IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 4, 2018, accepted April 21, 2018, date of publication April 27, 2018, date of current version May 24, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2830814

Motion Parameter Capturing of Multiple Mobile Targets in Robotic Sensor Networks

XIAOPING WU[1,2], SHENGHUI WANG[1], HAILING FENG[1,2], JUNGUO HU[1], AND GUOYING WANG[1]

[1]School of Information Engineering, Zhejiang A & F University, Hangzhou 311300, China
[2]Zhejiang Provincial Key Laboratory of Forestry Intelligent Monitoring and Information Technology, Zhejiang A & F University, Hangzhou 311300, China

Corresponding authors: Xiaoping Wu (wuxipu@gmail.com) and Hailing Feng (hlfeng@zafu.edu.cn)

This work was supported in part by the Public Welfare Technology Research Industry Project from Zhejiang Province under Grant 2015C31004, in part by the National Natural Science Fund China Program under Grant 31570629, in part by the Zhejiang Provincial Natural Science Foundation under Grant LY16F020036 and Grant LY18F020010, in part by the Zhejiang Province Key Science and Technology Project under Grant 2015C03008 and Grant 2015C32083.

ABSTRACT A motion parameter capturing method is proposed to jointly estimate the initial positions and velocities of multiple mobile targets in robotic sensor networks. By using the time of arrival (TOA) measurements between the sensor nodes, the proposed method does not require any motion sensors. Noncooperative unconstrained linear least square (ULLS), constrained linear least square (CLLS), semidefinite programming (SDP), and mixed second-order cone semidefinite programming (SOCSDP) algorithms are designed by only exploiting the TOA measurements between anchor nodes and mobile targets. Then, the SDP and SOCSDP algorithms are extended to the cooperative approach in which the measurements between the mobile targets are also employed into the optimization model, as well as the measurements in the noncooperative approach. The simulations and real experiments show that the cooperative SDP and SOCSDP algorithms provide better performance than the noncooperative ULLS, CLLS, SDP, and SOCSDP. Compared with the linear ULLS and CLLS, the computation complexity of the convex SDP and SOCSDP is higher for a large number of variables and equality constraints. The positioning error of the SOCSDP is approximately identical to the SDP, but the SOCSDP runs faster than the SDP.

INDEX TERMS Robotic sensor network, semidefinite programming, mobile target, parameter capturing.

I. INTRODUCTION

Recent advances in computing, communication, and related technologies have resulted in significant interest in sensor networks. Sensor network is a collection of sensor nodes that have a limited amount of computational and battery capacity, the ability to communicate with each other, and the ability to sense the environment. Mobile robots equipped with sensor nodes can be used to assist with completing some complex tasks of sensor networks and improving the network performance. Robotic sensor network shows great advantages for its flexible mobility ability [1], [2]. In the robotic sensor network, the position obtaining of sensor nodes is also a critical problem for its applications, such as, data collection, coverage control [3], target tracking, and others. Conventional global positioning system (GPS) is not suitable for the position obtaining of sensor nodes due to its huge volume, energy consumption and hardware cost. To reduce the positioning cost, a few anchor nodes with know their positions are used to derive the positions of the remaining sensor nodes. By using the ranging measurement information extracted from the signaling, the unknown positions of sensor nodes are estimated.

Among all kinds of ranging methods, time of arrival (TOA) approach [4] has gained an increasing attention for high-definition localization accuracy when low complexity devices are available. Based on whether there are mobile sensor nodes or not, sensor networks are divided into stationary or robotic sensor networks. The traditional triangulation positioning method may be feasible for the stationary sensor networks, but it is not applicable for the robotic sensor networks. So a motion parameter capturing method is proposed for robotic sensor networks in this paper. The motion parameters including the initial positions and velocities of multiple mobile targets are jointly estimated when assuming the velocities of the mobile targets to be invariable. The proposed method does not require any motion sensors. By exploiting the TOA measurements between anchor nodes and mobile targets, the unconstrained linear least squares (ULLS),

2169-3536 © 2018 IEEE. Translations and content mining are permitted for academic research only.
Personal use is also permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

constrained linear least squares (CLLS), the convex semidefinite programming (SDP) and mixed second order cone semidefinite programming (SOCSDP) algorithms are proposed in the noncooperative approach. Then the convex SDP and SOCSDP algorithms are extended to cooperative approach in which the TOA measurements between the mobile targets are additionally employed into the optimization model. The main contributions of this paper are listed as follows,

1) A motion parameter capturing method is proposed by using the TOA measurements between sensor nodes. The motion parameters including the initial positions and velocities of multiple mobile targets are jointly estimated. The proposed method does not require any motion sensors only by using the TOA measurements.
2) To avoid the local convergence of conventional estimator, the linear ULLS and CLLS estimators are designed to jointly estimate the initial positions and velocities for the proposed optimization model. Then convex SDP and SOCSDP algorithms are also put forward in the noncooperative or cooperative approach.
3) The proposed system is implemented on sensor nodes and evaluated by both simulations and real-world experiments. The results show that the convex SDP and SOCSDP algorithms provide more robust performance than the linear estimators in larger noise conditions. However, the convex SDP and SOCSDP algorithms run slower than the linear estimators for the high computational complexity.

This paper mainly presents a motion parameter capturing method for the initial positions and velocities of multiple mobile targets. The rest of this paper is structured as follows. Firstly we describe the related work in Section II. Section III presents the problem specification of motion parameter capturing. Section IV in detail describes the noncooperative ULLS, CLLS, SDP, and SOCSDP algorithms. Section V introduces the cooperative SDP and SOCSDP algorithms. Section VI derives the computational complexity of these proposed algorithms. Section VII analyzes the simulation results and real experiments. The conclusions are represented in Section VIII. This paper contains a number of symbols. Following the convention, we represent the matrices as bold case letters. If we denote the matrix as $(*)$, $(*)^{-1}$ and $(*)^T$ represent the matrix inverse and transpose operator, respectively. $\|*\|$ denotes ℓ_2 norm. diag$\{*\}$ represents a diagonal matrix. $[\mathbf{A}]_{i,j}$ denotes the element at the ith row and jth column of matrix $\mathbf{A}$. For arbitrary symmetric matrices $\mathbf{A}$, $\mathbf{A} \succeq 0$ means that $\mathbf{A}$ is positive semidefinite.

II. RELATED WORK

Sensor network has attracted much attention for its wide applications, such as battlefields, habitat monitoring, medical treatments, underwater sensing, ecological sensing and others [5], [6]. To complete more complex tasks, some sensor nodes are equipped to mobile robots in the sensor network. Due to the mobility ability of robotic sensor network, it has also received significant attention for its wide applications, such as exploring an unknown environment [7], data collection [8], [9], coverage control [3], leak detection in the dangerous field [10], and others [11], [12].

Robotic sensor network has the limitations of its short communication range, limited coverage, and limited computational power, so by cooperating with other agents as a group it can perform more complex tasks, ranging from search and rescue to environmental monitoring and surveillance. In order to perform sensing, navigate reliably, or coordination using robotic sensor networks, the position obtaining of the mobile robots is important for event detection, auxiliary localization and coverage control [13]–[15]. In sensor networks, it is often that a few anchor nodes with known positions are used to determine the remaining target nodes whose positions are unknown and required to be localized [16]. By using the ranging information extracted from the signaling between the nodes, the positions of target nodes are derived. The ranging approaches locate the target nodes by measuring the Euclidean distances between the nodes with certain range techniques. Many ranging methods use techniques such as, time of arrival (TOA) [17], time difference of arrival (TDOA) [18], [19], angle of arrival (AOA) [20], received signal strength (RSS) [21], [22], and acoustic energy strength [23], [24]. Due to the influence of reflection, antenna direction and multipath fading, it is difficult to accurately describe the RSS with mathematical models, so the ranging errors is possible to be too large. Since the range of sound propagation is limited, the application of acoustic energy method is relatively small. Among the different ranging methods, the TOA ranging method has gained an increasing attention for its easy hardware implementation and high-definition ranging accuracy.

To accurately estimate the positions of the target nodes, a lot of positioning algorithms are proposed for stationary sensor networks. Maximum likelihood (ML) estimator [25], [26] is asymptotically optimal, but the performance of the ML estimator highly relies on the initial solution provided for the iterative solver. To overcome the shortcoming of ML estimator, the linear estimator is proposed by converting the optimization problem of the ML estimator into a linear model. The linear estimator represents the position estimates as closed-form solutions which do not require the initialization [27]–[29]. However, the linear estimator is not applicable for the cooperative localization which provides better accuracy performance compared to the noncooperative localization [30], [31].

To improve the performance of the linear estimator, convex optimization methods are also proposed by relaxing the cost function of ML estimator into the convex optimization problem and efficiently solved by using existing algorithms such as interior point methods which also do not require any initial solutions [32], [33]. The convex optimization algorithms can be realized by the cooperative approach in which not only the measurements between anchor nodes and target nodes but

also the measurements between target nodes are employed into the optimization model. However, only the measurements between anchor node and target node are exploited in the noncooperative approach. Compared with the noncooperative approach, the cooperative convex optimization methods provide better accuracy performance [34], [35]. Due to the convex relaxation and the equivalent approximation, the solution of the convex optimization is sub-optimal and cannot achieve the best possible performance in the presence of noise. On the other hand, the computational complexity of convex optimization is high for a large number of variables and equality constraints produced in the relaxation of convex optimization.

The convex optimization can be realized by semidefinite programming (SDP) [36]–[38], second order cone programming (SOCP) [39], [40], or mixed second order cone semidefinite programming (SOCSDP) algorithm for the position estimates in sensor networks. The SDP algorithm provides a tighter relaxation and hence results in a better localization accuracy compared with the SOCP. However, the SDP algorithm runs slower than the SOCP for a number of variables and equality constraints produced in the process of convex relaxation. The lower complexity of SOCP is because for a given problem, the number and size of variables and constraints required for solving SOCP are smaller than those required for solving SDP. The mixed SOCSDP algorithm trades off the positioning accuracy and computational complexity.

In robotic sensor networks, the current positions of the mobile targets are also calculated by using the ranging measurements of the anchor nodes to the mobile targets. Moreover, the future positions of the mobile targets can also be predicted accurately by using some methods such as, dead-reckoning [41], simultaneous localization and mapping (SLAM) [42], and filtering methods [43]–[45]. In the dead-reckoning methods, the positions of the mobile targets are calculated by using the wheel rotation measurements when assuming the start positions to be known. However, the dead-reckoning methods require extra motion sensor and would be limited due to the accumulating errors especially when the wheels would skid under the unsteadily working conditions. The accumulating errors would be largen over time, so the dead-reckoning methods are not suitable for the exploring of far mobile distances.

The SLAM method has been extensively proposed to explore the environment map by availing of the mobility ability in robotic sensor networks. In [46], multitarget SLAM technique is designed to estimate the positions of the sensors in a sensor network. In [47] and [48], multipath propagation is used to explore the environment map, which would be served by the simultaneously navigating system. However, the main disadvantage of most SLAM methods is the high computational complexity, which makes them less efficient specially in larger multirobot networks.

When all kinds of ranging data are noisy, there are some position prediction methods which include Kalman filter (KF) [49], [50], extended Kalman filter (EKF), and particle filter (PF) [51]. KF filter updates the estimates of the mobile targets with continuous prediction and correction to improve the tracking accuracy of the mobile targets, only when the uncertainties are gaussian and the system dynamics are linear. To extend the prediction model to nonlinear model, EKF filter is proposed by linearizing the nonlinear measurements. However, KF and EKF can only be applicable for the processing of gaussian noise. So the PF filter is proposed to predict and update the mobile target positions by constantly resampling when there are non-gaussian noises.

Most of above proposed methods provide the accurate position estimates of the mobile targets based on their motion and sensor data. Moreover, to improve the performance of the position estimates, [52] introduced the linear estimator to estimate the relative node positions by given pair-wise range measurement and relative speed measurement between communicating nodes. In [38], the mobility-aided SDP algorithm is proposed by considering the velocity measurement to be available. As is known that the velocity measurement or the obtaining of the motion data requires extra hardware devices. Furthermore, the velocity measurements are subject to measurement errors, so the velocity estimate is not enough accurate especially when there are larger measurement errors.

