrix, respectively, and $\mathbf{0}_k$ and $\boldsymbol{O}_{k\times l}$ denote the $k\times 1$ zero column vector and the $k\times l$ zero matrix, respectively. $\mathrm{tr}(\boldsymbol{A})$ means the trace of $\boldsymbol{A}$, and $\boldsymbol{A}\succeq\boldsymbol{B}$ means that $\boldsymbol{A}-\boldsymbol{B}$ is positive semidefinite. $\boldsymbol{x}^o$ ($\dot{\boldsymbol{x}}^o$) and $\boldsymbol{s}_i^o$ ($\dot{\boldsymbol{s}}_i^o$) represent the true positions (velocities) of the source and the ith sensor, respectively. $\boldsymbol{x}$ ($\dot{\boldsymbol{x}}$) represents the unknown source position (velocity) variable in optimization problems, and $\bar{\boldsymbol{s}}_i^o$ ($\dot{\bar{\boldsymbol{s}}}_i^o$) represents the estimated position (velocity) of the ith sensor. To clarify the notations, we use the following rules: The notations $(\star)^o$, $(\star)$, $(\bar{\star})^o$, and $(\bar{\star})$ have the same form, but $(\star)^o$ contains $\boldsymbol{x}^o$ ($\dot{\boldsymbol{x}}^o$) and $\boldsymbol{s}_i^o$ ($\dot{\boldsymbol{s}}_i^o$), $(\star)$ contains $\boldsymbol{x}$ ($\dot{\boldsymbol{x}}$) and $\boldsymbol{s}_i^o$ ($\dot{\boldsymbol{s}}_i^o$), $(\bar{\star})^o$ contains $\boldsymbol{x}^o$ ($\dot{\boldsymbol{x}}^o$) and $\bar{\boldsymbol{s}}_i$ ($\dot{\bar{\boldsymbol{s}}}_i$), and $(\bar{\star})$ contains $\boldsymbol{x}$ ($\dot{\boldsymbol{x}}$) and $\bar{\boldsymbol{s}}_i$ ($\dot{\bar{\boldsymbol{s}}}_i$).

II. Measurement Models

Consider a scenario of N moving sensors and one moving source in a 3-D space. The position and velocity of the ith moving sensor are known and denoted by $\boldsymbol{s}_i^o$ $(i=1,\ldots,N)$ and $\dot{\boldsymbol{s}}_i^o$ $(i=1,\ldots,N)$, respectively, and the position and velocity of the source are unknown and denoted by $\boldsymbol{x}^o$ and $\dot{\boldsymbol{x}}^o$, respectively. The range difference measurements and their rates are, respectively, given by [16]

$$\begin{aligned} d_{i1} &= \|\boldsymbol{x}^o-\boldsymbol{s}_i^o\| - \|\boldsymbol{x}^o-\boldsymbol{s}_1^o\| + n_{i1}\\ \dot{d}_{i1} &= \frac{(\boldsymbol{x}^o-\boldsymbol{s}_i^o)^T(\dot{\boldsymbol{x}}^o-\dot{\boldsymbol{s}}_i^o)}{\|\boldsymbol{x}^o-\boldsymbol{s}_i^o\|} - \frac{(\boldsymbol{x}^o-\boldsymbol{s}_1^o)^T(\dot{\boldsymbol{x}}^o-\dot{\boldsymbol{s}}_1^o)}{\|\boldsymbol{x}^o-\boldsymbol{s}_1^o\|} + \dot{n}_{i1}\\ i &= 2,\ldots,N \end{aligned} \tag{1}$$

where n_{i1} and $\dot{n}_{i1}$ $(i=2,\ldots,N)$ are the measurement noise. The TDOA and FDOA measurements are then denoted by [16]

$$\tau_{i1} = d_{i1}/c \quad f_{i1} = f_0\dot{d}_{i1}/c, \quad i=2,\ldots,N \tag{2}$$

where c is the signal propagation speed, and f_0 is the carrier frequency. To simplify, we derive the proposed method by using the range difference measurements and their rates in (1).

Collect the measurement noise n_{i1} $(i=2,\ldots,N)$ and $\dot{n}_{i1}$ $(i=2,\ldots,N)$ into vectors $\boldsymbol{n}$ and $\dot{\boldsymbol{n}}$, and let $\Delta\boldsymbol{\alpha} = [\boldsymbol{n}^T\ \dot{\boldsymbol{n}}^T]^T$. Assume that $\Delta\boldsymbol{\alpha}$ follows a Gaussian distribution with zero mean and covariance $\boldsymbol{Q}_\alpha$, i.e., $\Delta\boldsymbol{\alpha}\sim\mathcal{N}(\mathbf{0},\boldsymbol{Q}_\alpha)$. Furthermore, we assume that $\boldsymbol{Q}_t = E[\boldsymbol{n}\boldsymbol{n}^T]$ and $\boldsymbol{Q}_f = E[\dot{\boldsymbol{n}}\dot{\boldsymbol{n}}^T]$. Note that $\boldsymbol{n}$ and $\dot{\boldsymbol{n}}$ are not necessarily uncorrelated.

By defining the following notations:

$$\begin{aligned} \boldsymbol{d} &= [d_{21}, d_{31}, \ldots, d_{N1}]^T, \quad \dot{\boldsymbol{d}} = [\dot{d}_{21}, \dot{d}_{31}, \ldots, \dot{d}_{N1}]^T\\ \boldsymbol{r}^o &= [r_1^o, r_2^o, \ldots, r_N^o]^T\\ &= [\|\boldsymbol{x}^o-\boldsymbol{s}_1^o\|, \|\boldsymbol{x}^o-\boldsymbol{s}_2^o\|, \ldots, \|\boldsymbol{x}^o-\boldsymbol{s}_N^o\|]^T\\ \dot{\boldsymbol{r}}^o &= [\dot{r}_1^o, \dot{r}_2^o, \ldots, \dot{r}_N^o]^T\\ &= \left[\frac{(\boldsymbol{x}^o-\boldsymbol{s}_1^o)^T(\dot{\boldsymbol{x}}^o-\dot{\boldsymbol{s}}_1^o)}{\|\boldsymbol{x}^o-\boldsymbol{s}_1^o\|}, \frac{(\boldsymbol{x}^o-\boldsymbol{s}_2^o)^T(\dot{\boldsymbol{x}}^o-\dot{\boldsymbol{s}}_2^o)}{\|\boldsymbol{x}^o-\boldsymbol{s}_2^o\|}\right.\\ &\qquad \left.\ldots, \frac{(\boldsymbol{x}^o-\boldsymbol{s}_N^o)^T(\dot{\boldsymbol{x}}^o-\dot{\boldsymbol{s}}_N^o)}{\|\boldsymbol{x}^o-\boldsymbol{s}_N^o\|}\right]^T \end{aligned} \tag{3}$$

we can rewrite (1) as

$$\begin{aligned} \boldsymbol{d} &= \boldsymbol{G}\boldsymbol{r}^o + \boldsymbol{n}\\ \dot{\boldsymbol{d}} &= \boldsymbol{G}\dot{\boldsymbol{r}}^o + \dot{\boldsymbol{n}} \end{aligned} \tag{4}$$

where $\boldsymbol{G} = [-\mathbf{1}_{N-1} \quad \boldsymbol{I}_{N-1}]$. Letting $\tilde{\boldsymbol{d}} = [\boldsymbol{d}^T \quad \dot{\boldsymbol{d}}^T]^T$, $\tilde{\boldsymbol{G}} = \mathrm{Diag}\{\boldsymbol{G}, \boldsymbol{G}\}$, and $\boldsymbol{\theta}^o = [\boldsymbol{r}^{oT} \quad \dot{\boldsymbol{r}}^{oT}]^T$, we can further rewrite (4) as

$$\tilde{\boldsymbol{d}} = \tilde{\boldsymbol{G}}\boldsymbol{\theta}^o + \boldsymbol{\Delta\alpha}. \tag{5}$$

In Sections III and IV, we develop the SDR methods to locate a moving source without and with errors in sensor positions and velocities, respectively. It is worth noting that the SDR methods can be easily tailored to locate a stationary source.

III. Semidefinite Relaxation Method for Source Localization Without Sensor Position and Velocity Errors

A. ML Estimation Without Sensor Position and Velocity Errors

To simplify the following derivations, we denote the true source position and velocity as $\boldsymbol{\phi}^o = [\boldsymbol{x}^{oT} \quad \dot{\boldsymbol{x}}^{oT}]^T$. According to (5), the ML estimation of $\boldsymbol{\phi}^o$ can be formulated as

$$\min_{\boldsymbol{\phi}} \quad (\tilde{\boldsymbol{d}} - \tilde{\boldsymbol{G}}\boldsymbol{\theta})^T \boldsymbol{Q}_{\boldsymbol{\alpha}}^{-1} (\tilde{\boldsymbol{d}} - \tilde{\boldsymbol{G}}\boldsymbol{\theta}). \tag{6}$$

Evidently, the ML problem (6) is nonconvex, implying that there exist multiple local minima and the global minimum can hardly be obtained. Indeed, using any local search algorithm runs the risk of being trapped in local minima, thus potentially resulting in quite inaccurate solutions.

B. Semidefinite Relaxation

Here, we present an SDR method to approximately solve the ML problem (6). To this end, we will first derive a WLS problem that is a close approximation to (6) and then apply SDR to this approximate WLS problem.

Proposition 1: The ML problem (6) can be approximated by the following WLS problem:

$$\min_{\boldsymbol{\phi}} \left[\boldsymbol{B}^{-1}(\boldsymbol{A}\boldsymbol{y} - \boldsymbol{b})\right]^T \boldsymbol{Q}_{\boldsymbol{\alpha}}^{-1} \left[\boldsymbol{B}^{-1}(\boldsymbol{A}\boldsymbol{y} - \boldsymbol{b})\right] = (\boldsymbol{A}\boldsymbol{y} - \boldsymbol{b})^T \boldsymbol{Q}^{-1} (\boldsymbol{A}\boldsymbol{y} - \boldsymbol{b}) \tag{7}$$