III. PROBLEM SPECIFICATION

In a deployed region of robotic sensor network, there are M anchor nodes with known position coordinates which are denoted by $\mathbf{a}_i = [a_i \quad b_i]^T \in \mathbb{R}^2$, $i = 1, 2, \ldots, M$. In the same region, N mobile targets equipped with sensor nodes start from the initial positions $\mathbf{x}_{j,0} = [x_{j,0} \quad y_{j,0}]^T \in \mathbb{R}^2$ with constant velocity $\mathbf{v}_j = [v_{j,x} \quad v_{j,y}]^T \in \mathbb{R}^2$, $j = M+1, M+2, \ldots, M+N$. As Fig. 1 shows that there are N ($N = 3$) mobile targets in the region surrounded by four anchor nodes ($M = 4$). At the same time, the TOAs are measured between anchor nodes and mobile targets at sampling period T. So at the kth interval, the TOA measurement $t_{i,j,k}$ between anchor node i and mobile target j can be obtained that

$$t_{i,j,k} = \frac{\| \mathbf{a}_i - \mathbf{x}_{j,k} \|}{c} + \varepsilon_{i,j,k} \tag{1}$$

where $i = 1, 2, \ldots, M$, $j = M+1, M+2, \ldots, M+N$, $k = 1, 2, \ldots, K$. K is the total number of samples. It is noted not of all links between anchor node i and mobile target j are measurable due to the limitation of communication range. $\varepsilon_{i,j,k}$ is noise which is modeled as uncorrelated zero-mean Gaussian variable with variance $\delta^2_{i,j,k}$. c is propagation speed of electromagnetic wave. $\| \mathbf{a}_i - \mathbf{x}_{j,k} \|$ denotes the distance between anchor node i and mobile target j at the kth interval period. On the other hand, the TOAs between the mobile targets each other are also measured and given by

$$t_{i,j,k} = \frac{\| \mathbf{x}_{i,k} - \mathbf{x}_{j,k} \|}{c} + \varepsilon_{i,j,k} \tag{2}$$

where $i = M+1, M+2, \ldots, M+N$, $j = M+1, M+2, \ldots, M+N$, $k = 1, 2, \ldots, K$. $\| \mathbf{x}_{i,k} - \mathbf{x}_{j,k} \|$ represents the distance between mobile target i and j at the kth sampling

FIGURE 1. Mobile targets and anchor nodes.

period ($i \neq j$). The mobile targets start from the initial positions $\mathbf{x}_{j,0} = [x_{j,0} \quad y_{j,0}]^T$ with a constant velocity $\mathbf{v}_j = [v_{j,x} \quad v_{j,y}]^T \in \mathbb{R}^2$, so at the kth sampling period the position equations of mobile target i and j are obtained with

$$\begin{cases} \mathbf{x}_{i,k} = \mathbf{x}_{i,0} + kT\mathbf{v}_i \\ \mathbf{x}_{j,k} = \mathbf{x}_{j,0} + kT\mathbf{v}_j \end{cases} \tag{3}$$

where $i = M+1, M+2, \ldots, M+N$, $j = M+1, M+2, \ldots, M+N$, $k = 1, 2, \ldots, K$. In above proposed model, the goal of motion parameter capturing is to estimate the initial position $\mathbf{x}_{j,0}$ and the velocity $\mathbf{v}_j$ of mobile target j by using the TOA measurements, $j = M+1, M+2, \ldots, M+N$. Then the paths of these mobile targets are traced according to the position equations of mobile targets. Let $\mathbf{e}_k = [1 \quad kT]^T$ and $\mathbf{U}_j = [\mathbf{x}_{j,0} \quad \mathbf{v}_j]^T$, the measurement equations can also be rewritten as

$$\begin{cases} t_{i,j,k} = \dfrac{\parallel \mathbf{a}_i - \mathbf{e}_k^T\mathbf{U}_j \parallel}{c} + \varepsilon_{i,j,k} \quad i = 1, 2, \ldots, M \\ t_{i,j,k} = \dfrac{\parallel \mathbf{e}_k^T(\mathbf{U}_i - \mathbf{U}_j) \parallel}{c} + \varepsilon_{i,j,k} \\ \quad i = M+1, M+2, \ldots, M+N \end{cases} \tag{4}$$

where $j = M+1, M+2, \ldots, M+N, k = 1, 2, \ldots, K$. Based on whether the TOAs between the mobile targets each other are exploited or not, the parameter capturing is also divided into noncooperative or cooperative approach. Only the TOA measurements between the anchor nodes and the mobile targets are employed into the estimation problem, but the TOAs between the mobile targets each other are not considered in the noncooperative approach. However, in the cooperative approach, the TOAs between the mobile targets are used to improve the estimation performance. For the convenience of description, $\mathcal{S} = \{j | j = M+1, M+2, \ldots, M+N\}$ represents the set of indices of the mobile targets. Let $\mathcal{A}_j$ be the set of the indices of the anchor nodes connected to mobile target j and $\mathcal{B}_j$ be the set of indices of the mobile targets connected to mobile target j. So $\mathcal{E}_j = \mathcal{A}_j \cup \mathcal{B}_j$ represents the set of the indices of all links connected to mobile target j. In this section, these defined notations are listed in Tab. 1.

TABLE 1. Notation descriptions used in this paper.

Notations	Descriptions
$\mathbf{a}_i$	known position of anchor node i
$\mathbf{x}_{j,k}$	position of mobile target j at the kth interval period
$\mathbf{x}_{j,0}$	initial position of mobile target j
$\mathbf{v}_j$	velocity of mobile target j
$t_{i,j,k}$	TOA measurement between anchor node i and mobile target j at the kth interval period
$\varepsilon_{i,j,k}$	measurement noise between anchor node i and mobile target j at the kth interval period
c	propagation speed of electromagnetic wave
M	number of anchor nodes
N	number of mobile targets
K	number of samples
T	sampling period

IV. NONCOOPERATIVE APPROACH

When mobile target j starts from the initial positions $\mathbf{x}_{j,0}$ with a constant velocity $\mathbf{v}_j$, the TOAs between the anchor nodes and the mobile targets are also measured. All TOA measurements are sent to the central computing unit for parameter estimation. In the noncooperative approach, only the TOAs between anchor node and mobile target are employed into the parameter estimation. Considering a single mobile target j, the ML estimator of the parameter capturing problem is obtained by the following optimization

$$\arg\min_{\mathbf{U}_j} \sum_{i \in \mathcal{A}_j} \sum_{k=1}^{K} \frac{1}{\delta_{i,j,k}^2} (t_{i,j,k} - \frac{\parallel \mathbf{a}_i - \mathbf{e}_k^T\mathbf{U}_j \parallel}{c})^2 \tag{5}$$

where $j = M+1, M+2, \ldots, M+N$. By using the optimization method of (5), the unknown parameter $\mathbf{U}_j$ is derived one by one. The solution of ML estimator should be approximately solved by numerical techniques which require an initial solution. When the initial solution is not enough close to the true solution, local convergence may occur and convergence is not guaranteed. To avoid the selecting problems of initial solution, the linear estimator and convex algorithm are introduced to uniquely derive the motion parameters including the initial positions and velocities of the mobile targets.

A. LINEAR ESTIMATOR

To capture the motion parameters of mobile targets, the linear estimator is firstly proposed by converting the optimization problem of (5) into unconstrained linear least squares (ULLS) estimation. Then the constrained linear least squares (CLLS) method is designed by availing of the constrained condition. The ULLS estimator represents the estimation value as an analytical closed-form solution by using all TOA measurements between anchor nodes and mobile targets. To obtain the closed-form solution to the parameter capturing problem, the first expression of (4) is equivalently transformed into

$$\parallel \mathbf{a}_i - \mathbf{e}_k^T\mathbf{U}_j \parallel = c(t_{i,j,k} - \varepsilon_{i,j,k}) \tag{6}$$

where $i \in \mathcal{A}_j$. Squaring both sides of (6) and neglecting the second order high term, we can obtain that

$$-2\mathbf{a}_i\mathbf{e}_k^T\mathbf{U}_j + \mathbf{U}_j^T\mathbf{e}_k\mathbf{e}_k^T\mathbf{U}_j = c^2t_{i,j,k}^2 - \mathbf{a}_i^T\mathbf{a}_i - 2c^2t_{i,j,k}\varepsilon_{i,j,k} \tag{7}$$

where $i \in \mathcal{A}_j$, $j = M+1, M+2, \ldots, M+N$, $k = 1, 2, \ldots, K$. We define a new unknown vector $\eta_j = [\mathbf{U}_j^T \quad \mathbf{U}_j^T\mathbf{U}_j \quad \mathbf{x}_{j,0}^T\mathbf{v}_j]^T \in \mathbb{R}^7$. By stacking all TOA measurements of mobile target j in an ascending of i and k, the matrix form of (7) is written as

$$\mathbf{A}_j\eta_j = \mathbf{b}_j + \varepsilon_j \tag{8}$$

where the row vector of $\mathbf{A}_j$ is equal to $[-2\mathbf{a}_i\mathbf{e}_k^T \quad \mathbf{e}_k^T\mathbf{e}_k \quad 2kT]$, the row elements of $\mathbf{b}_j$ and ε_j are equal to $[c^2t_{i,j,k}^2 - \mathbf{a}_i^T\mathbf{a}_i]$ and $[-2c^2t_{i,j,k}\varepsilon_{i,j,k}]$, respectively. The covariance of noise ε_j, Σ_j is obtained with

$$\Sigma_j = \mathrm{diag}\{4c^4t_{i,j,k}^2\delta_{i,j,k}^2\} \quad i \in \mathcal{A}_j \tag{9}$$

So the weighting least square (WLS) solution to (9) is obtained by

$$\eta_j = (\mathbf{A}_j^T\Sigma_j^{-1}\mathbf{A}_j)^{-1}\mathbf{A}_j^T\Sigma_j^{-1}\mathbf{b}_j \tag{10}$$

When the mobile target j is fully connected to all anchor nodes, $\mathbf{A}_j \in \mathbb{R}^{ML\times 7}$, $\mathbf{b}_j \in \mathbb{R}^{KM}$, $\varepsilon_j \in \mathbb{R}^{KM}$ and $\Sigma_j \in \mathbb{R}^{KM\times KM}$.

The estimation error of η_j is denoted by $\Delta\eta_j$, which is given by

$$\Delta\eta_j = (\mathbf{A}_j^T\Sigma_j^{-1}\mathbf{A}_j)^{-1}\mathbf{A}_j^T\Sigma_j^{-1}\varepsilon_j \tag{11}$$

So the covariance of estimation error $\Delta\eta_j$, $\mathrm{cov}(\Delta\eta_j)$ is obtained with

$$\mathrm{cov}(\Delta\eta_j) = (\mathbf{A}_j^T\Sigma_j^{-1}\mathbf{A}_j)^{-1} \tag{12}$$

Extracting from the estimated η_j, we can obtain the estimation parameters of the initial position and the velocity of mobile target j, $j = M+1, M+2, \ldots, M+N$. Above calculation does not consider the constraint relationship of the elements in the vector η_j each other. So it is called as unconstrained linear least square (ULLS) method for motion parameter capturing problem. Since the variable relaxation is performed in the ULLS method, the estimation error is large. To reduce the estimation error of motion parameters, the constrained linear least square (CLLS) is designed by considering the constraint relationship of the elements in η_j, $j = M+1, M+2, \ldots, M+N$.

The true initial position and velocity of mobile target j are denoted by $\hat{\mathbf{x}}_{j,0} = [\hat{x}_{j,0} \quad \hat{y}_{j,0}]^T$ and $\hat{\mathbf{v}}_j = [\hat{v}_{j,x} \quad \hat{v}_{j,y}]^T$, respectively. Considering the error between the estimated and the true, we can obtain that

$$\begin{cases} \hat{x}_{j,0}^2 = [\eta_j(1) + \Delta\eta_j(1)]^2 \approx \eta_j(1)^2 + 2\eta_j(1)\Delta\eta_j(1) \\ \hat{y}_{j,0}^2 = [\eta_j(2) + \Delta\eta_j(2)]^2 \approx \eta_j(2)^2 + 2\eta_j(2)\Delta\eta_j(2) \\ \hat{v}_{j,x}^2 = [\eta_j(3) + \Delta\eta_j(3)]^2 \approx \eta_j(3)^2 + 2\eta_j(3)\Delta\eta_j(3) \\ \hat{v}_{j,y}^2 = [\eta_j(4) + \Delta\eta_j(4)]^2 \approx \eta_j(4)^2 + 2\eta_j(4)\Delta\eta_j(4) \\ \hat{x}_{j,0}^2 + \hat{y}_{j,0}^2 = \eta_j(5) + \Delta\eta_j(5) \\ \hat{v}_{j,x}^2 + \hat{v}_{j,y}^2 = \eta_j(6) + \Delta\eta_j(6) \end{cases} \tag{13}$$

where $\eta_j(l)$ or $\Delta\eta_j(l)$ represents the lth element of the vector η_j or $\Delta\eta_j$, $l = 1, 2, \ldots, 6$. So the matrix form of the expression (13) is written as

$$\mathbf{G}_j\theta_j = \mathbf{h}_j + \gamma_j \tag{14}$$

where $\theta_j = [\hat{x}_{j,0}^2 \quad \hat{y}_{j,0}^2 \quad \hat{v}_{j,x}^2 \quad \hat{v}_{j,y}^2]^T$ denotes the unknown vector of mobile target j which is required to be estimated, $j = M+1, M+2, \ldots, M+N$. The others in (14) are defined as

$$\begin{cases} \mathbf{G}_j = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}^T \\ \mathbf{h}_j = \begin{bmatrix} \eta_j(1)^2 & \eta_j(2)^2 & \eta_j(3)^2 & \eta_j(4)^2 & \eta_j(5) & \eta_j(6) \end{bmatrix}^T \\ \mathbf{L}_j = \mathrm{diag}\begin{Bmatrix} 2\eta_j(1) & 2\eta_j(2) & 2\eta_j(3) & 2\eta_j(4) & 1 & 1 \end{Bmatrix} \\ \gamma_j = \mathbf{L}_j\Delta\eta_j \end{cases} \tag{15}$$

So the WLS solution to (14) is

$$\theta_j = (\mathbf{G}_j^T\Sigma_{\gamma,j}^{-1}\mathbf{G}_j)^{-1}\mathbf{G}_j^T\Sigma_{\gamma,j}^{-1}\mathbf{h}_j \tag{16}$$

where $\Sigma_{\gamma,j}$ is given by

$$\Sigma_{\gamma,j} = \mathrm{E}(\gamma_j^T\gamma_j) \tag{17}$$

Since $\gamma_j = \mathbf{L}_j\Delta\eta_j$, $\Sigma_{\gamma,j}$ is also obtained by

$$\Sigma_{\gamma,j} = \mathbf{L}_j^T\mathrm{cov}(\Delta\eta_j)\mathbf{L}_j = \mathbf{L}_j^T(\mathbf{A}_j^T\Sigma_j^{-1}\mathbf{A}_j)^{-1}\mathbf{L}_j \tag{18}$$

By using the definition of the vector $\mathbf{U}_j$, the more accurate motion parameter estimates of mobile target j can be represented by

$$\begin{cases} \hat{\mathbf{x}}_{j,0} = \mathrm{sign}\{\mathrm{diag}\{\eta_j(1:2)\}\}\sqrt{\theta_j(1:2)} \\ \hat{\mathbf{v}}_j = \mathrm{sign}\{\mathrm{diag}\{\eta_j(3:4)\}\}\sqrt{\theta_j(3:4)} \end{cases} \tag{19}$$

$j = M+1, M+2, \ldots, M+N$. The initial position and velocity of each mobile target are estimated by (19) which obtains more accurate estimates by availing of the constrained conditions. So it is called as constrained linear least square (CLLS) method of motion parameter capturing problem.

B. SDP ALGORITHM

To avoid the problem of divergence, an another approach is SDP algorithm which can be efficiently solved by using existing algorithms such as interior point methods. Similar to the proposed linear estimator, each mobile target is to be estimated for its motion parameters including the initial position and the velocity. Considering the mobile target j and its connected anchor nodes $i \in \mathcal{A}_j$, the ML estimator is firstly rewritten by the equivalent transformation. Let $d_{i,j,k} = \| \mathbf{a}_i - \mathbf{x}_{j,k} \|$ which denotes the distance between anchor node i and mobile target j, so (1) is rewritten as

$$d_{i,j,k} = c(t_{i,j,k} - \varepsilon_{i,j,k}) \tag{20}$$

By squaring both sides of (20) and neglecting the second order term, (20) is rewritten as

$$d_{i,j,k}^2 = c^2 t_{i,j,k}^2 - 2c^2 t_{i,j,k}\varepsilon_{i,j,k} \tag{21}$$

where $i \in \mathcal{A}_j, j = M+1, M+2, \ldots, M+N, k = 1, 2, \ldots, K$. So the ML estimator is expressed by

$$\begin{aligned} &\arg\min_{\mathbf{U}_j} \sum_{i\in\mathcal{A}_j}\sum_{k=1}^{K} \frac{1}{t_{i,j,k}^2\delta_{i,j,k}^2}(d_{i,j,k}^2 - c^2 t_{i,j,k}^2)^2 \\ &\text{s.t. } d_{i,j,k} = \| \mathbf{a}_i - \mathbf{x}_{j,k} \| \end{aligned} \tag{22}$$

To obtain the convex optimization form, a new matrix $\mathbf{Z}_j$ is defined as

$$\mathbf{Z}_j = \begin{bmatrix} \mathbf{I}_2 & \mathbf{U}_j^T \\ \mathbf{U}_j & \mathbf{U}_j\mathbf{U}_j^T \end{bmatrix} \tag{23}$$

where $j = M+1, M+2, \ldots, M+N$. It is found that

$$d_{i,j,k}^2 = \begin{bmatrix} \mathbf{a}_i \\ -\mathbf{e}_k \end{bmatrix}^T \mathbf{Z}_j \begin{bmatrix} \mathbf{a}_i \\ -\mathbf{e}_k \end{bmatrix} \quad i \in \mathcal{A}_j \tag{24}$$

So the optimization problem of (22) can also be rewritten as

$$\begin{aligned} &\min_{\mathbf{Z}_j,\{\alpha_{i,j,k}\},\{d_{i,j,k}\}} \sum_{i\in\mathcal{A}_j}\sum_{k=1}^{K} \alpha_{i,j,k} \\ &\text{s.t. } \frac{1}{t_{i,j,k}^2\delta_{i,j,k}^2}(d_{i,j,k}^2 - c^2 t_{i,j,k}^2)^2 \le \alpha_{i,j,k} \\ &\quad d_{i,j,k}^2 = \begin{bmatrix} \mathbf{a}_i \\ -\mathbf{e}_k \end{bmatrix}^T \mathbf{Z}_j \begin{bmatrix} \mathbf{a}_i \\ -\mathbf{e}_k \end{bmatrix} \quad i \in \mathcal{A}_j \end{aligned} \tag{25}$$

where $j = M+1, M+2, \ldots, M+N$. By relaxing $\mathbf{Z}_j \succeq \mathbf{0}_4$, the nonconvex optimization problem of (25) is relaxed into convex SDP form

$$\begin{aligned} &\min_{\mathbf{Z}_j,\{\alpha_{i,j,k}\},\{d_{i,j,k}\}} \sum_{i\in\mathcal{A}_j}\sum_{l=K}^{L} \alpha_{i,j,k} \\ &\text{s.t. } \begin{bmatrix} \alpha_{i,j,k} & \frac{d_{i,j}^2 - c^2 t_{i,j,k}^2}{t_{i,j,k}\delta_{i,j,k}} \\ \frac{d_{i,j}^2 - c^2 t_{i,j,k}^2}{t_{i,j,k}\delta_{i,j,k}} & 1 \end{bmatrix} \succeq \mathbf{0}_2 \\ &\quad d_{i,j,k}^2 = \begin{bmatrix} \mathbf{a}_i \\ -\mathbf{e}_k \end{bmatrix}^T \mathbf{Z}_j \begin{bmatrix} \mathbf{a}_i \\ -\mathbf{e}_k \end{bmatrix} \quad i \in \mathcal{A}_j \\ &\quad \mathbf{Z} \succeq \mathbf{0}_4 \end{aligned} \tag{26}$$

The proposed SDP algorithm provides a convex optimization solution. However, the SDP algorithm runs slower due to a larger of variables and equality constraints. The set of all possible solutions obtained from SDP relaxation is a subset of all possible solutions that can be obtained by SOCP. In other words, SDP provides a tighter relaxation than SOCP solution since the SOCP solution set includes the SDP. To reduce the computational complexity, here the robust SOCP algorithm is provided for the SDP optimization model. To obtain the SOCP form, a new variable $\lambda_{i,j,k}$ is defined and written as

$$\lambda_{i,j,k} = \frac{d_{i,j}^2 - c^2 t_{i,j,k}^2}{t_{i,j,k}\delta_{i,j,k}} \tag{27}$$

Then the inequality constraint of (25) is transformed as

$$\| \lambda_j \| \le \phi_j \tag{28}$$

where $\lambda_j = [\lambda_{i,j,k} | i \in \mathcal{A}_j, k = 1, 2 \ldots, K]$, the new variable ϕ_j is also defined by

$$\phi_j = \sum_{i\in\mathcal{A}_j}\sum_{k=1}^{K} \alpha_{i,j,k} \tag{29}$$

So the SDP optimization problem of (15) can be equivalently written as its epigraph form

$$\begin{aligned} &\min_{\mathbf{Z}_j,\lambda_j,\{d_{i,j}\},\phi_j} \phi_j \\ &\text{s.t. } \| \lambda_j \| \le \phi_j \\ &\quad \frac{d_{i,j,k}^2 - c^2 t_{i,j,k}^2}{t_{i,j,k}\delta_{i,j,k}} = \lambda_{i,j,k} \\ &\quad d_{i,j,k}^2 = \begin{bmatrix} \mathbf{a}_i \\ -\mathbf{e}_k \end{bmatrix}^T \mathbf{Z}_j \begin{bmatrix} \mathbf{a}_i \\ -\mathbf{e}_k \end{bmatrix} \quad i \in \mathcal{A}_j \\ &\quad \mathbf{Z} \succeq \mathbf{0}_4 \end{aligned} \tag{30}$$

where the SOCP and SDP constraints are mixed by only using the TOA measurements between the anchor nodes and the mobile targets, so the solution to (30) is also called the noncooperative SOCSDP algorithm. Due to the using of the similar optimization function, the solution to noncooperative SOCSDP algorithm is very close to that of the noncooperative SDP algorithm.

V. COOPERATIVE APPROACH

To improve the performance of the motion parameter capturing, the TOA measurements between the mobile targets each other are also employed in the cooperative approach. In the noncooperative approach, the motion parameters may be difficult to be estimated for less measurements. Compared with the noncooperative approach, the cooperative approach provides robust solutions for the parameter capturing since the TOAs between the mobile targets are also employed into the optimization model. In the following, the cooperative approach is designed for the motion parameter capturing by using not only the measurements between the anchor nodes and the mobile targets ($i \in \mathcal{A}_j$) but also the measurements between the mobile targets each other ($i \in \mathcal{B}_j$).

When the motion parameters of multiple mobile targets are defined as $\mathbf{U} = [\mathbf{U}_{M+1}^T \quad \mathbf{U}_{M+2}^T \quad \ldots \quad \mathbf{U}_{M+N}^T]^T$, the ML estimator for cooperative approach can be expressed as the following optimization problem

$$\begin{aligned} &\arg\min_{\mathbf{U}} \sum_{i\in\mathcal{A}_j}\sum_{j=M+1}^{M+N}\sum_{k=1}^{K} \frac{1}{\delta_{i,j,k}^2}\left(t_{i,j,k} - \frac{\| \mathbf{a}_i - \mathbf{e}_k^T\mathbf{U}_j \|}{c}\right)^2 \\ &+ \sum_{i\in\mathcal{B}_j}\sum_{j=M+1}^{M+N}\sum_{k=1}^{K} \frac{1}{\delta_{i,j,k}^2}\left(t_{i,j,k} - \frac{\| \mathbf{e}_k^T(\mathbf{U}_i - \mathbf{U}_j) \|}{c}\right)^2 \end{aligned} \tag{31}$$

Similarly, the solution of ML estimator is not available and should be approximately solved by numerical techniques which require the initialization. When some approximations are used to linearize the nonlinear measurement equation, the linear estimator obtains an analytical closed-form solution to the unknown parameter estimates. The exploiting of linear estimator is difficult in the cooperative approach, since the TOA measurement equations can not to be linearized. So semidefinite programming algorithm is proposed for the cooperative approach by using the additional TOA measurements between the mobile targets each other.