where

$$\boldsymbol{A} = 2\begin{bmatrix} (\boldsymbol{s}_2^o - \boldsymbol{s}_1^o)^T & \boldsymbol{O}_3^T & d_{21} & 0 \\ \vdots & \vdots & \vdots & \vdots \\ (\boldsymbol{s}_N^o - \boldsymbol{s}_1^o)^T & \boldsymbol{O}_3^T & d_{N1} & 0 \\ (\dot{\boldsymbol{s}}_2^o - \dot{\boldsymbol{s}}_1^o)^T & (\boldsymbol{s}_2^o - \boldsymbol{s}_1^o)^T & \dot{d}_{21} & d_{21} \\ \vdots & \vdots & \vdots & \vdots \\ (\dot{\boldsymbol{s}}_N^o - \dot{\boldsymbol{s}}_1^o)^T & (\boldsymbol{s}_N^o - \boldsymbol{s}_1^o)^T & \dot{d}_{N1} & d_{N1} \end{bmatrix}$$
$$\boldsymbol{b} = -\begin{bmatrix} d_{21}^2 - \|\boldsymbol{s}_2^o\|^2 + \|\boldsymbol{s}_1^o\|^2 \\ \vdots \\ d_{N1}^2 - \|\boldsymbol{s}_N^o\|^2 + \|\boldsymbol{s}_1^o\|^2 \\ 2d_{21}\dot{d}_{21} - 2\dot{\boldsymbol{s}}_2^{oT}\boldsymbol{s}_2^o + 2\dot{\boldsymbol{s}}_1^{oT}\boldsymbol{s}_1^o \\ \vdots \\ 2d_{N1}\dot{d}_{N1} - 2\dot{\boldsymbol{s}}_N^{oT}\boldsymbol{s}_N^o + 2\dot{\boldsymbol{s}}_1^{oT}\boldsymbol{s}_1^o \end{bmatrix} \tag{8}$$
$$\boldsymbol{Q} = \boldsymbol{B}\boldsymbol{Q}_{\boldsymbol{\alpha}}\boldsymbol{B}^T$$

$$\boldsymbol{y} = [\boldsymbol{x}^T \ \dot{\boldsymbol{x}}^T \ r_1 \ \dot{r}_1]^T$$
$$\boldsymbol{B} = \begin{bmatrix} \boldsymbol{B}_1 & \boldsymbol{O}_{(N-1)\times(N-1)} \\ \dot{\boldsymbol{B}}_1 & \boldsymbol{B}_1 \end{bmatrix} \tag{9}$$

with $\boldsymbol{B}_1 = 2\mathrm{diag}\{r_2, \ldots, r_N\}$ and $\dot{\boldsymbol{B}}_1 = 2\mathrm{diag}\{\dot{r}_2, \ldots, \dot{r}_N\}$.

Proof: The proof is motivated by the derivations in [16]. See Appendix A for details. ■

From the derivations in Appendix A, we see that (7) is obtained by neglecting the second-order noise terms in (36) and (38), shown below, which are far less than the first-order noise terms. For this reason, (7) is a very close approximation to (6). To our best knowledge, the nonlinear WLS problem is formulated for the first time that is a close approximation to the original ML problem.

Note that (7) is still a nonconvex problem. However, as compared with the original ML problem (6), it is much easier to apply the SDR technique to solve (7). In the following, we will apply this technique to relax the nonconvex problem into convex SDP, which can then be solved efficiently.

We solve (7) in two steps. Assume that we have an initial estimate of $\boldsymbol{\phi}^o$: $\hat{\boldsymbol{\phi}}_0 \triangleq [\hat{\boldsymbol{x}}_0^T \quad \hat{\dot{\boldsymbol{x}}}_0^T]^T$ (the way of obtaining $\hat{\boldsymbol{\phi}}_0$ is shown in Section III-C). In the first step, we substitute $\hat{\boldsymbol{x}}_0$ and $\hat{\dot{\boldsymbol{x}}}_0$ into $\boldsymbol{B}$ to obtain an estimate of $\boldsymbol{B}$: $\hat{\boldsymbol{B}}$. Replacing $\boldsymbol{B}$ with $\hat{\boldsymbol{B}}$ in $\boldsymbol{Q}$ to obtain an approximate weighting matrix $\hat{\boldsymbol{Q}} = \hat{\boldsymbol{B}}\boldsymbol{Q}_{\alpha}\hat{\boldsymbol{B}}^T$, we have the following approximate WLS problem:

$$\min_{\boldsymbol{\phi}} \quad (\boldsymbol{A}\boldsymbol{y} - \boldsymbol{b})^T \hat{\boldsymbol{Q}}^{-1} (\boldsymbol{A}\boldsymbol{y} - \boldsymbol{b}). \tag{10}$$

Problem (10) can be equivalently written as

$$\begin{aligned} \min_{\boldsymbol{y}=[\boldsymbol{x}^T \ \dot{\boldsymbol{x}}^T \ r_1 \ \dot{r}_1]^T} \quad & (\boldsymbol{A}\boldsymbol{y} - \boldsymbol{b})^T \hat{\boldsymbol{Q}}^{-1} (\boldsymbol{A}\boldsymbol{y} - \boldsymbol{b}) \\ \text{subject to} \quad & \boldsymbol{y}(7) = \|\boldsymbol{y}(1{:}3) - \boldsymbol{s}_1^o\| \\ & \boldsymbol{y}(8) = \frac{(\boldsymbol{y}(1{:}3) - \boldsymbol{s}_1^o)^T (\boldsymbol{y}(4{:}6) - \dot{\boldsymbol{s}}_1^o)}{\boldsymbol{y}(7)}. \end{aligned} \tag{11}$$

The objective function of (11) can be written as

$$(\boldsymbol{A}\boldsymbol{y} - \boldsymbol{b})^T \hat{\boldsymbol{Q}}^{-1} (\boldsymbol{A}\boldsymbol{y} - \boldsymbol{b}) = \mathrm{tr}\left\{\begin{bmatrix} \boldsymbol{Y} & \boldsymbol{y} \\ \boldsymbol{y}^T & 1 \end{bmatrix} \boldsymbol{F}\right\} \tag{12}$$

where

$$\boldsymbol{Y} = \boldsymbol{y}\boldsymbol{y}^T, \quad \boldsymbol{F} = \begin{bmatrix} \boldsymbol{A}^T\hat{\boldsymbol{Q}}^{-1}\boldsymbol{A} & -\boldsymbol{A}^T\hat{\boldsymbol{Q}}^{-1}\boldsymbol{b} \\ -\boldsymbol{b}^T\hat{\boldsymbol{Q}}^{-1}\boldsymbol{A} & \boldsymbol{b}^T\hat{\boldsymbol{Q}}^{-1}\boldsymbol{b} \end{bmatrix}. \tag{13}$$

The constraints in (11) can be rewritten as

$$\begin{aligned} \boldsymbol{Y}(7,7) &= \mathrm{tr}\{\boldsymbol{Y}(1:3, 1:3)\} - 2\boldsymbol{s}_1^{oT}\boldsymbol{y}(1:3) + \|\boldsymbol{s}_1^o\|^2 \\ \boldsymbol{Y}(7,8) &= \mathrm{tr}\{\boldsymbol{Y}(1:3, 4:6)\} - \dot{\boldsymbol{s}}_1^{oT}\boldsymbol{y}(1:3) \\ &\quad - \boldsymbol{s}_1^{oT}\boldsymbol{y}(4:6) + \dot{\boldsymbol{s}}_1^{oT}\boldsymbol{s}_1^o. \end{aligned} \tag{14}$$

Moreover, by the Cauchy–Schwartz inequality, we have

$$\begin{aligned} |\boldsymbol{y}(8)| &\le \frac{\|(\boldsymbol{y}(1:3) - \boldsymbol{s}_1^o)\| \, \|(\boldsymbol{y}(4:6) - \dot{\boldsymbol{s}}_1^o)\|}{\|\boldsymbol{y}(1:3) - \boldsymbol{s}_1^o\|} \\ &= \|(\boldsymbol{y}(4:6) - \dot{\boldsymbol{s}}_1^o)\| \end{aligned} \tag{15}$$

which is equivalent to

$$\boldsymbol{Y}(8,8) \leq \operatorname{tr}\{\boldsymbol{Y}(4:6,4:6)\} - 2\dot{\boldsymbol{s}}_1^{oT}\boldsymbol{y}(4:6) + \|\dot{\boldsymbol{s}}_1^o\|^2. \quad (16)$$

Now, the optimization problem (11) can be equivalently written as

$$\begin{aligned}
&\min_{\boldsymbol{Y},\boldsymbol{y}} \quad \operatorname{tr}\left\{\begin{bmatrix} \boldsymbol{Y} & \boldsymbol{y} \\ \boldsymbol{y}^T & 1 \end{bmatrix} \boldsymbol{F}\right\} \\
&\text{subject to} \\
&\boldsymbol{Y}(7,7) = \operatorname{tr}\{\boldsymbol{Y}(1:3,1:3)\} - 2\boldsymbol{s}_1^{oT}\boldsymbol{y}(1:3) + \|\boldsymbol{s}_1^o\|^2 \\
&\boldsymbol{Y}(7,8) = \operatorname{tr}\{\boldsymbol{Y}(1:3,4:6)\} - \dot{\boldsymbol{s}}_1^{oT}\boldsymbol{y}(1:3) \\
&\qquad\qquad - \boldsymbol{s}_1^{oT}\boldsymbol{y}(4:6) + \dot{\boldsymbol{s}}_1^{oT}\boldsymbol{s}_1^o \\
&\boldsymbol{Y}(8,8) \leq \operatorname{tr}\{\boldsymbol{Y}(4:6,4:6)\} - 2\dot{\boldsymbol{s}}_1^{oT}\boldsymbol{y}(4:6) + \|\dot{\boldsymbol{s}}_1^o\|^2 \\
&\boldsymbol{Y} = \boldsymbol{y}\boldsymbol{y}^T. \qquad (17)
\end{aligned}$$

In the second step, we perform SDR to problem (11). It can be easily verified that the last constraint $\boldsymbol{Y} = \boldsymbol{y}\boldsymbol{y}^T$ is equivalent to $\boldsymbol{Y} \succeq \boldsymbol{y}\boldsymbol{y}^T$ and $\operatorname{rank}(\boldsymbol{Y}) = 1$. By dropping the rank-1 constraint and establishing the equivalence between $\boldsymbol{Y} \succeq \boldsymbol{y}\boldsymbol{y}^T$ and $\begin{bmatrix} \boldsymbol{Y} & \boldsymbol{y} \\ \boldsymbol{y}^T & 1 \end{bmatrix} \succeq 0$ [29], we obtain the following SDP:

$$\begin{aligned}
&\min_{\boldsymbol{Y},\boldsymbol{y}} \quad \operatorname{tr}\left\{\begin{bmatrix} \boldsymbol{Y} & \boldsymbol{y} \\ \boldsymbol{y}^T & 1 \end{bmatrix} \boldsymbol{F}\right\} \\
&\text{subject to} \\
&\boldsymbol{Y}(7,7) = \operatorname{tr}\{\boldsymbol{Y}(1:3,1:3)\} - 2\boldsymbol{s}_1^{oT}\boldsymbol{y}(1:3) + \|\boldsymbol{s}_1^o\|^2 \\
&\boldsymbol{Y}(7,8) = \operatorname{tr}\{\boldsymbol{Y}(1:3,4:6)\} - \dot{\boldsymbol{s}}_1^{oT}\boldsymbol{y}(1:3) \\
&\qquad\qquad - \boldsymbol{s}_1^{oT}\boldsymbol{y}(4:6) + \dot{\boldsymbol{s}}_1^{oT}\boldsymbol{s}_1^o \\
&\boldsymbol{Y}(8,8) \leq \operatorname{tr}\{\boldsymbol{Y}(4:6,4:6)\} - 2\dot{\boldsymbol{s}}_1^{oT}\boldsymbol{y}(4:6) + \|\dot{\boldsymbol{s}}_1^o\|^2 \\
&[\boldsymbol{Y}\ \boldsymbol{y}; \boldsymbol{y}^T\ 1] \succeq 0 \qquad (18)
\end{aligned}$$

which can be solved very efficiently by using the interior-point methods.

Assume that the solution of (18) is denoted by $\{\hat{\boldsymbol{Y}}, \hat{\boldsymbol{y}}\}$, and the rank of $\hat{\boldsymbol{Y}}$ is $\operatorname{rank}(\hat{\boldsymbol{Y}}) = m$. According to [30], m satisfies the following relationship: $m(m+1) \leq 2u$, where u is the number of equality constraints. In (18), the number of equality constraints is $u = 4$. Hence, the rank of $\hat{\boldsymbol{Y}}$ is $m = 1$ or $m = 2$. Although $\hat{\boldsymbol{Y}}$ is not guaranteed to have rank 1, we find in our simulations that $\operatorname{rank}(\hat{\boldsymbol{Y}}) = 1$ can be frequently satisfied, indicating that SDP can frequently find the solution of the approximate WLS problem (10). Furthermore, similar to [16], the approximation of the weighting matrix $\hat{\boldsymbol{Q}}$ (and, hence, $\boldsymbol{F}$) has an insignificant effect on the final estimation accuracy. Our simulation results to be discussed in Section VI also verify this conclusion. Recall that (7) is a close approximation to the original ML problem (6), and from the above analysis, we conclude that the SDP (18) can provide good estimates.

C. Obtaining the Initial Estimate

In Section III-B, we have assumed that an initial position and velocity estimate $\hat{\phi}_0$ is obtained. Here, we propose a simple method to obtain this initial estimate: We will first estimate the source position using TDOA measurements and then use the position estimate to estimate the source velocity using FDOA measurements. Note that this has a lower computational cost than that of estimating the position and velocity simultaneously. In addition, we use a different initial estimate from that in [16]. Here, we use the second-order cone programming method presented in [14] to obtain the source position estimate, which is denoted by $\hat{\boldsymbol{x}}_0$. Substituting $\hat{\boldsymbol{x}}_0$ into the FDOA measurement model in (1), we have

$$\begin{aligned}
\dot{d}_{i1} &\approx \frac{(\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_i^o)^T(\dot{\boldsymbol{x}}^o - \dot{\boldsymbol{s}}_i^o)}{\|\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_i^o\|} - \frac{(\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_1^o)^T(\dot{\boldsymbol{x}}^o - \dot{\boldsymbol{s}}_1^o)}{\|\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_1^o\|} + \dot{n}_{i1} \\
&= \left(\frac{\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_i^o}{\|\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_i^o\|} - \frac{\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_1^o}{\|\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_1^o\|}\right)^T \dot{\boldsymbol{x}}^o \\
&\quad - \left(\frac{(\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_i^o)^T\boldsymbol{s}_i^o}{\|\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_i^o\|} - \frac{(\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_1^o)^T\boldsymbol{s}_1^o}{\|\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_1^o\|}\right) + \dot{n}_{i1} \\
i &= 2, \ldots, N \qquad (19)
\end{aligned}$$

which is linear with respect to $\dot{\boldsymbol{x}}^o$. Thus, we can obtain the linear WLS estimate of the source velocity as follows:

$$\hat{\dot{\boldsymbol{x}}}_0 = \left(\boldsymbol{D}^T\boldsymbol{Q}_f^{-1}\boldsymbol{D}\right)^{-1}\boldsymbol{D}^T\boldsymbol{Q}_f^{-1}\boldsymbol{f} \qquad (20)$$

where

$$\boldsymbol{D} = \begin{bmatrix} \frac{(\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_2^o)^T}{\|\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_2^o\|} - \frac{(\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_1^o)^T}{\|\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_1^o\|} \\ \vdots \\ \frac{(\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_N^o)^T}{\|\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_N^o\|} - \frac{(\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_1^o)^T}{\|\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_1^o\|} \end{bmatrix}$$

$$\boldsymbol{f} = \begin{bmatrix} \frac{(\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_2^o)^T\boldsymbol{s}_2^o}{\|\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_2^o\|} - \frac{(\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_1^o)^T\boldsymbol{s}_1^o}{\|\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_1^o\|} \\ \vdots \\ \frac{(\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_N^o)^T\boldsymbol{s}_N^o}{\|\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_N^o\|} - \frac{(\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_1^o)^T\boldsymbol{s}_1^o}{\|\hat{\boldsymbol{x}}_0 - \boldsymbol{s}_1^o\|} \end{bmatrix}. \qquad (21)$$

Substituting $\hat{\boldsymbol{x}}_0$ and $\hat{\dot{\boldsymbol{x}}}_0$ into $\boldsymbol{F}$ and then solving the SDP (18) give the final estimate of $\boldsymbol{x}^o$ and $\dot{\boldsymbol{x}}^o$.

D. Comparison With Other Methods

Here, we compare the proposed SDR method with the existing methods presented in [20] and [21].

The QCS method in [20] finds two Lagrange multipliers through Newton's method. Indeed, the QCS method solves problem (11). As aforementioned, problem (11) is a nonconvex problem, which implies that the QCS method cannot guarantee that it will find its global solution. This is particularly true at high noise levels that result in performance degradation. In comparison, the SDP can always find a global solution. Although the SDP (18) is a relaxation and approximation to the WLS problem (11), we have pointed out at the end of Section III-B that SDP can find good estimates, as verified by simulations.

The SDR method presented in [21] is a relaxation of the original ML problem (6). In this SDR method, the condition that the solution has a maximum rank of 2 is not satisfied, implying that it may be a loose relaxation to the ML problem. As a result, the SDP solution is not good enough, and local search is needed to find a solution of the ML problem. This, in turn, may bring local convergence problem. In comparison, the proposed SDR method is a relaxation to the approximate ML problem (which is shown in Proposition 1), the SDP solution is good enough, and local search is not needed.

IV. Semidefinite Relaxation Method for Source Localization With Sensor Position and Velocity Errors

In practice, the sensor positions and velocities are typically not exactly known, and the localization performance can be significantly improved by taking sensor position and velocity errors into account [17]. Here, we extend the SDR method to address localization in the presence of errors in sensor positions and velocities. We first introduce some new notations. We assume that $\bar{s}_i$ $(i = 1 \ldots, N)$ and $\dot{\bar{s}}_i$ $(i = 1 \ldots, N)$ are the estimated position and velocity of the ith sensor, respectively. Moreover, we assume that $\bar{\boldsymbol{s}}_i = \boldsymbol{s}_i^o + \Delta \boldsymbol{s}_i$ and $\dot{\bar{\boldsymbol{s}}}_i = \dot{\boldsymbol{s}}_i^o + \Delta \dot{\boldsymbol{s}}_i$, where $\Delta \boldsymbol{s}_i$ and $\Delta \dot{\boldsymbol{s}}_i$ are the position and velocity errors, respectively. For notational simplicity, we stack the true sensor positions and velocities into vector $\boldsymbol{\beta}^o$, i.e., $\boldsymbol{\beta}^o \triangleq [\boldsymbol{s}^{oT} \quad \dot{\boldsymbol{s}}^{oT}]^T$, and similarly, $\bar{\boldsymbol{\beta}} \triangleq [\bar{\boldsymbol{s}}^T \quad \dot{\bar{\boldsymbol{s}}}^T]^T$, and $\Delta\boldsymbol{\beta} \triangleq [\Delta \boldsymbol{s}^T \quad \Delta \dot{\boldsymbol{s}}^T]^T$. Obviously, $\bar{\boldsymbol{\beta}} = \boldsymbol{\beta}^o + \Delta\boldsymbol{\beta}$ holds. We further assume that $\Delta\boldsymbol{\beta}$ follows a zero-mean Gaussian distribution with covariance matrix $\boldsymbol{Q}_\beta = E[\Delta\boldsymbol{\beta}\Delta\boldsymbol{\beta}^T]$ and is mutually independent with the TDOA/FDOA measurement noise vector $\Delta\boldsymbol{\alpha}$. With the use of these notations, we first derive an ML formulation that can achieve the CRLB accuracy in the presence of sensor position and velocity errors and then derive an approximate WLS formulation and apply SDR to obtain an SDP in a similar manner to that in Section III-B.

A. ML Estimation With Sensor Position and Velocity Errors

Substituting $\bar{\boldsymbol{s}}_i = \boldsymbol{s}_i^o + \Delta \boldsymbol{s}_i$ and $\dot{\bar{\boldsymbol{s}}}_i = \dot{\boldsymbol{s}}_i^o + \Delta \dot{\boldsymbol{s}}_i$ into (1) and applying the first-order Taylor series expansion, we have [17]

$$\begin{aligned} &d_{i1} - (\|\boldsymbol{x}^o - \bar{\boldsymbol{s}}_i\| - \|\boldsymbol{x}^o - \bar{\boldsymbol{s}}_1\|) \\ &\quad \approx n_{i1} + \boldsymbol{u}_i^{oT}\Delta\boldsymbol{s}_i - \boldsymbol{u}_1^{oT}\Delta\boldsymbol{s}_1 \\ &\dot{d}_{i1} - \left[\frac{(\boldsymbol{x}^o - \bar{\boldsymbol{s}}_i)^T(\dot{\boldsymbol{x}}^o - \dot{\bar{\boldsymbol{s}}}_i)}{\|\boldsymbol{x}^o - \bar{\boldsymbol{s}}_i\|} - \frac{(\boldsymbol{x}^o - \bar{\boldsymbol{s}}_1)^T(\dot{\boldsymbol{x}}^o - \dot{\bar{\boldsymbol{s}}}_1)}{\|\boldsymbol{x}^o - \bar{\boldsymbol{s}}_1\|}\right] \\ &\quad \approx \dot{n}_{i1} + \boldsymbol{u}_i^{oT}\Delta\dot{\boldsymbol{s}}_i - \boldsymbol{u}_1^{oT}\Delta\dot{\boldsymbol{s}}_1 + \dot{\boldsymbol{u}}_i^{oT}\Delta\boldsymbol{s}_i - \dot{\boldsymbol{u}}_1^{oT}\Delta\boldsymbol{s}_1 \end{aligned} \quad (22)$$

where $\boldsymbol{u}_i^o = (\boldsymbol{x}^o - \boldsymbol{s}_i^o)/\|\boldsymbol{x}^o - \boldsymbol{s}_i^o\|$, and $\dot{\boldsymbol{u}}_i^o = (\dot{\boldsymbol{x}}^o - \dot{\boldsymbol{s}}_i^o)/\|\boldsymbol{x}^o - \boldsymbol{s}_i^o\| - [(\boldsymbol{x}^o - \boldsymbol{s}_i^o)(\boldsymbol{x}^o - \boldsymbol{s}_i^o)^T(\dot{\boldsymbol{x}}^o - \dot{\boldsymbol{s}}_i^o)]/\|\boldsymbol{x}^o - \boldsymbol{s}_i^o\|^3$.