Considering not only the measurements between the anchor nodes and the mobile targets ($i \in \mathcal{A}_j$) but also the measurements between the mobile targets ($i \in \mathcal{B}_j$), we obtain the ML estimator for the cooperative approach with

$$\begin{aligned} \arg\min_{\mathbf{U}} & \sum_{i\in\mathcal{E}_j}\sum_{j=M+1}^{M+N}\sum_{k=1}^{K}\frac{1}{t_{i,j,k}^2\delta_{i,j,k}^2}(d_{i,j,k}^2-c^2t_{i,j,k}^2)^2 \\ \text{s.t. } d_{i,j,k} &= \| \mathbf{a}_i-\mathbf{x}_{j,k}\| \quad i\in\mathcal{A}_j \\ d_{i,j,k} &= \| \mathbf{x}_{i,k}-\mathbf{x}_{j,k}\| \quad i\in\mathcal{B}_j \end{aligned} \tag{32}$$

To design the convex form for the cooperative approach, we also should convert the nonconvex optimization model into convex problem. Firstly, we also define an unknown matrix $\mathbf{Z}$, which is given by

$$\mathbf{Z}=\begin{bmatrix}\mathbf{I}_2 & \mathbf{U}^T\\ \mathbf{U} & \mathbf{U}\mathbf{U}^T\end{bmatrix} \tag{33}$$

Similarly, it is not difficultly found that

$$\begin{cases} d_{i,j,k}^2=\begin{bmatrix}\mathbf{a}_i\\ -\mathbf{s}_{j,k}\end{bmatrix}^T\mathbf{Z}\begin{bmatrix}\mathbf{a}_i\\ -\mathbf{s}_{j,k}\end{bmatrix} & i\in\mathcal{A}_j \\ d_{i,j,k}^2=\begin{bmatrix}\mathbf{0}_2\\ \mathbf{s}_{i,k}-\mathbf{s}_{j,k}\end{bmatrix}^T\mathbf{Z}\begin{bmatrix}\mathbf{0}_2\\ \mathbf{s}_{i,k}-\mathbf{s}_{j,k}\end{bmatrix} & i\in\mathcal{B}_j \end{cases} \tag{34}$$

where $i=M+1,M+2,\ldots,M+N, j=M+1,M+2,\ldots,M+N$,$s_{i,k}$ and $s_{j,k}$ are obtained with

$$\begin{cases} \mathbf{s}_{i,k}=[\underbrace{\mathbf{0}_2}_{1}\ \cdots\ \underbrace{\mathbf{e}_k}_{i-M}\ \cdots\ \underbrace{\mathbf{0}_2}_{N}]^T \\ \mathbf{s}_{j,k}=[\underbrace{\mathbf{0}_2}_{1}\ \cdots\ \underbrace{\mathbf{e}_k}_{j-M}\ \cdots\ \underbrace{\mathbf{0}_2}_{N}]^T \end{cases} \tag{35}$$

Then the optimization problem of (32) is equivalent to

$$\begin{aligned} \min_{\mathbf{Z},\{\alpha_{i,j,k}\},\{d_{i,j,k}\}} & \sum_{i\in\mathcal{E}_j}\sum_{j=M+1}^{M+N}\sum_{k=1}^{K}\alpha_{i,j,k} \\ \text{s.t. } & \frac{1}{t_{i,j,k}^2\delta_{i,j,k}^2}(d_{i,j,k}^2-c^2t_{i,j,k}^2)^2\le\alpha_{i,j,k} \\ & d_{i,j,k}^2=\begin{bmatrix}\mathbf{a}_i\\ -\mathbf{s}_{j,k}\end{bmatrix}^T\mathbf{Z}\begin{bmatrix}\mathbf{a}_i\\ -\mathbf{s}_{j,k}\end{bmatrix} \quad i\in\mathcal{A}_j \\ & d_{i,j,k}^2=\begin{bmatrix}\mathbf{0}_2\\ \mathbf{s}_{i,k}-\mathbf{s}_{j,k}\end{bmatrix}^T\mathbf{Z}\begin{bmatrix}\mathbf{0}_2\\ \mathbf{s}_{i,k}-\mathbf{s}_{j,k}\end{bmatrix} \quad i\in\mathcal{B}_j \end{aligned} \tag{36}$$

When relaxing $\mathbf{Z}\succeq\mathbf{0}_{2N+2}$, (36) is converted into convex SDP form

$$\begin{aligned} \min_{\mathbf{Z},\{\alpha_{i,j,k}\},\{d_{i,j,k}\}} & \sum_{i\in\mathcal{E}_j}\sum_{j=M+1}^{M+N}\sum_{k=1}^{K}\alpha_{i,j,k} \\ \text{s.t. } & \begin{bmatrix}\alpha_{i,j,k} & \frac{d_{i,j}^2-c^2t_{i,j,k}^2}{t_{i,j,k}\delta_{i,j,k}}\\ \frac{d_{i,j}^2-c^2t_{i,j,k}^2}{t_{i,j,k}\delta_{i,j,k}} & 1\end{bmatrix}\succeq\mathbf{0}_2 \\ & d_{i,j,k}^2=\begin{bmatrix}\mathbf{a}_i\\ -\mathbf{s}_{j,k}\end{bmatrix}^T\mathbf{Z}\begin{bmatrix}\mathbf{a}_i\\ -\mathbf{s}_{j,k}\end{bmatrix} \quad i\in\mathcal{A}_j \\ & d_{i,j,k}^2=\begin{bmatrix}\mathbf{0}_2\\ \mathbf{s}_{i,k}-\mathbf{s}_{j,k}\end{bmatrix}^T\mathbf{Z}\begin{bmatrix}\mathbf{0}_2\\ \mathbf{s}_{i,k}-\mathbf{s}_{j,k}\end{bmatrix} \quad i\in\mathcal{B}_j \\ & \mathbf{Z}\succeq\mathbf{0}_{2N+2} \end{aligned} \tag{37}$$

To reduce the computational complexity and obtain the convex SOCP form, we use the same definition of (27). Then the inequality constraints in (36) is rewritten as

$$\| \lambda \| \le \phi \tag{38}$$

where $\lambda=[\lambda_{i,j,k}|i\in\mathcal{E}_j, j=M+1,M+2\ldots,M+N,k=1,2\ldots,K]$, ϕ is defined by

$$\phi=\sum_{i\in\mathcal{E}_j}\sum_{j=M+1}^{M+N}\sum_{k=1}^{K}\alpha_{i,j,k} \tag{39}$$

Similar to the noncooperative approach, (37) is also rewritten as its convex SOCSDP form

$$\begin{aligned} \min_{\mathbf{Z},\lambda,\{d_{i,j,k}\},\phi} & \ \phi \\ \text{s.t. } & \| \lambda \| \le \phi \\ & \frac{d_{i,j,k}^2-c^2t_{i,j,k}^2}{t_{i,j,k}\delta_{i,j,k}}=\lambda_{i,j,k} \\ & d_{i,j,k}^2=\begin{bmatrix}\mathbf{a}_i\\ -\mathbf{s}_{j,k}\end{bmatrix}^T\mathbf{Z}\begin{bmatrix}\mathbf{a}_i\\ -\mathbf{s}_{j,k}\end{bmatrix} \quad i\in\mathcal{A}_j \\ & d_{i,j,k}^2=\begin{bmatrix}\mathbf{0}_2\\ \mathbf{s}_{i,k}-\mathbf{s}_{j,k}\end{bmatrix}^T\mathbf{Z}\begin{bmatrix}\mathbf{0}_2\\ \mathbf{s}_{i,k}-\mathbf{s}_{j,k}\end{bmatrix} \quad i\in\mathcal{B}_j \\ & \mathbf{Z}\succeq\mathbf{0}_{2N+2} \end{aligned} \tag{40}$$

The cost function of the convex problem is linear. So it ensures that there is only one minimum point. The convex optimization problem requires no initialization from the user. It can be solved with well known algorithms such as interior point methods which are self initialized and avoid the solution to the initialization. Extracting from $\mathbf{Z}$ we can obtain the motion parameters of the multiple mobile targets.

VI. COMPLEXITY ANALYSIS

When considering the full measurements between the mobile targets and the anchor nodes, each mobile target has a total of M connected anchor nodes. In the noncooperative ULLS of (10), $\mathbf{A}_j\in\mathbb{R}^{KM\times 7}$, $\Sigma_j\in\mathbb{R}^{KM\times KM}$, $\mathbf{b}_j\in\mathbb{R}^{KM}$. The solution of the noncooperative ULLS estimator

TABLE 2. Parameters in computing the computational complexity.

Algorithms	Variables	m	N_{socp}	n_i^{socp}	N_{sdp}	n_i^{sdp}
noncooperative SDP	$4KM+16$	$2KM+3$	0	0	$KM+1$	KM of size 2, 4
noncooperative SOCSDP	$KM+17$	$KM+3$	1	$KM+1$	1	4
cooperative SDP	$4KL+(2N+2)^2$	$2KL+3$	0	0	$KL+1$	KL of size 2, $2N+2$
cooperative SOCSDP	$KL+(2N+2)^2+1$	$KL+3$	1	$KL+1$	1	$2N+2$

includes five matrix multiplications and one matrix inversion. Referred by [53] the computational complexity of (10) is upper bounded by $O(14K^2M^2)$ when assuming $MK \gg 7$. Since a total of N mobile targets is required to be estimated in the network, the computational complexity of noncooperative ULLS estimator is $O(14NK^2M^2)$. In the noncooperative CLLS depicted as (16), $\mathbf{G}_j \in \mathbb{R}^{6\times4}$, $\Sigma_{\gamma,j} \in \mathbb{R}^{4\times4}$, $\mathbf{h}_j \in \mathbb{R}^6$. So the computational complexity of N mobile targets is upper bounded by $O(488N)$.

To illustrate the proposed convex algorithm, the number of the SOCP and SDP constraints are denoted as N_{socp} and N_{sdp}, respectively. The number of equality constraints is denoted as m in the designed convex optimization model. The corresponding dimension of the ith SOCP and SDP cone are denoted as n_i^{socp} and n_i^{sdp}, respectively. Tab. 2 lists the number of variables, equality constraints, SOCP constraints, SDP constraints and dimension of SOCP and SDP cone for these proposed algorithms. The computational complexity of these proposed algorithms is calculated as a function of K, the number of samples, M, the number of anchor nodes, N, the number of mobile targets, and L, the total number of connections. For a network with full connectivity, we can obtain that $L = N\big(M + (N-1)/2\big)$.

As can be seen from Tab. 2, the noncooperative SDP algorithm has $4KM + 16$ variables, while the proposed noncooperative SOCSDP has only $KM + 17$ variables. Moreover, the number of equality constraints for noncooperative SOCSDP is $KM + 3$, which is always smaller than $2KM + 3$, the number of the noncooperative SDP. Two different cooperative approaches including SDP and SOCSDP are also proposed to estimate the unknown parameters of the mobile targets. The number of variables for the cooperative SDP is $4KL + (2N + 2)^2$, which is greatly larger than $KL + (2N + 2)^2 + 1$, the number of the cooperative SOCSDP. Similarly, the number of the cooperative SOCSDP is always smaller than that of the cooperative SDP algorithm.

A convex optimization problem can be solved by iterative optimization techniques, e.g., interior-point methods. As is known, the worst-case complexity of solving the mixed SOCP and SDP algorithm is $O\big((m^2\sum_{i=1}^{N_{socp}} n_i^{socp} + m^2\sum_{i=1}^{N_{sdp}} n_i^{sdp^2} + m\sum_{i=1}^{N_{sdp}} n_i^{sdp^3} + m^3)\sqrt{N}\log(1/\epsilon)\big)$, where $\sqrt{N}\log(1/\epsilon)$ is the required least iterations, ϵ is the accuracy of the convex optimization solution. The complexity of solving the mixed SOCSDP is linear with n_i^{socp}, while that of solving the SDP is quadratic in n_i^{sdp}. Using the parameters listed in Tab. 2 and assuming $KM \gg 17$, we further calculate the computational complexity for N mobile targets.

$$\begin{cases} \text{Noncooperative SDP} \simeq O\big(16NK^3M^3\sqrt{N}\log(1/\epsilon)\big) \\ \text{Noncooperative SOCSDP} \simeq O\big(2NK^3M^3\sqrt{N}\log(1/\epsilon)\big) \\ \text{Cooperative SDP} \simeq O\big(24K^3L^3\sqrt{N}\log(1/\epsilon)\big) \\ \text{Cooperative SOCSDP} \simeq O\big(2K^3L^3\sqrt{N}\log(1/\epsilon)\big) \end{cases} \tag{41}$$

As can be seen that the computational complexity of the SOCSDP is always less than that of the SDP, whether it is a cooperative approach or not.