Define $\boldsymbol{U}^o$ as

$$\boldsymbol{U}^o = \begin{bmatrix} \boldsymbol{U}_1^o & \boldsymbol{O}_{(N-1)\times 3N} \\ \dot{\boldsymbol{U}}_1^o & \boldsymbol{U}_1^o \end{bmatrix} \quad (23)$$

where the ith $(i = 1, \ldots, N-1)$ row of $\boldsymbol{U}_1^o$ and $\dot{\boldsymbol{U}}_1^o$ are, respectively, given by

$$\begin{aligned} \boldsymbol{U}_1^o(i,:) &= [\,-\boldsymbol{u}_1^{oT} \quad \boldsymbol{O}_{3(i-1)}^T \quad \boldsymbol{u}_i^{oT} \quad \boldsymbol{O}_{3(M-i-1)}^T\,] \\ \dot{\boldsymbol{U}}_1^o(i,:) &= [\,-\dot{\boldsymbol{u}}_1^{oT} \quad \boldsymbol{O}_{3(i-1)}^T \quad \dot{\boldsymbol{u}}_i^{oT} \quad \boldsymbol{O}_{3(M-i-1)}^T\,]. \end{aligned} \quad (24)$$

Writing all the equations in (22) in matrix form gives

$$\tilde{\boldsymbol{d}} - \tilde{\boldsymbol{G}}\bar{\boldsymbol{\theta}}^o = \Delta\boldsymbol{\alpha} + \boldsymbol{U}^o\Delta\boldsymbol{\beta} \quad (25)$$

where $\bar{\boldsymbol{\theta}}^o = [\bar{\boldsymbol{r}}^{oT} \quad \dot{\bar{\boldsymbol{r}}}^{oT}]^T$ with

$$\begin{aligned} \bar{\boldsymbol{r}}^o &= [\bar{r}_1^o, \bar{r}_2^o, \ldots, \bar{r}_N^o]^T \\ &= [\|\boldsymbol{x}^o - \bar{\boldsymbol{s}}_1\|, \|\boldsymbol{x}^o - \bar{\boldsymbol{s}}_2\|, \ldots, \|\boldsymbol{x}^o - \bar{\boldsymbol{s}}_N\|]^T \\ \dot{\bar{\boldsymbol{r}}}^o &= [\dot{\bar{r}}_1^o, \dot{\bar{r}}_2^o, \ldots, \dot{\bar{r}}_N^o]^T \\ &= \left[\frac{(\boldsymbol{x}^o - \bar{\boldsymbol{s}}_1)^T(\dot{\boldsymbol{x}}^o - \dot{\bar{\boldsymbol{s}}}_1)}{\|\boldsymbol{x}^o - \bar{\boldsymbol{s}}_1\|}, \frac{(\boldsymbol{x}^o - \bar{\boldsymbol{s}}_2)^T(\dot{\boldsymbol{x}}^o - \dot{\bar{\boldsymbol{s}}}_2)}{\|\boldsymbol{x}^o - \bar{\boldsymbol{s}}_2\|}\right. \\ &\qquad \left. \ldots, \frac{(\boldsymbol{x}^o - \bar{\boldsymbol{s}}_N)^T(\dot{\boldsymbol{x}}^o - \dot{\bar{\boldsymbol{s}}}_N)}{\|\boldsymbol{x}^o - \bar{\boldsymbol{s}}_N\|}\right]^T. \end{aligned} \quad (26)$$

According to (25), the ML estimation in the presence of sensor position and velocity errors can be formulated as

$$\min_{\boldsymbol{\phi}} \quad (\tilde{\boldsymbol{d}} - \tilde{\boldsymbol{G}}\boldsymbol{\theta})^T(\boldsymbol{Q}_{\boldsymbol{\alpha}} + \boldsymbol{U}^o\boldsymbol{Q}_{\boldsymbol{\beta}}\boldsymbol{U}^{oT})^{-1}(\tilde{\boldsymbol{d}} - \tilde{\boldsymbol{G}}\boldsymbol{\theta}). \quad (27)$$

It is worth noting that, in the ML formulation (27), we assume that $\boldsymbol{U}^o$ is exactly known, which is clearly not true in practice. Hence, this ML problem is unsolvable in practice. In [17], a CRLB was derived for the case when the errors in sensor positions and velocities are present. Similar to the derivations in [17], we can show that the mean square error (MSE) of the ML estimation problem (27) can attain this CRLB under mild conditions. This means that this CRLB can be achieved only when the positions and velocities of the source and sensors in $\boldsymbol{U}^o$ are exactly known. Indeed, this CRLB is the performance lower bound for the joint ML estimation of the positions and velocities of both the source and the sensors, and the joint ML estimation has a potential ability to reduce sensor position and velocity errors [31]. Hence, for any practical estimator that estimates the position and velocity of the source only, this CRLB is a loose bound. However, extensive simulations (e.g., in [17] and Section VI of this paper) show that the CRLB can be achieved at low error levels, indicating that the ability of reducing sensor position and velocity errors is quite limited. Hence, this CRLB is still very useful in practice.

B. Semidefinite Relaxation

The ML problem (27) is unsolvable; hence, it is not useful in practice. Here, we first give an approximate WLS formulation and then apply SDR to the approximate WLS formulation to obtain the estimate of the source position and velocity.

We first present the following proposition.

Proposition 2: The ML problem (27) can be approximated by the following WLS problem:

$$\begin{aligned} \min_{\boldsymbol{\phi}} \left[\bar{\boldsymbol{B}}^{-1}(\bar{\boldsymbol{A}}\bar{\boldsymbol{y}} - \bar{\boldsymbol{b}})\right]^T &(\boldsymbol{Q}_{\boldsymbol{\alpha}} + \bar{\boldsymbol{B}}^{o-1}\bar{\boldsymbol{C}}^o\boldsymbol{Q}_{\boldsymbol{\beta}}\bar{\boldsymbol{C}}^{oT}\bar{\boldsymbol{B}}^{o-T})^{-1} \\ &\times \left[\bar{\boldsymbol{B}}^{-1}(\bar{\boldsymbol{A}}\bar{\boldsymbol{y}} - \bar{\boldsymbol{b}})\right] \end{aligned} \quad (28)$$

where $\bar{\boldsymbol{B}}^o$, $\bar{\boldsymbol{A}}$, $\bar{\boldsymbol{b}}$, and $\bar{\boldsymbol{y}}^o$ are, respectively, obtained by replacing $\boldsymbol{s}_i^o$ and $\dot{\boldsymbol{s}}_i^o$ with $\bar{\boldsymbol{s}}_i$ and $\bar{\dot{\boldsymbol{s}}}_i$ in $\boldsymbol{B}^o$, $\boldsymbol{A}$, $\boldsymbol{b}$, and $\boldsymbol{y}^o$, and $\bar{\boldsymbol{C}}^o$ is defined as

$$\bar{\boldsymbol{C}}^o \triangleq \begin{bmatrix} \bar{\boldsymbol{C}}_1^o & \boldsymbol{O}_{(N-1)\times 3N} \\ \bar{\dot{\boldsymbol{C}}}_1^o & \bar{\boldsymbol{C}}_1^o \end{bmatrix}. \tag{29}$$

The ith $(i = 1, \ldots, N-1)$ row of $\bar{\boldsymbol{C}}_1^o$ and $\bar{\dot{\boldsymbol{C}}}_1^o$ are, respectively, given in (30), shown at the bottom of the page.

Proof: The proof is partially based on the derivations in [17]. See Appendix B for details. ∎

In (28), we have assumed that $\bar{\boldsymbol{B}}^o$ and $\bar{\boldsymbol{C}}^o$ in the weighting matrix are known, which is clearly not true in practice. Hence, (28) is still unsolvable, indicating that further approximation must be needed. We use the initial estimate $\hat{\boldsymbol{\phi}}_0$ to obtain an estimate of $\bar{\boldsymbol{B}}^o$ and $\bar{\boldsymbol{C}}^o$, i.e., $\hat{\bar{\boldsymbol{B}}}$ and $\hat{\bar{\boldsymbol{C}}}$ ($\hat{\bar{\boldsymbol{C}}}$ is obtained in the same way as $\hat{\bar{\boldsymbol{B}}}$), respectively, and then, (28) can be further approximated by

$$\min_{\boldsymbol{\phi}} \left[\bar{\boldsymbol{B}}^{-1}(\bar{\boldsymbol{A}}\bar{\boldsymbol{y}} - \bar{\boldsymbol{b}})\right]^T (\boldsymbol{Q}_{\boldsymbol{\alpha}} + \hat{\bar{\boldsymbol{B}}}^{-1}\hat{\bar{\boldsymbol{C}}}\boldsymbol{Q}_{\boldsymbol{\beta}}\hat{\bar{\boldsymbol{C}}}^T\hat{\bar{\boldsymbol{B}}}^{-T})^{-1} \times \left[\bar{\boldsymbol{B}}^{-1}(\bar{\boldsymbol{A}}\bar{\boldsymbol{y}} - \bar{\boldsymbol{b}})\right]. \tag{31}$$

Combining (28) and (49) shown below, and comparing (31) with (27), we see that (31) is actually an approximation of (27) by approximating the true source and sensor position and velocity values $\boldsymbol{\phi}^o$ and $\boldsymbol{\beta}^o$ in $\boldsymbol{U}^o$ in (27) using their estimates $\hat{\boldsymbol{\phi}}_0$ and $\bar{\boldsymbol{\beta}}$, respectively.

Note that, now, (31) has the same form as (7), and we can use the same procedure (two steps) in Section III-B to solve (31). In the first step, we obtain $\hat{\boldsymbol{Q}}$ as $\hat{\boldsymbol{Q}} = \hat{\bar{\boldsymbol{B}}}\boldsymbol{Q}_\alpha\hat{\bar{\boldsymbol{B}}}^T + \hat{\bar{\boldsymbol{C}}}\boldsymbol{Q}_\beta\hat{\bar{\boldsymbol{C}}}^T$. In the second step, we obtain the SDP that has exactly the same form as that in (18), except with the replacement of $\boldsymbol{s}_i^o$ and $\dot{\boldsymbol{s}}_i^o$ by $\bar{\boldsymbol{s}}_i$ and $\bar{\dot{\boldsymbol{s}}}_i$, respectively, as follows:

$$\begin{aligned}
&\min_{\bar{\boldsymbol{Y}},\bar{\boldsymbol{y}}} \quad \mathrm{tr}\left\{\begin{bmatrix} \bar{\boldsymbol{Y}} & \bar{\boldsymbol{y}} \\ \bar{\boldsymbol{y}}^T & 1 \end{bmatrix} \bar{\boldsymbol{F}}\right\} \\
&\text{subject to} \\
&\bar{\boldsymbol{Y}}(7,7) = \mathrm{tr}\left\{\bar{\boldsymbol{Y}}(1:3,1:3)\right\} - 2\bar{\boldsymbol{s}}_1^T\bar{\boldsymbol{y}}(1:3) + \|\bar{\boldsymbol{s}}_1\|^2 \\
&\bar{\boldsymbol{Y}}(7,8) = \mathrm{tr}\left\{\bar{\boldsymbol{Y}}(1:3,4:6)\right\} - \bar{\dot{\boldsymbol{s}}}_1^T\bar{\boldsymbol{y}}(1:3) \\
&\qquad\qquad - \bar{\boldsymbol{s}}_1^T\bar{\boldsymbol{y}}(4:6) + \bar{\dot{\boldsymbol{s}}}_1^T\bar{\boldsymbol{s}}_1 \\
&\bar{\boldsymbol{Y}}(8,8) \le \mathrm{tr}\left\{\bar{\boldsymbol{Y}}(4:6,4:6)\right\} - 2\bar{\dot{\boldsymbol{s}}}_1^T\bar{\boldsymbol{y}}(4:6) + \|\bar{\dot{\boldsymbol{s}}}_1\|^2 \\
&[\bar{\boldsymbol{Y}}\ \bar{\boldsymbol{y}}; \bar{\boldsymbol{y}}^T\ 1] \succeq 0
\end{aligned} \tag{32}$$

where

$$\bar{\boldsymbol{F}} = \begin{bmatrix} \bar{\boldsymbol{A}}^T\hat{\boldsymbol{Q}}^{-1}\bar{\boldsymbol{A}} & -\bar{\boldsymbol{A}}^T\hat{\boldsymbol{Q}}^{-1}\bar{\boldsymbol{b}} \\ -\bar{\boldsymbol{b}}^T\hat{\boldsymbol{Q}}^{-1}\bar{\boldsymbol{A}} & \bar{\boldsymbol{b}}^T\hat{\boldsymbol{Q}}^{-1}\bar{\boldsymbol{b}} \end{bmatrix}$$

with $\hat{\boldsymbol{Q}} = \hat{\bar{\boldsymbol{B}}}\boldsymbol{Q}_\alpha\hat{\bar{\boldsymbol{B}}}^T + \hat{\bar{\boldsymbol{C}}}\boldsymbol{Q}_\beta\hat{\bar{\boldsymbol{C}}}^T$.

TABLE I
POSITIONS AND VELOCITIES OF THE SENSORS

Sensor no.	x	y	z	$\dot{x}$	$\dot{y}$	$\dot{z}$
1	300	100	150	30	-20	20
2	400	150	100	-30	10	20
3	300	500	200	10	-20	10
4	350	200	100	10	20	30
5	-100	-100	-100	-20	10	10

It is worth mentioning that we still use the same scheme described in Section III-C to obtain the initial estimate $\hat{\boldsymbol{\phi}}_0$.

V. COMPLEXITY ANALYSIS

Here, we analyze the computational complexity of the proposed method. The complexity of the first step is *at most* $\mathcal{O}(N(n+1)^2 + L(n+1)^3)$, where L is the number of iterations,[1] and n is the dimension of the source location (here, $n = 3$) [14]. In the second step, the complexity of computing $\boldsymbol{F}$ (or $\bar{\boldsymbol{F}}$) is roughly $\mathcal{O}(16(N^2n + Nn^2))$. The *worst case* complexity of solving SDP is $\mathcal{O}((u^3 + u^2v^2 + uv^3)v^{0.5})$ [32], where u is the number of equality constraints (SDP in the standard primal form), and v is the problem size. In the SDP (18) and (32), $u = 4$, and $v = 2n + 3$. Since $v \gg u$, the complexity is roughly $\mathcal{O}(4(2n+3)^{3.5})$. Hence, the total complexity is roughly $\mathcal{O}(16(N^2n + Nn^2) + L(n+1)^3 + 4(2n+3)^{3.5})$. In the SDR method in [21], $u = \mathcal{O}(N^2)$, and $v = \mathcal{O}(N)$, which results in the complexity of roughly $\mathcal{O}(N^{6.5})$.

VI. SIMULATIONS

Here, simulations are conducted to verify the performance of the proposed method, which is compared with that of several existing methods [16], [17], [19].

A. Localization Performance Without Sensor Position and Velocity Errors

Consider a scenario in which five moving sensors are used to locate one moving source. The positions and velocities of the sensors are listed in Table I; they are the same as those in [16]. The performance is evaluated in terms of root MSEs (RMSEs), which are defined by $\sqrt{E[(\hat{\boldsymbol{x}} - \boldsymbol{x}^o)^T(\hat{\boldsymbol{x}} - \boldsymbol{x}^o)]}$ and $\sqrt{E[(\hat{\dot{\boldsymbol{x}}} - \dot{\boldsymbol{x}}^o)^T(\hat{\dot{\boldsymbol{x}}} - \dot{\boldsymbol{x}}^o)]}$ for position and velocity estimations, respectively. In the simulation, RMSEs are obtained using 3000 Monte Carlo runs. The TDOA measurement noise and FDOA measurement noise are assumed to be independent, and the covariance matrices are $\boldsymbol{Q}_t = \sigma_d^2\boldsymbol{V}_d$ and $\boldsymbol{Q}_f = 0.1\sigma_d^2\boldsymbol{V}_d$, respectively, where σ_d^2 represents the measurement noise level,

[1] L is related to the search interval $[p, q]$, and the solution precision ϵ: L is the smallest integer that satisfies $L > \log_2[(q-p)/\epsilon]$.

$$\begin{aligned}
\bar{\boldsymbol{C}}_1^o(i,:) &= \left[-(d_{i1}\bar{\boldsymbol{u}}_1^o + (\boldsymbol{x}^o - \bar{\boldsymbol{s}}_1))^T \quad \mathbf{0}_{3(i-1)}^T \ (\boldsymbol{x}^o - \bar{\boldsymbol{s}}_i)^T \quad \mathbf{0}_{3(M-i-1)}^T\right] \\
\bar{\dot{\boldsymbol{C}}}_1^o(i,:) &= \left[-\left(\dot{d}_{i1}\bar{\boldsymbol{u}}_1^o + d_{i1}\bar{\dot{\boldsymbol{u}}}_1^o + (\dot{\boldsymbol{x}}^o - \bar{\dot{\boldsymbol{s}}}_1)\right)^T \quad \mathbf{0}_{3(i-1)}^T \ (\dot{\boldsymbol{x}}^o - \bar{\dot{\boldsymbol{s}}}_i)^T \quad \mathbf{0}_{3(M-i-1)}^T\right]
\end{aligned} \tag{30}$$

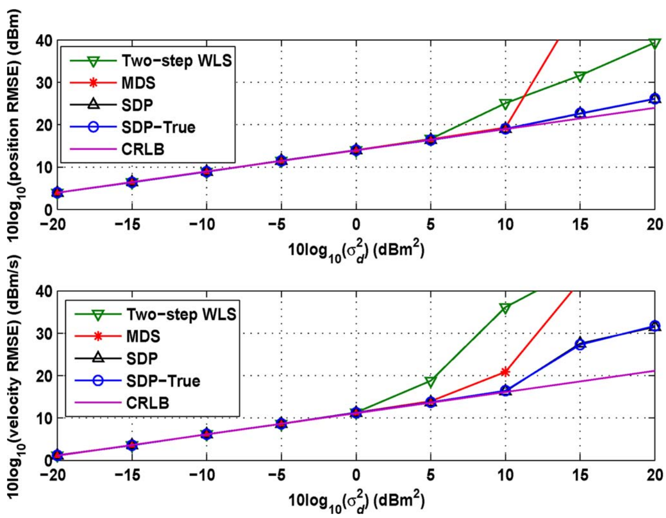

Fig. 1. Comparison of RMSEs using different methods in the absence of sensor position errors: the near-field scenario.

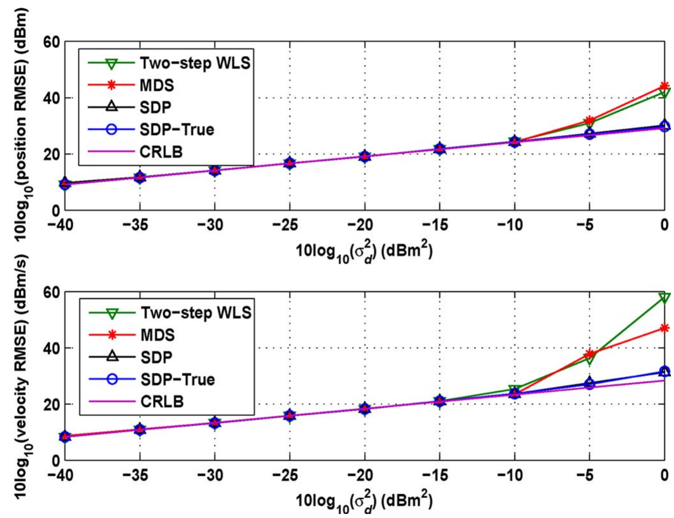

Fig. 2. Comparison of RMSEs using different methods in the absence of sensor position errors: the far-field scenario.

and $\boldsymbol{V}_d$ is set to 1 in the diagonal elements and 0.5 elsewhere. The SDPs in this paper are solved using the MATLAB toolbox CVX[2] [33], where the solver is SeDuMi [34].

We first consider the near-field localization case. The source is assumed to be located at $\boldsymbol{x}^o = (600, 650, 550)$ with velocity $\dot{\boldsymbol{x}}^o = (-20, 15, 40)$. The RMSEs and the CRLB [16] are shown in Fig. 1, from which we see that the proposed method performs the best. Both the two-stage WLS and MDS methods have the "threshold effect" when the noise level is higher than 10 dB. In comparison, the RMSE of the position estimates using the SDP method can attain the CRLB, even at high noise levels. Although the RMSE of the velocity estimates cannot attain the CRLB, it is still much smaller than those using the other methods.

We next consider the far-field case. The source is assumed to be located at $\boldsymbol{x}^o = (2000, 2500, 3000)$ with velocity $\dot{\boldsymbol{x}}^o = (-20, 15, 40)$. The simulation results are shown in Fig. 2, from which we see that the SDP method still performs much better than the other methods. Even at high noise levels, the RMSE of velocity estimates can attain the CRLB. It is worth noting that in both cases, we do not use any postprocessing procedures (e.g., local search), indicating that the SDP solution is very accurate although it is generally not the solution of the original ML problem.