VII. EVALUATION

The ULLS, CLLS, SDP and SOCSDP algorithms for the noncooperative or cooperative approach are proposed to estimate the motion parameters of the multiple mobile targets, when the mobile targets start from the initial positions at the constant velocities. To evaluate the performance of these proposed algorithms, two different scenarios were examined in the simulations. In the first scenario, four anchor nodes were placed regularly on the corners of a square 100 m × 100 m. In the square region, five mobile targets started from the initial positions which were randomly determined. Fig. 2(a) shows the geographical coordinates of the deployed anchor nodes, the initial positions and moving velocities of mobile targets. However, four anchor nodes were placed irregularly in the second scenario. The geography of four anchor nodes is plotted in the Fig. 2(b), where the initial positions and moving velocities of the mobile targets are the same as in Fig. 2(a). The moving velocities of five mobile targets are denoted in the circle brackets. Full connectivity is initially assumed, meaning that each mobile target can be connected to all anchor nodes and also to all other mobile targets, unless otherwise noted. The propagation speed c is set to 3×10^8 m/s. Each standard deviation $\delta_{i,j,k}$ of the TOA measurement noise is equal to δ. The proposed convex SDP or SOCSDP algorithm is implemented by the CVX toolbox using SeDuMi as the solver.

A. IMPACTS OF THE NOISES

To demonstrate the performance of the proposed noncooperative ULLS, CLLS, SDP, SOCSDP, the cooperative SDP and SOCSDP algorithms, we perform Monte Carlo (MC) simulations with 500 ensemble runs to evaluate the root mean square error (RMSE) of the estimated parameters. The RMSEs of

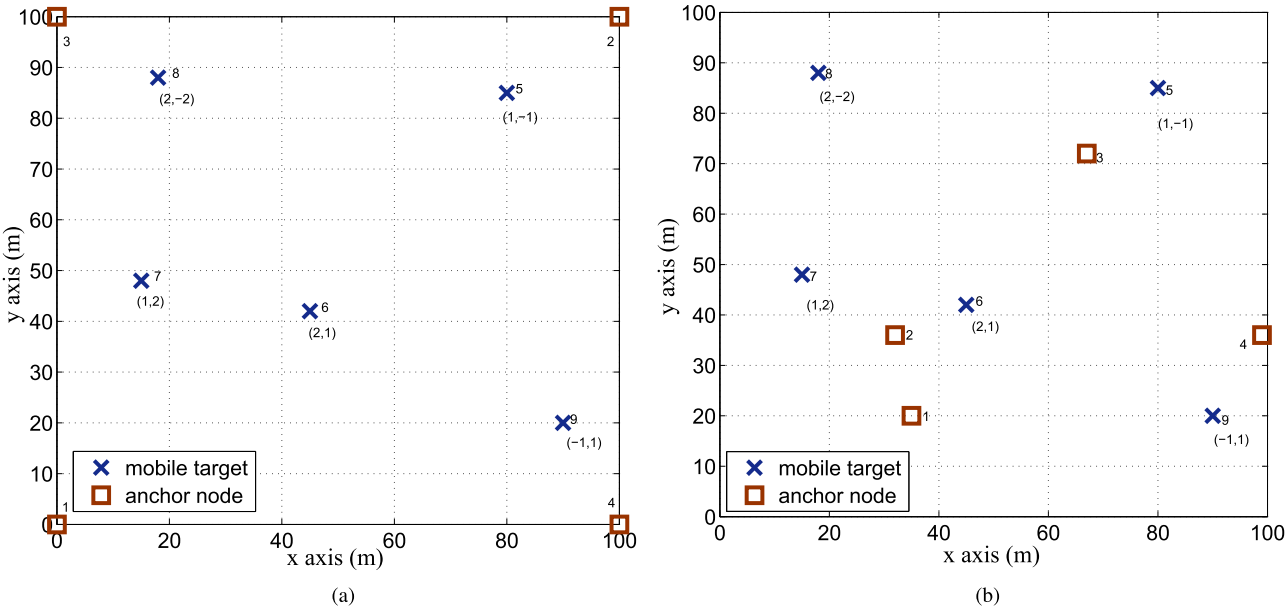

FIGURE 2. The geography and parameter setting of four anchor nodes and five mobile targets. (a) The first scenario. (b) The second scenario.

these proposed algorithms are calculated by averaging over the motion parameters of five mobile targets. The number of samples K and sampling period T are set to 5 and 1 s, respectively, when the noise standard deviation δ is varied from 1 ns to 10 ns.

When the anchor nodes are placed regularly as Fig. 2(a), the RMSE performance of these proposed algorithms is plotted in Fig. 3(a) where the CRLB of estimated parameters is derived in appendix. It is shown that the RMSE performance of each algorithm degrades as the noise standard deviation increases. When the noise standard deviation is small, these proposed algorithms perform well. The RMSE difference between the CRLB and each proposed algorithm becomes larger with the increasing of noise standard deviation. Compared to the noncooperative ULLS estimator, the RMSE of the CLLS is less due to the exploiting of the constraint conditions. Since the same optimization function is used, the RMSE of the SOCSDP is very close to that of the SDP. Furthermore, the cooperative approaches including the SOCSDP and the SDP perform significantly better than the noncooperative approaches since the TOA measurements between the mobile targets are additionally employed into the optimization model.

The velocities of all mobile targets are estimated along with the initial positions when the noise standard deviation is also varied from 1 ns to 10 ns. Fig. 3(b) shows the RMSE performance of these proposed algorithms when the anchor nodes are deployed in the first scenario. It can also be seen from Fig. 3(b) that the RMSE of each proposed algorithms becomes larger with the increasing of noise standard deviation. When the noise standard deviation is set to 1 ns, the RMSEs of noncooperative ULLS and cooperative SOCSDP are 0.11 m and 0.08 m, respectively. However, when the noise standard deviation is increased to 10 ns, the RMSEs of noncooperative ULLS and cooperative SOCSDP achieve 1.08 m and 0.75 m, respectively. Compared with the RMSE performance of initial positions plotted in Fig. 3(a), the difference between the RMSE of cooperative approach and its CRLB becomes larger in Fig. 3(b).

The deployment of the anchor nodes is irregular and shown as Fig. 2(b). When the noise standard deviation is varied from 1 ns to 10 ns, Fig. 3(c) and Fig. 3(d) plot the RMSEs of estimated initial positions and velocities, respectively. As can be seen that the cooperative SOCSDP and SDP algorithms provide better accuracy performance than the noncooperative approaches. In comparison with the previous scenario, the RMSE difference between the ULLS and the CLLS is much larger. So the ULLS performs worse when the anchor nodes are placed as irregularly. It is also shown that the RMSE of SOCSDP is very close to that of the SDP when the standard deviation is increased from 1 ns to 10 ns.

B. NUMBER OF SAMPLES

More number of samples means the increasing of the measurement information, so the CRLB of estimated unknown parameters should be reduced. The noise standard deviation and the sampling period are set to 1 ns and 1 s, respectively. By using the regular placement of the anchor nodes, the RMSEs of ULLS, CLLS, noncooperative SOCSDP and cooperative SOCSDP are plotted in Fig. 4(a) when the number of samples is increased from 3 to 10. The RMSE of the SDP is very close to that of the SOCSDP for the same network configuration, so the RMSE of the noncooperative or the cooperative SDP is not shown here. As can be seen from Fig. 4(a) that the RMSE of estimated initial position is reduced with the increasing of the number of samples.

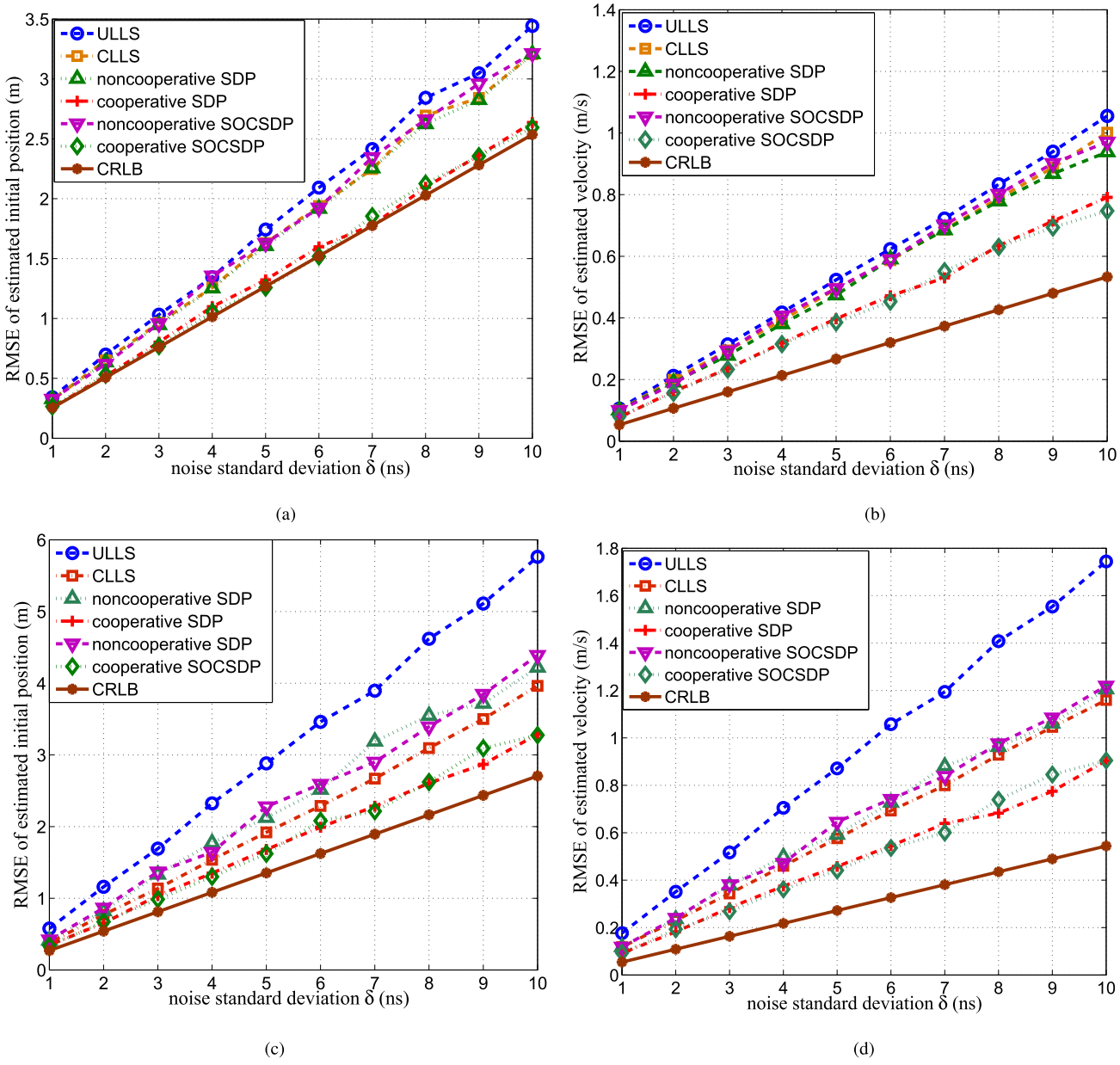

FIGURE 3. Impacts of the noises in the different scenarios, $K=5$, $T=1$ s. (a) RMSE of estimated initial position in the first scenario. (b) RMSE of estimated velocity in the first scenario. (c) RMSE of estimated initial position in the second scenario. (d) RMSE of estimated velocity in the second scenario.

The cooperative SOCSDP performs best among four algorithms since the measurements between the mobile targets are employed into the optimization model in the cooperative approach. When the number of samples is set to 3, the RMSE of the cooperative SOCSDP is 0.38 m. However, the RMSE of the cooperative SOCSDP is reduced to 0.17 m, when the number of samples is set to 10.

Fig. 4(b) also plots the RMSE of estimated velocity with four proposed algorithms, when the number of samples is increased from 3 to 10. As also can be seen that the RMSE of estimated velocity is less at a larger number of samples. The RMSE of estimated velocity is reduced from 0.18 m/s to 0.03 m/s with the cooperative SOCSDP algorithm, when the number of samples is increased from 3 to 10. However, a larger number of samples leads to increasing of the computational complexity of the proposed algorithms. The RMSE of the cooperative SOCSDP is also least among four algorithms. Compared with the estimated initial position, the RMSE of the estimated velocity is farther from its corresponding CRLB with the cooperative SOCSDP algorithm.