B. Localization Performance With Sensor Position and Velocity Errors

Here, we consider localization in the presence of sensor position and velocity errors. The RMSE is also used to evaluate the performance of the proposed method. As earlier, we first consider the near-field case. The source is still located at $\boldsymbol{x}^o = (600, 650, 550)$ with velocity $\dot{\boldsymbol{x}}^o = (-20, 15, 40)$, and the positions and velocities of the sensors are the same as those in Table I. We assume that the covariance of the sensor position and velocity errors is $\boldsymbol{Q}_\beta = \mathrm{Diag}\{\boldsymbol{V}_s, \dot{\boldsymbol{V}}_s\}$,

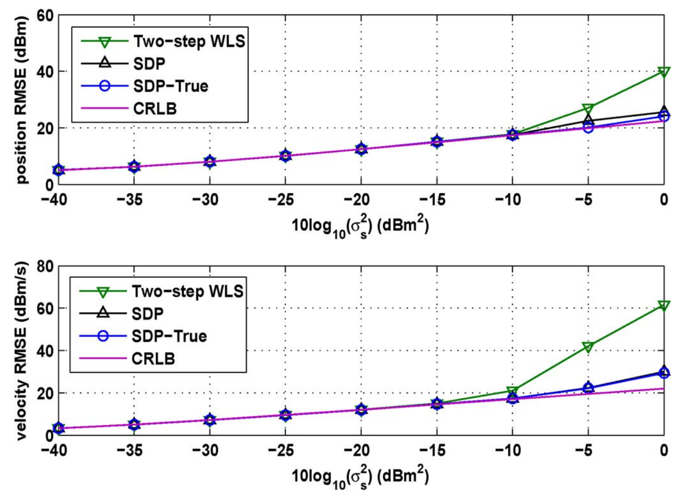

Fig. 3. Comparison of RMSEs using different methods in the presence of sensor position errors: the near-field scenario.

where $\boldsymbol{V}_s \triangleq E[\Delta \boldsymbol{s} \Delta \boldsymbol{s}^T] = \sigma_s^2 \mathrm{diag}\{1, 1, 1, 2, 2, 2, 10, 10, 10, 40, 40, 40, 20, 20, 20\}$, and $\dot{\boldsymbol{V}}_s \triangleq E[\Delta \dot{\boldsymbol{s}} \Delta \dot{\boldsymbol{s}}^T] = 0.5 \boldsymbol{V}_s$. We fix the TDOA measurement noise level σ_d^2 as $\sigma_d^2 = -20$ dB and examine the variation of the RMSE with the sensor position error level σ_s^2. The RMSEs and the CRLB [17] are shown in Fig. 3, from which we see that the SDP method performs much better than the two-stage WLS method, and it can always achieve the CRLB accuracy. Fig. 4 shows the simulation results for locating a far-field source located at $\boldsymbol{x}^o = (2000, 2500, 3000)$ with velocity $\dot{\boldsymbol{x}}^o = (-20, 15, 40)$. In this case, we add a sensor located at $(200, -300, -200)$ with velocity $(20, -10, 10)$ to the above sensor network to obtain better localization performance, i.e., we use six sensors to locate this far-field source. In this case, we set $\sigma_d^2 = -40$ dB, $\boldsymbol{V}_s = \sigma_s^2 \mathrm{diag}\{1, 1, 1, 2, 2, 2, 10, 10, 10, 40, 40, 40, 20, 20, 20, 3, 3, 3\}$, and $\dot{\boldsymbol{V}}_s = 0.5\, \boldsymbol{V}_s$. Again, we see in Fig. 4 that the SDP method performs much better than the two-stage WLS method. In comparing Fig. 3 with Fig. 4, we see that the localization performance in the far-field case is far more sensitive to the sensor position and velocity errors than that in

[2]The solution precision in CVX is set as "best" [33].

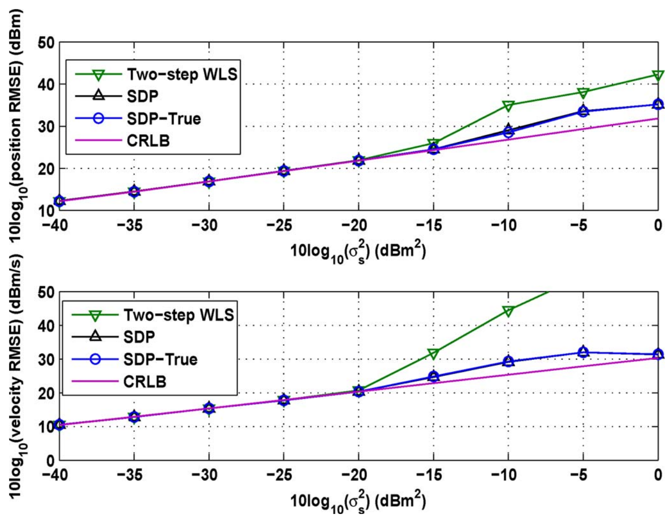

Fig. 4. Comparison of RMSEs using different methods in the presence of sensor position errors: the far-field scenario.

the near-field case. An important observation revealed in Figs. 3 and 4 is that the SDP solution is quite reliable, even at high position error levels, although approximation and relaxation are applied to the original ML problem.

VII. Conclusion

In this paper, we have proposed an SDR method to solve the TDOA/FDOA-based localization problem. This method applies the SDR technique to the reformulated WLS problem to obtain an SDP, whose solution is very accurate; thus, local search is not needed. Moreover, the proposed SDR method has low complexity, making it applicable in real-time applications. Simulation results show that this method significantly outperforms several existing methods at high noise levels.

Appendix A
Proof of Proposition 1

Consider the noise-free range difference measurement, i.e.,

$$d_{i1}^o = \|\boldsymbol{x}^o - \boldsymbol{s}_i^o\| - \|\boldsymbol{x}^o - \boldsymbol{s}_1^o\| \tag{33}$$

which is equivalent to

$$d_{i1}^o + \|\boldsymbol{x}^o - \boldsymbol{s}_1^o\| = \|\boldsymbol{x}^o - \boldsymbol{s}_i^o\|. \tag{34}$$

Squaring both sides of (34) yields

$$d_{i1}^{o2} + 2d_{i1}^o r_1^o = \|\boldsymbol{s}_i^o\|^2 - \|\boldsymbol{s}_1^o\|^2 - 2\left(\boldsymbol{s}_i^o - \boldsymbol{s}_1^o\right)^T \boldsymbol{x}^o \quad i = 2, \ldots, N. \tag{35}$$

Substituting $d_{i1}^o = d_{i1} - n_{i1}$ into (35) gives

$$d_{i1}^2 + 2d_{i1}r_1^o - \|\boldsymbol{s}_i^o\|^2 + \|\boldsymbol{s}_1^o\|^2 + 2\left(\boldsymbol{s}_i^o - \boldsymbol{s}_1^o\right)^T \boldsymbol{x}^o \approx 2r_i^o n_{i1} \tag{36}$$

where the second-order noise term n_{i1}^2 is neglected.

Taking the time derivative of (35), we have the following equations related to the FDOAs:

$$\begin{aligned} d_{i1}^o \dot{d}_{i1}^o + d_{i1}^o \dot{r}_1^o + \dot{d}_{i1}^o r_1^o = \dot{\boldsymbol{s}}_i^{oT} \boldsymbol{s}_i^o - \dot{\boldsymbol{s}}_1^{oT} \boldsymbol{s}_1^o - \left(\dot{\boldsymbol{s}}_i^o - \dot{\boldsymbol{s}}_1^o\right)^T \boldsymbol{x}^o \\ - \left(\boldsymbol{s}_i^o - \boldsymbol{s}_1^o\right)^T \dot{\boldsymbol{x}}^o, \quad i = 2, \ldots, N \end{aligned} \tag{37}$$

where $\dot{d}_{i1}^o = ((\boldsymbol{x}^o - \boldsymbol{s}_i^o)^T(\dot{\boldsymbol{x}}^o - \dot{\boldsymbol{s}}_i^o))/\|\boldsymbol{x}^o - \boldsymbol{s}_i^o\| - ((\boldsymbol{x}^o - \boldsymbol{s}_1^o)^T(\dot{\boldsymbol{x}}^o - \dot{\boldsymbol{s}}_1^o))/\|\boldsymbol{x}^o - \boldsymbol{s}_1^o\|$ $(i = 2, \ldots, N)$.

Substituting $d_{i1}^o = d_{i1} - n_{i1}$ and $\dot{d}_{i1}^o = \dot{d}_{i1} - \dot{n}_{i1}$ into (37) gives

$$\begin{aligned} 2d_{i1}\dot{d}_{i1} + 2d_{i1}\dot{r}_1^o + 2\dot{d}_{i1}r_1^o - 2\dot{\boldsymbol{s}}_i^{oT}\boldsymbol{s}_i^o + 2\dot{\boldsymbol{s}}_1^{oT}\boldsymbol{s}_1^o + 2(\dot{\boldsymbol{s}}_i^o - \dot{\boldsymbol{s}}_1^o)^T\boldsymbol{x}^o \\ + 2(\boldsymbol{s}_i^o - \boldsymbol{s}_1^o)^T\dot{\boldsymbol{x}}^o \approx 2\dot{d}_{i1}^o n_{i1} + 2d_{i1}^o \dot{n}_{i1} \end{aligned} \tag{38}$$

where the second-order noise term $n_{i1}\dot{n}_{i1}$ is also neglected.

All the equations in (36) and (38) can be combined to yield the following matrix form:

$$\boldsymbol{A}\boldsymbol{y}^o - \boldsymbol{b} \approx \boldsymbol{B}^o \boldsymbol{\Delta\alpha} \tag{39}$$

where $\boldsymbol{A}$ and $\boldsymbol{b}$ are defined in (8) and

$$\begin{aligned} \boldsymbol{y}^o &= \begin{bmatrix} \boldsymbol{x}^{oT} & \dot{\boldsymbol{x}}^{oT} & r_1^o & \dot{r}_1^o \end{bmatrix}^T \\ \boldsymbol{b}^o &= \begin{bmatrix} \boldsymbol{B}_1^o & \boldsymbol{O}_{(N-1)\times(N-1)} \\ \dot{\boldsymbol{B}}_1^o & \boldsymbol{B}_1^o \end{bmatrix} \end{aligned} \tag{40}$$

with $\boldsymbol{B}_1^o = 2\mathrm{diag}\{r_2^o, \ldots, r_N^o\}$, and $\dot{\boldsymbol{B}}_1^o = 2\mathrm{diag}\{\dot{r}_2^o, \ldots, \dot{r}_N^o\}$.

From (39), we have

$$\boldsymbol{B}^{o-1}(\boldsymbol{A}\boldsymbol{y}^o - \boldsymbol{b}) \approx \boldsymbol{\Delta\alpha}. \tag{41}$$

By comparing (41) with (5), we can approximately write the ML problem (6) as

$$\begin{aligned} &\min_{\boldsymbol{\phi}} \left[\boldsymbol{B}^{-1}(\boldsymbol{A}\boldsymbol{y} - \boldsymbol{b})\right]^T \boldsymbol{Q}_{\boldsymbol{\alpha}}^{-1} \left[\boldsymbol{B}^{-1}(\boldsymbol{A}\boldsymbol{y} - \boldsymbol{b})\right] \\ &= (\boldsymbol{A}\boldsymbol{y} - \boldsymbol{b})^T(\boldsymbol{B}\boldsymbol{Q}_{\boldsymbol{\alpha}}\boldsymbol{B}^T)^{-1}(\boldsymbol{A}\boldsymbol{y} - \boldsymbol{b}) \\ &= (\boldsymbol{A}\boldsymbol{y} - \boldsymbol{b})^T\boldsymbol{Q}^{-1}(\boldsymbol{A}\boldsymbol{y} - \boldsymbol{b}) \end{aligned} \tag{42}$$

where $\boldsymbol{Q} = \boldsymbol{B}\boldsymbol{Q}_\alpha\boldsymbol{B}^T$.

Appendix B
Proof of Proposition 2

For convenience, we rewrite (35) and (37) as follows:

$$\begin{aligned} d_{i1}^{o2} + 2d_{i1}^o r_1^o &= \|\boldsymbol{s}_i^o\|^2 - \|\boldsymbol{s}_1^o\|^2 - 2\left(\boldsymbol{s}_i^o - \boldsymbol{s}_1^o\right)^T \boldsymbol{x}^o \\ d_{i1}^o \dot{d}_{i1}^o + d_{i1}^o \dot{r}_1^o + \dot{d}_{i1}^o r_1^o &= \dot{\boldsymbol{s}}_i^{oT} \boldsymbol{s}_i^o - \dot{\boldsymbol{s}}_1^{oT} \boldsymbol{s}_1^o - \left(\dot{\boldsymbol{s}}_i^o - \dot{\boldsymbol{s}}_1^o\right)^T \boldsymbol{x}^o \\ &\quad - \left(\boldsymbol{s}_i^o - \boldsymbol{s}_1^o\right)^T \dot{\boldsymbol{x}}^o, \quad i = 2, \ldots, N. \end{aligned} \tag{43}$$

Substituting $d_{i1}^o = d_{i1} - n_{i1}$, $\dot{d}_{i1}^o = \dot{d}_{i1} - \dot{n}_{i1}$, $\boldsymbol{s}_i^o = \bar{\boldsymbol{s}}_i - \Delta\boldsymbol{s}_i$, and $\dot{\boldsymbol{s}}_i^o = \bar{\dot{\boldsymbol{s}}}_i - \Delta\dot{\boldsymbol{s}}_i$ into (43) and applying the first-order Taylor series expansion, we have [18]

$$\begin{aligned} &d_{i1}^2 - \|\bar{\boldsymbol{s}}_i\|^2 + \|\bar{\boldsymbol{s}}_1\|^2 + 2(\bar{\boldsymbol{s}}_i - \bar{\boldsymbol{s}}_1)^T\boldsymbol{x}^o + 2d_{i1}\bar{r}_1^o \\ &\approx 2\bar{r}_i^o n_{i1} + 2(\boldsymbol{x}^o - \bar{\boldsymbol{s}}_i)^T\boldsymbol{\Delta s}_i - \left[2d_{i1}\bar{\boldsymbol{u}}_1^o + 2(\boldsymbol{x}^o - \bar{\boldsymbol{s}}_1)\right]^T \boldsymbol{\Delta s}_1 \end{aligned}$$

$$2\Big[d_{i1}\dot{d}_{i1} - \bar{\dot{s}}_i^T\bar{s}_i + \bar{\dot{s}}_1^T\bar{s}_1 + (\bar{\dot{s}}_i - \bar{\dot{s}}_1)^T\boldsymbol{x}^o + (\bar{s}_i - \bar{s}_1)^T\dot{\boldsymbol{x}}^o + \dot{d}_{i1}\bar{r}_1^o + d_{i1}\bar{\dot{r}}_1^o\Big]$$
$$\approx 2\Big\{\bar{\dot{r}}_i^o n_{i1} + \bar{r}_i^o\dot{n}_{i1} - \Big[\dot{d}_{i1}\bar{\boldsymbol{u}}_1^o + d_{i1}\bar{\dot{\boldsymbol{u}}}_1^o + (\dot{\boldsymbol{x}}^o - \bar{\dot{s}}_1)\Big]^T\boldsymbol{\Delta s}_1 - [d_{i1}\bar{\boldsymbol{u}}_1^o + (\boldsymbol{x}^o - \bar{s}_1)]^T\boldsymbol{\Delta}\dot{\boldsymbol{s}}_1 + (\dot{\boldsymbol{x}}^o - \bar{\dot{s}}_i)^T\boldsymbol{\Delta s}_i + (\boldsymbol{x}^o - \bar{s}_i)^T\boldsymbol{\Delta}\dot{\boldsymbol{s}}_i\Big\}, \qquad i = 2, \ldots, N \tag{44}$$

where $\bar{r}_i^o$ and $\bar{\dot{r}}_i^o$ are defined in (26), and $\bar{\boldsymbol{u}}_i^o$ and $\bar{\dot{\boldsymbol{u}}}_i^o$ are obtained by replacing $\boldsymbol{s}_i^o$ and $\dot{\boldsymbol{s}}_i^o$ in $\boldsymbol{u}_i^o$ and $\dot{\boldsymbol{u}}_i^o$ [defined in (22)] with $\bar{s}_i$ and $\bar{\dot{s}}_i$, respectively. As done in [17] and [18], all the second-order noise terms are neglected in (44).

Collecting all the equations together and writing them in matrix form, we obtain

$$\bar{\boldsymbol{A}}\bar{\boldsymbol{y}}^o - \bar{\boldsymbol{b}} \approx \bar{\boldsymbol{B}}^o\boldsymbol{\Delta\alpha} + \bar{\boldsymbol{C}}^o\boldsymbol{\Delta\beta} \tag{45}$$

where $\bar{\boldsymbol{B}}^o$, $\bar{\boldsymbol{A}}$, $\bar{\boldsymbol{b}}$, and $\bar{\boldsymbol{y}}^o$ are, respectively, obtained by replacing $\boldsymbol{s}_i^o$ and $\dot{\boldsymbol{s}}_i^o$ with $\bar{s}_i$ and $\bar{\dot{s}}_i$ in $\boldsymbol{B}^o$, $\boldsymbol{A}$, $\boldsymbol{b}$, and $\boldsymbol{y}^o$, and $\bar{\boldsymbol{C}}^o$ is defined in (29). From (45), we have

$$\bar{\boldsymbol{B}}^{o-1}(\bar{\boldsymbol{A}}\bar{\boldsymbol{y}}^o - \bar{\boldsymbol{b}}) \approx \boldsymbol{\Delta\alpha} + \bar{\boldsymbol{B}}^{o-1}\bar{\boldsymbol{C}}^o\boldsymbol{\Delta\beta}. \tag{46}$$

According to (46), we can obtain the nonlinear WLS estimation problem (28) by assuming that $\bar{\boldsymbol{B}}^o$ and $\bar{\boldsymbol{C}}^o$ on the right-hand side are known.

In the following, we establish the relationship between the problem (28) and the original ML problem (27). To this end, we first obtain an approximate ML problem. We begin from the measurement equations (1). Substituting $\boldsymbol{s}_i^o = \bar{s}_i - \Delta\boldsymbol{s}_i$ and $\dot{\boldsymbol{s}}_i^o = \bar{\dot{s}}_i - \Delta\dot{\boldsymbol{s}}_i$ into (1) and applying the first-order Taylor series expansion, we have

$$d_{i1} - (\|\boldsymbol{x}^o - \bar{s}_i\| - \|\boldsymbol{x}^o - \bar{s}_1\|) \approx n_{i1} + \bar{\boldsymbol{u}}_i^{oT}\boldsymbol{\Delta s}_i - \bar{\boldsymbol{u}}_1^{oT}\boldsymbol{\Delta s}_1$$
$$\dot{d}_{i1} - \left[\frac{(\boldsymbol{x}^o - \bar{s}_i)^T(\dot{\boldsymbol{x}}^o - \bar{\dot{s}}_i)}{\|\boldsymbol{x}^o - \bar{s}_i\|} - \frac{(\boldsymbol{x}^o - \bar{s}_1)^T(\dot{\boldsymbol{x}}^o - \bar{\dot{s}}_1)}{\|\boldsymbol{x}^o - \bar{s}_1\|}\right] \approx \dot{n}_{i1} + \bar{\boldsymbol{u}}_i^{oT}\boldsymbol{\Delta}\dot{\boldsymbol{s}}_i - \bar{\boldsymbol{u}}_1^{oT}\boldsymbol{\Delta}\dot{\boldsymbol{s}}_1 + \bar{\dot{\boldsymbol{u}}}_i^{oT}\boldsymbol{\Delta s}_i - \bar{\dot{\boldsymbol{u}}}_1^{oT}\boldsymbol{\Delta s}_1 \tag{47}$$

where $\|\boldsymbol{x}^o - \boldsymbol{s}_i^o\| \approx \|\boldsymbol{x}^o - \bar{s}_i\| + \bar{\boldsymbol{u}}_i^{oT}\Delta\boldsymbol{s}_i$ and $(\boldsymbol{x}^o - \boldsymbol{s}_i^o)^T(\dot{\boldsymbol{x}}^o - \dot{\boldsymbol{s}}_i^o)/\|\boldsymbol{x}^o - \boldsymbol{s}_i^o\| \approx (\boldsymbol{x}^o - \bar{s}_i)^T(\dot{\boldsymbol{x}}^o - \bar{\dot{s}}_i)/\|\boldsymbol{x}^o - \bar{s}_i\| + \bar{\boldsymbol{u}}_i^{oT}\Delta\dot{\boldsymbol{s}}_i + \bar{\dot{\boldsymbol{u}}}_i^{oT}\Delta\boldsymbol{s}_i$ are used.

The equations in (47) can be written in the matrix form as

$$\tilde{\boldsymbol{d}} - \tilde{\boldsymbol{G}}\bar{\boldsymbol{\theta}}^o \approx \boldsymbol{\Delta\alpha} + \bar{\boldsymbol{U}}^o\boldsymbol{\Delta\beta} \tag{48}$$

where $\bar{\boldsymbol{U}}^o$ is obtained by replacing $\boldsymbol{s}_i^o$ and $\dot{\boldsymbol{s}}_i^o$ with $\bar{s}_i$ and $\bar{\dot{s}}_i$ in $\boldsymbol{U}^o$.

Based on (48), we can obtain the following approximate ML formulation:

$$\min_{\boldsymbol{\phi}}(\tilde{\boldsymbol{d}} - \tilde{\boldsymbol{G}}\bar{\boldsymbol{\theta}})^T(\boldsymbol{Q}_{\boldsymbol{\alpha}} + \bar{\boldsymbol{U}}^o\boldsymbol{Q}_{\boldsymbol{\beta}}\bar{\boldsymbol{U}}^{oT})^{-1}(\tilde{\boldsymbol{d}} - \tilde{\boldsymbol{G}}\bar{\boldsymbol{\theta}}). \tag{49}$$

Comparing (49) with (27) reveals that the only difference between them is that, in (27), the true values $\boldsymbol{s}_i^o$ and $\dot{\boldsymbol{s}}_i^o$ are used, whereas in (49), the estimated values $\bar{s}_i$ and $\bar{\dot{s}}_i$ are used. Thus, (49) can be seen as an approximation to (27).

It can be verified that $\bar{\boldsymbol{B}}^{o-1}\bar{\boldsymbol{C}}^o = \bar{\boldsymbol{U}}^o$ [18]. Using this and comparing (46) with (48), we obtain $\bar{\boldsymbol{B}}^{o-1}(\bar{\boldsymbol{A}}\bar{\boldsymbol{y}}^o - \bar{\boldsymbol{b}}) \approx \tilde{\boldsymbol{d}} - \tilde{\boldsymbol{G}}\bar{\boldsymbol{\theta}}^o$, from which we see that problems (28) and (49) are approximately equivalent. Hence, (28) is also an approximation to the original ML problem (27).