C. COMPARISON WITH THE STATIC METHOD

In [35], a static SOCSDP method is proposed when mobile targets are not assumed. Due to the extra unknown parameter

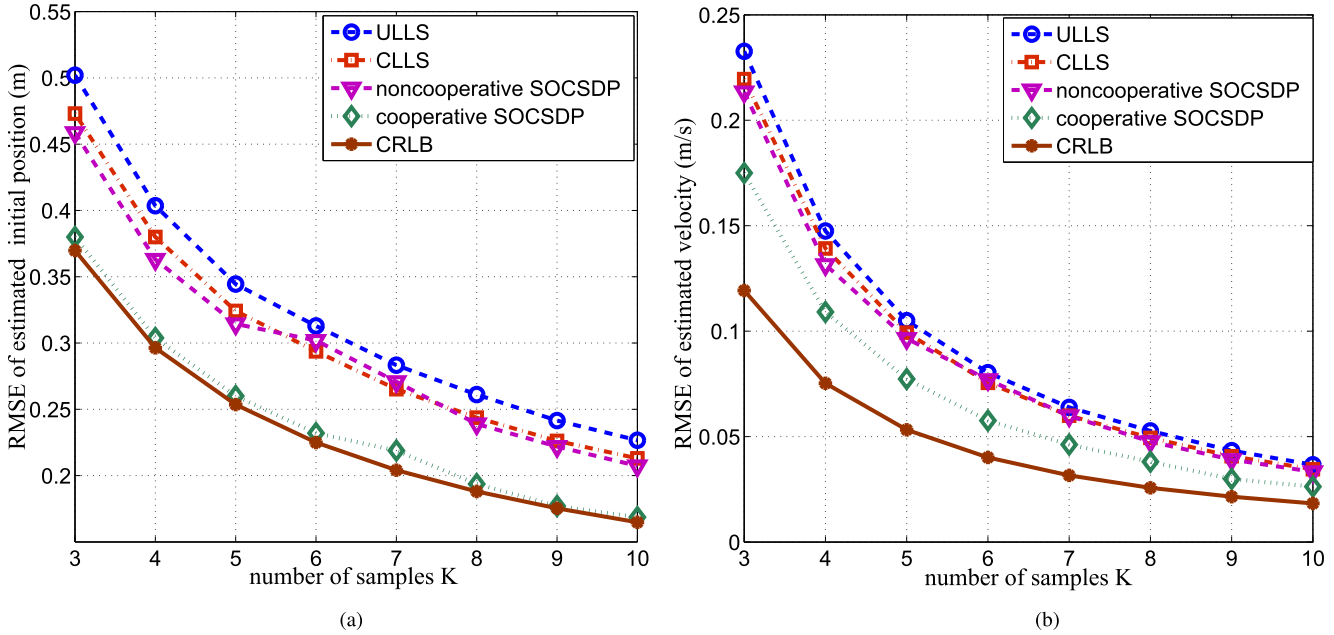

FIGURE 4. Impacts of number of samples, δ=1 ns, T=1 s. (a) RMSE of estimated initial positions in the first scenario. (b) RMSE of estimated velocities in the first scenario.

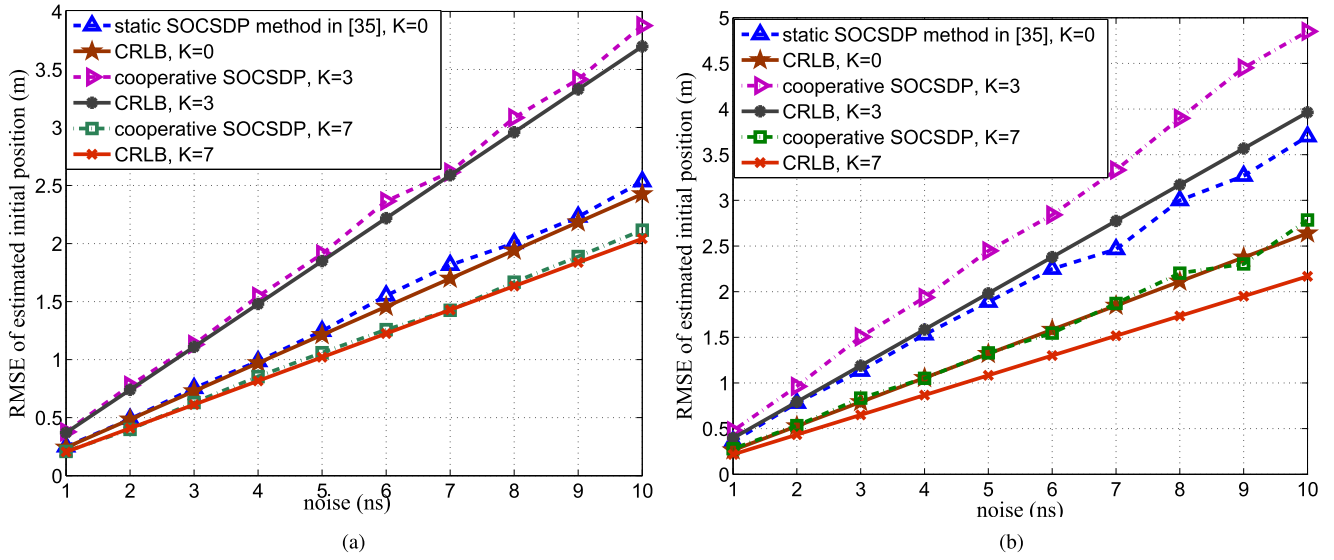

FIGURE 5. RMSE of estimated initial position under different number of samples, T=1 s. (a) The first scenario. (b) The second scenario.

of the velocity, the RMSE performance of our proposed method is worse than the static SOCSDP method proposed in [35], when the number of samples is less. However, the larger number of samples leads to the increasing of measurement information and would improve the RMSE performance of our proposed algorithms. When the sampling period is set to 1 s and the noise standard deviation is varied from 1 ns to 10 ns, the RMSEs of different methods are plotted by considering different number of samples in Fig. 5.

As can be seen that the RMSE of estimated initial position is larger with a smaller number of samples. When the number of samples and the noise standard deviation are respectively set to 3 and 10 ns, the RMSE of estimated initial position is 3.88 m with the cooperative SOCSDP. The RMSE of the static SOCSDP method is 2.54 m. However, When the number of samples is increased to 10, the RMSE of estimated initial position is reduced to 2.08 m with the cooperative SOCSDP. So it is shown that our cooperative SOCSDP performs better than the static SOCSDP method at a larger number of samples.

When the anchor nodes are placed irregularly, the RMSE of estimated initial position is also plotted in Fig. 5(b) with these

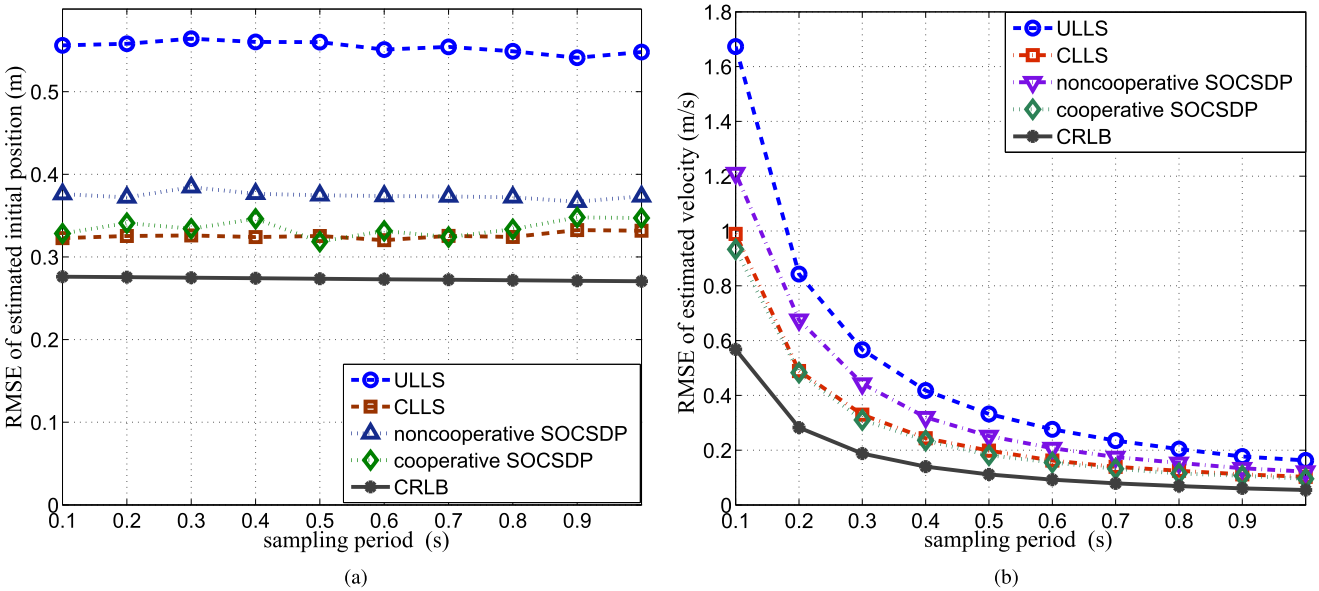

FIGURE 6. RMSE performance with different sampling period, K=5, δ=1 ns. (a) RMSE of estimated initial position in the second scenario. (b) RMSE of estimated velocity in the second scenario.

different methods. It is also shown the proposed algorithms perform worse as the noise standard deviation increases. When the anchor nodes are placed regularly in the first scenario, the RMSE difference between each algorithm and its corresponding CRLB is small. However, in the irregular deployment of the anchor nodes, the RMSE difference between each algorithm and its CRLB is increased. When the number of samples and the noise standard deviation are respectively set to 7 and 10 ns, the RMSE of the cooperative SOCSDP and its CRLB are 2.12 m and 2.04 m in the first scenario. In the second scenario, the RMSE of the cooperative SOCSDP and its CRLB are 2.78 m and 2.16 m, when the number of samples and the noise standard deviation are also set to 7 and 10 ns, respectively. The RMSE difference between the cooperative SOCSDP algorithm and its CRLB is increased from 0.08 m to 0.62 m, when the anchor nodes are placed as the second scenario.

D. SAMPLING PERIOD

Smaller sampling period can shorten the moving distance of the mobile targets when keeping the moving velocities invariable. So it ensure that the mobile targets are in the communication range of the anchor nodes. Then the motion parameters would be possible to be instantaneously captured. In this subsection, we examine the impacts of sampling period on the RMSE performance of estimated motion parameters. The number of samples and the noise standard deviation are set to 5 and 1 ns, respectively. When the sampling period is increased from 0.1 s to 1 s, the RMSE is plotted in Fig. 6 for the irregular deployment of the anchor nodes.

Firstly, the RMSE of estimated initial positions is evaluated and plotted in Fig. 6(a). As can be seen that the RMSE of the estimated initial position is essentially unchanged as the sampling period increases. When the sampling period is increased from 0.1 s to 1 s, the RMSE of the cooperative SOCSDP is fluctuated from 0.32 m to 0.35 m. The fluctuated range of each algorithm is very small with the increasing of sampling period.

Similarly, the RMSE of estimated velocity is plotted in Fig. 6(a), when the sampling period is increased from 0.1 s to 1 s. The shown result in Fig. 6(b) is different from Fig. 6(a). The RMSE of estimated velocity is reduced as the sampling period increases with each algorithm. When the sampling period is varied from 0.1 s to 1 s, the RMSE of the cooperative SOCSDP is reduced from 0.93 m/s to 0.09 m/s. Each algorithm performs better at a larger sampling period, but it prolongs the moving distances of the mobile targets and would lead that the mobile targets is out of the communication range of the anchor nodes.

E. REAL EXPERIMENT

To compare the performance of these different algorithms, we also conduct a real experiment by using ultra wide band (UWB) module of DWM1000. The UWB module DWM1000 directly provides the ranging between sensor nodes by using the TOA measurement. So we should modify our model by setting the propagation speed c to 1. The initial positions of the mobile targets and anchor nodes are manually set and measured. Motion sensor module are equipped to the sensor nodes for obtaining the ground truth of the velocities. In the real experiment, four anchor nodes are placed on the corners of a square 12 m $\times$ 12 m. Three mobile targets started from the initial positions (3, 3), (6, 7.5), and (1.5, 8) with the velocities of (0.1, $-$0.2), ($-$0.3, 0.1), and (0.1, $-$0.3) in an indoor environment shown as Fig. 7(a). When considering the velocities of three mobile targets to be unknown, we perform

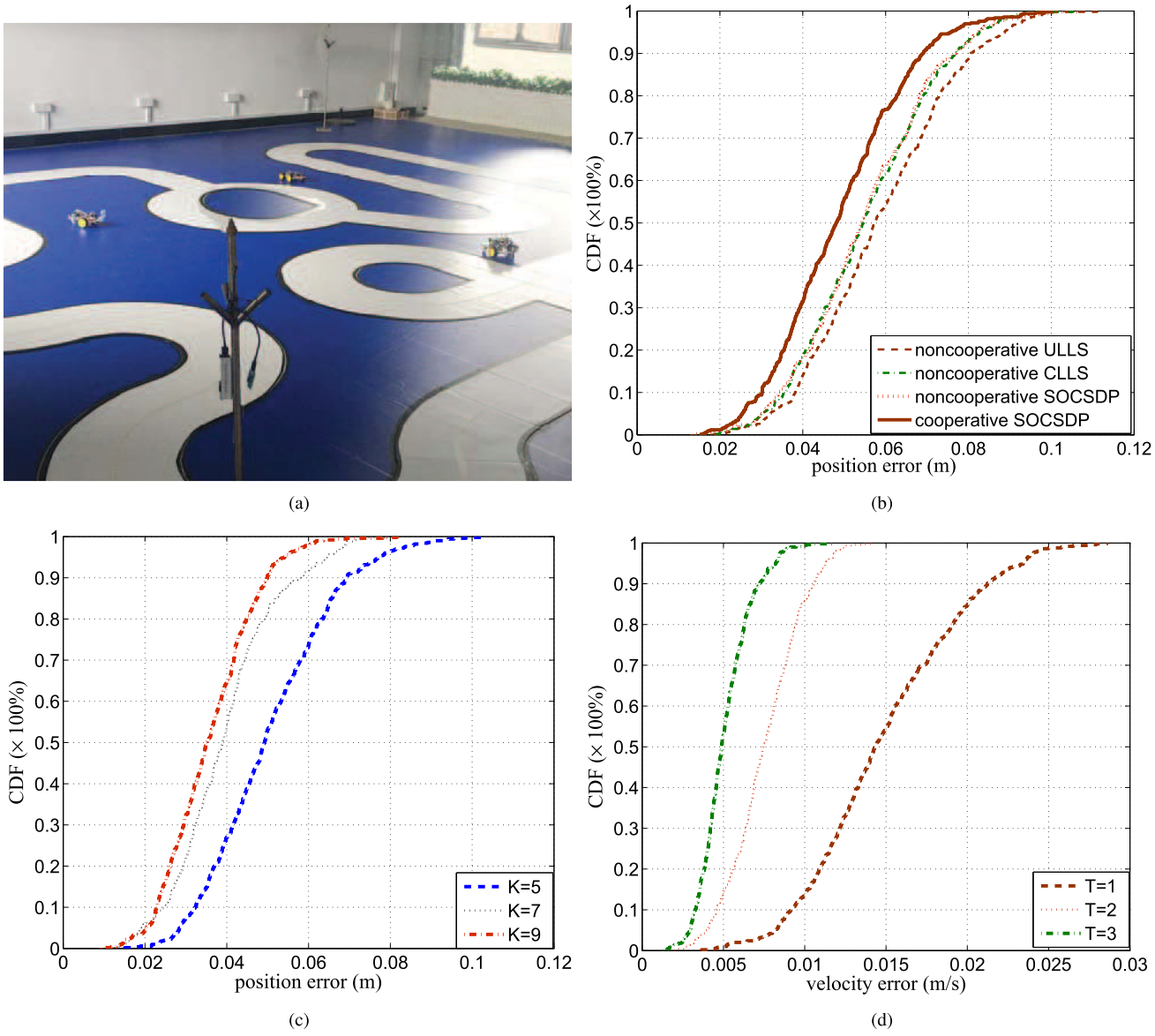

FIGURE 7. The setting and performance in real experiment. (a) The scene of real experiment. (b) position error CDF with different algorithms, T=1 s, K=5. (c) position error CDF with the cooperative SOCSDP under different number of samples, T=1 s. (d) velocity error CDF with the cooperative SOCSDP under different sampling period, K=5.

500 samples and estimate the motion parameters of three mobile targets. The cumulative distribution function (CDF) of positioning error is plotted in Fig. 7(b) with the noncooperative ULLS, CLLS, SOCSDP, and cooperative SOCSDP, when the sampling period and number of samples are set to 1 s and 5, respectively. As can be seen that 10% of the positioning error is larger than 0.081 m by using the noncooperative ULLS. However, only about 6% of the positioning error is larger than 0.081 m with the noncooperative CLLS. Compared with the noncooperative ULLS, the positioning error of noncooperative CLLS is less due to the exploiting of the constrained condition. It is also interesting to see that the cooperative SOCSDP provides the best accuracy performance among the four algorithms. The average positioning errors are 0.048 m, 0.054 m, 0.054 m, and 0.057 m with the cooperative SOCSDP, the noncooperative SOCSDP, the CLLS and the ULLS, respectively.

We also investigate the performance of the cooperative SOCSDP with different number of samples in the real experiment since the accuracy performance of the cooperative SOCSDP is best among four different algorithms. When the sampling period is set to 1 s, the position error CDF of the cooperative SOCSDP is plotted in Fig. 7(c) with different number of samples. When the number of samples is set to 5, 7, and 9, 10% of the position error is larger than 0.079 m, 0.057 m, and 0.048 m, respectively. So the position error is reduced as the number of samples increases.

The velocities of the mobile targets are also estimated along with the initial positions. The performance of the velocity error is also investigated, when the number of samples is invariably set to 5. The velocity error CDF is plotted in Fig. 7(d) with the cooperative SOCSDP, when the sampling period is set to 1 s, 2 s, and 3 s, respectively. As can be seen that 10% of the velocity error is more than 0.022 m/s when the sampling period is set to 1 s. However, the largest velocity error is only 0.012 m/s when the sample period is increased to 3 s. The estimated velocity error is also greatly reduced as the sampling period increases.

VIII. CONCLUSIONS

By using time of arrival (TOA) measurements between mobile targets and anchor nodes, the noncooperative approaches including the unconstrained linear least square (ULLS), constrained linear least square (CLLS), semidefinite programming (SDP) and mixed second order cone semidefinite programming (SOCSDP) are proposed to estimate the initial positions and the velocities of the mobile targets. Then the noncooperative convex approaches are extended to the cooperative SDP and SOCSDP by employing the TOA measurements between the mobile targets each other. The linear estimators including the ULLS and CLLS provide a closed-form fast solution to the unknown parameters of the mobile targets. Compared with the ULLS, the CLLS provides better accuracy performance due to the exploiting of the constraint condition. The RMSE of the cooperative approach is less than that of the noncooperative approach, since the TOA measurements between the mobile targets each other are additionally exploited as well as the TOA measurements between the mobile targets and the anchor nodes. The convex SOCSDP algorithm provides the approximately identical accuracy performance with the SDP, but the computational complexity of the SOCSDP is less than that of the SDP.

APPENDIX

CRLB FOR MOTION PARAMETER CAPTURING

When there are gaussian measurement noises in the proposed model, CRLB defines the a lower bound on the variance which provides a benchmark for evaluating the accuracy performance of estimation algorithm. The CRLB of unknown parameters is the diagonal elements of the inverse of the Fisher information matrix (FIM). When the TOA measurements are designed to estimate the motion parameters of multiple mobile targets, we derive the CRLB for cooperative approaches which include noncooperative case as a special case ($N = 1$).

In the proposed model, both the initial positions and the velocities of the mobile targets are required to be estimated and considered as unknown parameters. A new vector is defined and denoted as $\varphi = [\mathbf{x}_0^T \quad \mathbf{v}^T]^T \in \mathbb{R}^{4N}$, $\mathbf{x}_0 = [\mathbf{x}_{M+1,0}^T \quad \mathbf{x}_{M+2,0}^T \quad \cdots \quad \mathbf{x}_{M+N,0}^T]^T$, $\mathbf{v} = [\mathbf{v}_{M+1}^T \quad \mathbf{v}_{M+2}^T \quad \cdots \quad \mathbf{v}_{M+N}^T]^T$.

The probability density function of all TOA measurements is given by

$$p(\mathbf{t};\varphi) = \prod_{i\in\mathcal{E}_|}\prod_{j=M+1}^{M+N}\prod_{k=1}^{K}\{\frac{1}{\sqrt{2\pi}\delta_{i,j,k}}\exp\{-\frac{(\frac{d_{i,j,k}}{c}-t_{i,j,k})^2}{2\delta_{i,j,k}^2}\}\} \tag{42}$$

where $d_{i,j,k}$ is obtained with

$$\begin{cases} d_{i,j,k} = \parallel \mathbf{a}_i - \mathbf{x}_{j,0} - kT\mathbf{v}_j \parallel & i \in \mathcal{A}_j \\ d_{i,j,k} = \parallel \mathbf{x}_{i,0} - \mathbf{x}_{j,0} + kT(\mathbf{v}_i - \mathbf{v}_j) \parallel & i \in \mathcal{B}_j \end{cases} \tag{43}$$

So the FIM of the parameter capturing problem is obtained with

$$\mathbf{F} = -\frac{\partial^2 \ln p(\mathbf{t};\varphi)}{\partial\varphi^T\partial\varphi} \tag{44}$$

So (44) can also be rewritten as

$$\mathbf{F} = (\frac{\partial\mathbf{r}(\varphi)}{\partial\varphi})^T\Sigma\frac{\partial\mathbf{r}(\varphi)}{\partial\varphi} \tag{45}$$

where $\Sigma = \mathrm{diag}\{\delta_{i,j,k}^2\}$, $\mathbf{r}(\varphi) = [r_{i,j,k}]$, $i \in \mathcal{E}_j, j = M+1, M+2\ldots, M+N, k = 1, 2\ldots, K$. $r_{i,j,k} = \frac{d_{i,j,k}}{c} - t_{i,j,k}$, $\frac{\partial\mathbf{r}(\varphi)}{\partial\varphi} = [r_{i,j,k}^{\varphi}]$, $r_{i,j,k}^{\varphi} = [r_{i,j,k}^{\mathbf{x}_0} \quad r_{i,j,k}^{\mathbf{v}}]$, $r_{i,j,k}^{\mathbf{x}_0}$ and $r_{i,j,k}^{\mathbf{v}}$ are given by

$$\begin{cases} r_{i,j,k}^{\mathbf{x}_0} = [\mathbf{0}_{2(j-M-1)} \quad \frac{\mathbf{x}_{j,0}+kT\mathbf{v}_j-\mathbf{a}_i}{d_{i,j,k}} \quad \mathbf{0}_{2(N+M-j)}], \\ \quad i \in \mathcal{A}_j \\ r_{i,j,k}^{\mathbf{v}} = [\mathbf{0}_{2(j-M-1)} \quad \frac{kT(\mathbf{x}_{j,0}+kT\mathbf{v}_j-\mathbf{a}_i)}{d_{i,j,k}} \quad \mathbf{0}_{2(N+M-j)}], \\ \quad i \in \mathcal{A}_j \\ r_{i,j,k}^{\mathbf{x}_0} = [\mathbf{0}_{2(i-M-1)} \quad \frac{\mathbf{x}_{i,0}-\mathbf{x}_{j,0}+kT(\mathbf{v}_i-\mathbf{v}_j)}{d_{i,j,k}} \quad \mathbf{0}_{2(j-i-1)} \\ \quad \frac{\mathbf{x}_{j,0}-\mathbf{x}_{i,0}+kT(\mathbf{v}_j-\mathbf{v}_i)}{d_{i,j,k}} \quad \mathbf{0}_{2(N+M-j)}](i<j), \quad i \in \mathcal{B}_j \\ r_{i,j,k}^{\mathbf{v}} = [\mathbf{0}_{2(i-M-1)} \quad \frac{kT[\mathbf{x}_{i,0}-\mathbf{x}_{j,0}+kT(\mathbf{v}_i-\mathbf{v}_j)]}{d_{i,j,k}} \quad \mathbf{0}_{2(j-i-1)} \\ \quad \frac{kT[\mathbf{x}_{j,0}-\mathbf{x}_{i,0}+kT(\mathbf{v}_j-\mathbf{v}_i)]}{d_{i,j,k}} \quad \mathbf{0}_{2(N+M-j)}](i<j), \\ \quad i \in \mathcal{B}_j \end{cases} \tag{46}$$

So the CRLB of unknown motion parameters is obtained with

$$\mathrm{CRLB}([\varphi]_l) = [\mathbf{F}^{-1}(\varphi)]_{l,l} \tag{47}$$

where $l = 1, 2, \ldots, 4N$.

REFERENCES

[1] S. Shu and J. M. Conrad, "A survey of robotic applications in wireless sensor networks," in *Proc. IEEE Southeastcon*, Jacksonville, FL, USA, Apr. 2013, pp. 1–5.
[2] J. Suh, S. You, S. Choi, and S. Oh, "Vision-based coordinated localization for mobile sensor networks," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 2, pp. 611–620, Apr. 2016.
[3] M. Schwager, M. P. Vitus, S. Powers, D. Rus, and C. J. Tomlin, "Robust adaptive coverage control for robotic sensor networks," *IEEE Trans. Control Netw. Syst.*, vol. 4, no. 3, pp. 462–476, Sep. 2017.