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers and the Associate Editor for their valuable comments, which have greatly improved the quality of the paper.

REFERENCES

[1] P. Sakarindr and N. Ansari, "Security services in group communications over wireless infrastructure, mobile ad-hoc, and wireless sensor networks," *IEEE Wireless Commun.*, vol. 14, no. 5, pp. 8–20, Oct. 2007.

[2] H. Nakayama, N. Ansari, A. Jamalipour, and N. Kato, "Fault-resilient sensing in wireless sensor networks," *Comput. Commun.*, vol. 30, no. 11/12, pp. 2375–2384, Sep. 2007.

[3] Y. Zhang, N. Ansari, and W. Su, "Optimal decision fusion based automatic modulation classification by using wireless sensor networks in multipath fading channel," in *Proc. IEEE GLOBECOM*, Houston, TX, Dec. 5–9, 2011, pp. 1–5.

[4] B. Liu, H. Chen, Z. Zhong, and H. V. Poor, "Asymmetrical round trip based synchronization-free localization in large-scale underwater sensor networks," *IEEE Trans. Wireless Commun.*, vol. 9, no. 11, pp. 3532–3542, Nov. 2010.

[5] H. Chen, M. H. T. Martins, P. Huang, H. C. So, and K. Sezaki, "Cooperative node localization for mobile sensor networks," in *Proc. Int. Conf. EUC*, Shanghai, China, Dec. 2008, vol. I, pp. 302–308.

[6] N. Patwari, J. N. Ash, S. Kyperountas, A. Hero, R. L. Moses, and N. S. Correal, "Locating the nodes: Cooperative localization in wireless sensor networks," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 54–69, Jul. 2005.

[7] J. O. Smith and J. S. Abel, "Closed-form least-squares source location estimation from range-difference measurements," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, no. 12, pp. 1661–1669, Dec. 1987.

[8] Y. T. Chan and K. C. Ho, "A simple and efficient estimator for hyperbolic location," *IEEE Trans. Signal Process.*, vol. 42, no. 8, pp. 1905–1915, Aug. 1994.

[9] Y. Huang, J. Benesty, G. W. Elko, and R. Mersereati, "Real-time passive source localization: A practical linear-correction least-squares approach," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 8, pp. 943–956, Nov. 2001.

[10] A. Beck, P. Stoica, and J. Li, "Exact and approximate solutions of source localization problems," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1770–1778, May 2008.

[11] A. N. Bishop, B. Fidan, B. D. O. Anderson, K. Dogancay, and P. N. Pathirana, "Optimal range-difference-based localization considering geometrical constraints," *IEEE Trans. Ocean. Eng.*, vol. 33, no. 3, pp. 289–301, Jul. 2008.

[12] K. W. K. Lui, F. K. W. Chan, and H. C. So, "Semidefinite programming approach for range-difference based source localization," *IEEE Trans. Signal Process.*, vol. 57, no. 4, pp. 1630–1633, Apr. 2009.

[13] K. Yang, G. Wang, and Z.-Q. Luo, "Efficient convex relaxation methods for robust target localization by a sensor network using time difference of arrivals," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2775–2784, Jul. 2009.

[14] G. Wang and H. Chen, "An importance sampling method for TDOA-based source localization," *IEEE Trans. Wireless Comm.*, vol. 10, no. 5, pp. 1560–1568, May 2011.

[15] D. J. Torrieri, "Statistical theory of passive location systems," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-20, no. 2, pp. 183–198, Mar. 1984.

[16] K. C. Ho and W. Xu, "An accurate algebraic solution for moving source location using TDOA and FDOA measurements," *IEEE Trans. Signal Process.*, vol. 52, no. 9, pp. 2453–2463, Sep. 2004.

[17] K. C. Ho and W. Xu, "Source localization using TDOA and FDOA measurements in the presence of receiver location errors: Analysis and

solution," *IEEE Trans. Signal Process.*, vol. 55, no. 2, pp. 684–696, Feb. 2007.

[18] M. Sun and K. C. Ho, "An asymptotically efficient estimator for TDOA and FDOA positioning of multiple disjoint sources in the presence of sensor location uncertainties," *IEEE Trans. Signal Process.*, vol. 59, no. 7, pp. 3434–3440, Jul. 2011.

[19] H.-W. Wei, R. Peng, Q. Wan, Z.-X. Chen, and S.-F. Ye, "Multidimensional scaling analysis for passive moving target localization with TDOA and FDOA measurements," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1677–1688, Mar. 2010.

[20] F. Quo and K. C. Ho, "A quadratic constraint solution method for TDOA and FDOA localization," in *Proc. IEEE ICASSP*, Prague, Czech Republic, May 2011, pp. 2588–2591.

[21] K. Yang, L. Jiang, and Z.-Q. Luo, "Efficient semidefinite relaxation for robust geolocation of unknown emitter by a satellite cluster using TDOA and FDOA measurements," in *Proc. IEEE ICASSP*, Prague, Czech Republic, May 2011, pp. 2584–2587.

[22] P. Biswas, T.-C. Lian, T.-C. Wang, and Y. Ye, "Semidefinite programming based algorithms for sensor network localization," *ACM Trans. Sens. Netw.*, vol. 2, no. 2, pp. 188–220, May 2006.

[23] G. Wang, "A semidefinite relaxation method for energy-based source localization in sensor networks," *IEEE Trans. Veh. Technol.*, vol. 60, no. 5, pp. 2293–2301, Jun. 2011.

[24] W.-K. Ma, T. N. Davidson, K. M. Wong, Z.-Q. Luo, and P. C. Ching, "Quasi-maximum-likelihood multiuser detection using semi-definite relaxation with application to synchronous CDMA," *IEEE Trans. Signal Process.*, vol. 50, no. 4, pp. 912–922, Apr. 2002.

[25] N. D. Sidiropoulos and Z.-Q. Luo, "A semidefinite relaxation approach to MIMO detection for higher-order constellations," *IEEE Signal Process. Lett.*, vol. 13, no. 9, pp. 525–528, Sep. 2006.

[26] C. Li, X. Wang, L. Yang, and W.-P. Zhu, "A joint source and relay power allocation scheme for a class of MIMO relay systems," *IEEE Trans. Signal Process.*, vol. 57, no. 12, pp. 4852–4860, Dec. 2009.

[27] C. Li, X. Wang, L. Yang, and W.-P. Zhu, "Joint source-and-relay power allocation in multiple-input multiple-output amplify-and-forward relay systems: A nonconvex problem and its solution," *IET Signal Process.*, vol. 5, no. 6, pp. 612–622, Sep. 2011.

[28] N. D. Sidiropoulos, T. N. Davidson, and Z.-Q. Luo, "Transmit beamforming for physical layer multicasting," *IEEE Trans. Signal Process.*, vol. 54, no. 6, pp. 2239–2251, Jun. 2006.

[29] L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM Rev.*, vol. 38, no. 1, pp. 49–95, Mar. 1996.

[30] A. I. Barvinok, "Problems of distance geometry and convex properties of quadratic maps," *Discrete Comput. Geom.*, vol. 13, no. 1, pp. 189–202, 1995.

[31] L. Yang and K. C. Ho, "An approximately efficient TDOA localization algorithm in closed-form for locating multiple disjoint sources with erroneous sensor positions," *IEEE Trans. Signal Process.*, vol. 57, no. 12, pp. 4598–4615, Dec. 2009.

[32] K. Fujisawa, M. Kojima, and K. Nakata, "Exploiting sparsity in primal-dual interior-point methods for semidefinite programming," *Math. Program.*, vol. 79, no. 1–3, pp. 235–253, Oct. 1997.

[33] M. Grant and S. Boyd, CVX: Matlab Software for Disciplined Convex Programming, 2009. [Online]. Available: http://stanford.edu/~boyd/cvx

[34] J. F. Sturm, "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones," *Optim. Methods Softw.*, vol. 11/12, pp. 625–653, Aug. 1999.

Gang Wang received the B.Eng. degree from Shandong University, Jinan, China, and the Ph.D. degree from Xidian University, Xi'an, China, in 2006 and 2011, respectively, all in electrical engineering.

Since January 2012, he has been with Ningbo University, Ningbo, China. His research interests include target detection, localization, and tracking.

Youming Li received the B.S. degree in computational mathematics from Lanzhou University, Lanzhou, China, in 1985; the M.S. degree in computational mathematics from Xi'an Jiaotong University, Xi'an, China, in 1988; and the Ph.D. degree in electrical engineering from Xidian University, Xi'an, in 1995.

From 1988 to 1998, he was with the Department of Applied Mathematics, Xidian University, where he was an Associate Professor. From 1999 to 2004, he worked with the School of Electrical and Electronics Engineering, Nanyang Technological University, Singapore; DSO National Laboratories, Singapore; and the School of Engineering, Bar-Ilan University, Ramat Gan, Israel, respectively. Since 2005, he has been with Ningbo University, Ningbo, China, where he is currently a Professor. His research interests include cognitive radio and wireless/wireline communications.

Nirwan Ansari (S'78–M'83–SM'94–F'09) received the B.S.E.E. (*summa cum laude* with a perfect GPA) from the New Jersey Institute of Technology (NJIT), Newark; the M.S.E.E. degree from the University of Michigan, Ann Arbor; and the Ph.D. degree from Purdue University, West Lafayette, IN.

In 1988, he joined the Department of Electrical and Computer Engineering, NJIT, as an Assistant Professor and has been a Full Professor since 1997. He has also assumed various administrative positions with NJIT. He has been a Visiting (Chair) Professor with several universities. He is a coauthor of *Media Access Control* and *Resource Allocation for Next Generation Passive Optical Networks* (Springer, 2013), with J. Zhang, and *Computational Intelligence for Optimization* (Springer, 1997, translated into Chinese in 2000) with E. S. H. Hou and is a Coeditor of *Neural Networks in Telecommunications* (Springer, 1994) with B. Yuhas. He is also an author of over 400 technical papers, more than one third of which were published in widely cited refereed journals/magazines. He is a holder of over 15 patents. His current research focuses on various aspects of broadband networks and multimedia communications.

Dr. Ansari served as a member-at-large of the IEEE Communications Society (ComSoc) Board of Governors and as the IEEE North Jersey Section Chair of the IEEE Region 1 Board of Governors. He was designated for two terms as a Distinguished Lecturer of ComSoc from 2006 to 2009. He has chaired ComSoc technical committees and has been actively organizing numerous IEEE international conferences/symposia/workshops, assuming various leadership roles. He has served on the editorial and advisory boards of eight journals, including as a Senior Technical Editor of the *IEEE Communications Magazine* (2006–2009). He is also a Guest Editor for a number of special issues, covering various emerging topics in communications and networking. He has received several Excellence in Teaching Awards, the IEEE Member and Geographic Activities Leadership Award in 2008, a couple of Best Paper Awards, a Thomas Alva Edison Patent Award in 2010, and the New Jersey Inventors Hall of Fame Inventor of the Year Award in 2012.