[4] I. Guvenc and C.-C. Chong, "A survey on TOA based wireless localization and NLOS mitigation techniques," *IEEE Commun. Surveys Tuts.*, vol. 11, no. 3, pp. 107–124, Aug. 2009.

[5] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Commun. Mag.*, vol. 40, no. 8, pp. 102–114, Aug. 2002.

[6] M. Amjad, M. Sharif, M. K. Afzal, and S. W. Kim, "TinyOS-new trends, comparative views, and supported sensing applications: A review," *IEEE Sensors J.*, vol. 16, no. 9, pp. 2865–2889, May 2016.

[7] Y. Wang, R. Tan, G. Xing, J. Wang, and X. Tan, "Profiling aquatic diffusion process using robotic sensor networks," *IEEE Trans. Mobile Comput.*, vol. 13, no. 4, pp. 880–893, Apr. 2014.

[8] G. A. Hollinger *et al.*, "Underwater data collection using robotic sensor networks," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 5, pp. 899–911, Jun. 2012.

[9] R. Graham and J. Cortes, "Adaptive information collection by robotic sensor networks for spatial estimation," *IEEE Trans. Autom. Control*, vol. 57, no. 6, pp. 1404–1419, Jun. 2012.

[10] D. Wu, D. M. Chatzigeorgiou, K. Youcef-Toumi, and R. Ben-Mansour, "Node localization in robotic sensor networks for pipeline inspection," *IEEE Trans. Ind. Inform.*, vol. 12, no. 2, pp. 809–819, Apr. 2016.

[11] G. A. Hollinger, C. Choudhuri, U. Mitra, and G. S. Sukhatme, "Squared error distortion metrics for motion planning in robotic sensor networks," in *Proc. IEEE Globecom Workshop*, Atlanta, GA, USA, Dec. 2013, pp. 1426–1431.

[12] L. He, P. Cheng, Y. Gu, J. Pan, T. Zhu, and C. Liu, "Mobile-to-mobile energy replenishment in mission-critical robotic sensor networks," in *Proc. IEEE Conf. Comput. Commun.*, Toronto, ON, Canada, May 2014, pp. 1195–1203.

[13] D. Herrero and H. Martínez, "Range-only fuzzy Voronoi-enhanced localization of mobile robots in wireless sensor networks," *Robotica*, vol. 30, no. 7, pp. 1063–1077, 2012.

[14] X. Li, G. Fletcher, A. Nayak, and I. Stojmenovic, "Randomized carrier-based sensor relocation in wireless sensor and robot networks," *Ad Hoc Netw.*, vol. 11, no. 7, pp. 1951–1962, 2013.

[15] L. V. Nguyen, S. Kodagoda, R. Ranasinghe, and G. Dissanayake, "Information-driven adaptive sampling strategy for mobile robotic wireless sensor network," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 1, pp. 372–379, Jan. 2016.

[16] J. Zhao *et al.*, "Localization of wireless sensor networks in the wild: Pursuit of ranging quality," *IEEE/ACM Trans. Netw.*, vol. 21, no. 1, pp. 311–323, Feb. 2013.

[17] J. Shen, A. F. Molisch, and J. Salmi, "Accurate passive location estimation using TOA measurements," *IEEE Trans. Wireless Commun.*, vol. 11, no. 6, pp. 2182–2192, Jun. 2012.

[18] M. D. Gillette and H. F. Silverman, "A linear closed-form algorithm for source localization from time-differences of arrival," *IEEE Signal Process. Lett.*, vol. 15, no. 1, pp. 1–4, Jan. 2008.

[19] 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.

[20] H.-J. Shao, X.-P. Zhang, and Z. Wang, "Efficient closed-form algorithms for AOA based self-localization of sensor nodes using auxiliary variables," *IEEE Trans. Signal Process.*, vol. 62, no. 10, pp. 2580–2594, May 2014.

[21] L. Lin, H. C. So, and Y. T. Chan, "Received signal strength based positioning for multiple nodes in wireless sensor networks," *Digit. Signal Process.*, vol. 25, pp. 41–50, Feb. 2014.

[22] S. Tomic, M. Beko, and R. Dinis, "Distributed RSS-based localization in wireless sensor networks based on second-order cone programming," *Sensors*, vol. 14, no. 10, pp. 18410–18432, Oct. 2014.

[23] W. Meng, W. Xiao, and L. Xie, "An efficient EM algorithm for energy-based multisource localization in wireless sensor networks," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 3, pp. 1017–1027, Mar. 2011.

[24] X. Wu, G. Wang, D. Dai, and M. Tong, "Accurate acoustic energy-based localization with beacon position uncertainty in wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 43, pp. 76–83, Aug. 2014.

[25] X. Sheng and Y.-H. Hu, "Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 53, no. 1, pp. 44–53, Jan. 2005.

[26] A. Simonetto and G. Leus, "Distributed maximum likelihood sensor network localization," *IEEE Trans. Signal Process.*, vol. 62, no. 6, pp. 1424–1437, Mar. 2014.

[27] K. W. Cheung, H. C. So, W.-K. Ma, and Y. T. Chan, "A constrained least squares approach to mobile positioning: Algorithms and optimality," *EURASIP J. Appl. Signal Process.*, vol. 2006, no. 1, pp. 1–23, Jan. 2006.

[28] M. Sun, L. Yang, and K. Ho, "Accurate sequential self-localization of sensor nodes in closed-form," *Signal Process.*, vol. 92, pp. 2940–2951, Dec. 2012.

[29] X. Qu and L. Xie, "An efficient convex constrained weighted least squares source localization algorithm based on TDOA measurements," *Signal Process.*, vol. 119, pp. 142–152, Feb. 2016.

[30] R. W. Ouyang, A. K.-S. Wong, and C.-T. Lea, "Received signal strength-based wireless localization via semidefinite programming: Noncooperative and cooperative schemes," *IEEE Trans. Veh. Technol.*, vol. 59, no. 3, pp. 1307–1318, Mar. 2010.

[31] B. Zhou and Q. Chen, "On the particle-assisted stochastic search mechanism in wireless cooperative localization," *IEEE Trans. Wireless Commun.*, vol. 15, no. 7, pp. 4765–4777, Jul. 2016.

[32] P. Oguz-Ekim, J. P. Gomes, J. Xavier, M. Stosic, and P. Oliveira, "An angular approach for range-based approximate maximum likelihood source localization through convex relaxation," *IEEE Trans. Wireless Commun.*, vol. 13, no. 7, pp. 3951–3964, Jul. 2014.

[33] P. M. Ghari, R. Shahbazian, and S. A. Ghorashi, "Wireless sensor network localization in harsh environments using SDP relaxation," *IEEE Commun. Lett.*, vol. 20, no. 1, pp. 137–140, Jan. 2016.

[34] R. M. Vaghefi, M. R. Gholami, R. M. Buehrer, and E. G. Ström, "Cooperative received signal strength-based sensor localization with unknown transmit powers," *IEEE Trans. Signal Process.*, vol. 61, no. 6, pp. 1389–1403, Mar. 2013.

[35] J. Zheng and X. Wu, "Convex optimization algorithms for cooperative RSS-based sensor localization," *Pervas. Mobile Comput.*, vol. 37, pp. 78–93, Jun. 2017.

[36] K. W. K. Lui, W.-K. Ma, H. C. So, and F. K. W. Chan, "Semi-definite programming algorithms for sensor network node localization with uncertainties in anchor positions and/or propagation speed," *IEEE Trans. Signal Process.*, vol. 57, no. 2, pp. 752–763, Feb. 2009.

[37] C. Soares, J. Xavier, and J. Gomes, "Simple and fast convex relaxation method for cooperative localization in sensor networks using range measurements," *IEEE Trans. Signal Process.*, vol. 63, no. 17, pp. 4532–4543, Sep. 2015.

[38] S. Salari, S. ShahbazPanahi, and K. Ozdemir, "Mobility-aided wireless sensor network localization via semidefinite programming," *IEEE Trans. Wireless Commun.*, vol. 12, no. 12, pp. 5966–5978, Dec. 2013.

[39] G. Wang, S. Cai, Y. Li, and M. Jin, "Second-order cone relaxation for TOA-based source localization with unknown start transmission time," *IEEE Trans. Veh. Technol.*, vol. 63, no. 6, pp. 2973–2977, Jul. 2014.

[40] G. Naddafzadeh-Shirazi, M. B. Shenouda, and L. Lampe, "Second order cone programming for sensor network localization with anchor position uncertainty," *IEEE Trans. Wirelss Commun.*, vol. 13, no. 2, pp. 749–763, Feb. 2014.

[41] B.-S. Cho, W.-S. Moon, W.-J. Seo, and K.-R. Baek, "A dead reckoning localization system for mobile robots using inertial sensors and wheel revolution encoding," *J. Mech. Sci. Technol.*, vol. 25, no. 11, pp. 2907–2917, 2011.

[42] T. Dumont and S. L. Corff, "Simultaneous localization and mapping in wireless sensor networks," *Signal Process.*, vol. 101, pp. 192–203, Aug. 2014.

[43] O. Demigha, W.-K. Hidouci, and T. Ahmed, "On energy efficiency in collaborative target tracking in wireless sensor network: A review," *IEEE Commun. Survey Tuts.*, vol. 15, no. 3, pp. 1210–1222, 3rd Quart., 2013

[44] É. L. Souza, E. F. Nakamura, and R. W. Pazzi, "Target tracking for sensor networks: A survey," *ACM Comput. Surv.*, vol. 49, no. 2, pp. 1–35, 2016.

[45] K. Zheng *et al.*, "Energy-efficient localization and tracking of mobile devices in wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2714–2726, Mar. 2017.

[46] A. F. Garcia-Fernandez, M. R. Morelande, and J. Grajal, "Multitarget simultaneous localization and mapping of a sensor network," *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4544–4558, Oct. 2011.

[47] C. Gentner, T. Jost, W. Wang, S. Zhang, A. Dammann, and U.-C. Fiebig, "Multipath assisted positioning with simultaneous localization and mapping," *IEEE Trans. Wireless Commun.*, vol. 15, no. 9, pp. 6104–6117, Sep. 2016.

[48] H. Naseri and V. Koivunen, "Cooperative simultaneous localization and mapping by exploiting multipath propagation," *IEEE Trans. Signal Process.*, vol. 65, no. 1, pp. 200–211, Jan. 2017.

[49] E. Xu, Z. Ding, and S. Dasgupta, "Target tracking and mobile sensor navigation in wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 1, pp. 177–186, Jan. 2013.
[50] Y. Yu, "Consensus-based distributed mixture Kalman filter for maneuvering target tracking in wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 10, pp. 8669–8681, Oct. 2016.
[51] Q. Gao, J. Wang, M. Jin, H. Chen, and H. Wang, "Target tracking by lightweight blind particle filter in wireless sensor networks," *Wireless Commun. Mobile Comput.*, vol. 14, no. 2, pp. 210–220, 2014.
[52] L. Dong, "Cooperative network localization via node velocity estimation," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Budapest, Hungary, Apr. 2009, pp. 2231–2236.
[53] S. Arora and B. Barak, *Computational Complexity: A Modern Approach*. Cambridge U.K.: Cambridge Univ. Press, 2009.

XIAOPING WU received the Ph.D. degree from Shanghai University in 2013. Since 2004, he has been with Zhejiang A & F University, where he is currently an Associate Professor with the School of Information Engineering. His research interests include mathematical modeling, optimization method, numerical computing, and its application in wireless sensor networks and ad hoc networks.

SHENGHUI WANG received the B.E. degree from Zhejiang A & F University in 2012, where he is currently pursuing the M.Eng. degree. His main interests include ubiquitous computing, communication technology, and the applications of wireless networks.

HAILING FENG received the Ph.D. degree in computer science from the University of Science and Technology of China in 2007. Since 2007, he has been with Zhejiang A & F University, where he is currently a Professor with the School of Information Engineering. His main interests include nondestructive testing, wireless networks, and computer vision.

JUNGUO HU received the Ph.D. degree from Beijing Forestry University in 2016. Since 2001, he has been with Zhejiang A & F University, where he is currently a Professor with the School of Information Engineering. His research interests include embedded systems, forest carbon sink monitoring and measurements, intelligent computing, and wireless sensor networks.

GUOYING WANG received the master's degree from Guangxi University in 2004. Since 2004, he has been with Zhejiang A & F University, where he is currently an Associate Professor with the School of Information Engineering. His research interests include topology optimization, adaptive sampling, and the applications of sensor networks and IoT.