COOPERATIVE LOCALIZATION USING EFFICIENT KALMAN FILTERING FOR MOBILE WIRELESS SENSOR NETWORKS

Hadi Jamali Rad, Toon van Waterschoot, and Geert Leus

Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology (TU Delft), Delft, The Netherlands
e-mail:{h.jamalirad, t.j.m.vanwaterschoot, g.j.t.leus}@tudelft.nl

ABSTRACT

We consider the problem of cooperative localization in mobile wireless sensor networks (WSNs). To be able to continuously localize the mobile network, we propose to exploit the knowledge of the location of the anchor nodes to linearize the nonlinear distance measurements with respect to the location of the unknown nodes. Based on this linearized measurement model, we estimate the location of the unknown nodes using a Kalman filter (KF) instead of a suboptimal extended KF (EKF) and try to estimate the corresponding unknown measurement noise covariance matrix using an iterative process. The simulation results illustrate that the proposed algorithm (only with a few iterations) attains the posterior Cramer-Rao bound (PCRB) of mobile location estimation and clearly outperforms related anchorless and anchored mobile localization algorithms.

1. INTRODUCTION

Accurate and cost-efficient sensor node localization is a critical requirement of WSNs in a wide variety of applications. In many practical scenarios, the nodes cannot be equipped with global positioning systems to locate themselves, and thus to remove this constraint, many research efforts have focused on proposing solutions to estimate the node locations using their pairwise distance measurements in a cooperative context. The aforementioned studies can be divided into two main categories, i.e., anchorless and anchored localization. In the former, there are no nodes with known locations (so-called anchors) and determining the relative location of the nodes is the ultimate goal. One popular solution to find the relative locations of the nodes based on distance measurements in a static network is to use multidimensional scaling (MDS) or its distributed version for large-scale networks [1]. On the other hand, by exploiting the knowledge of anchor locations a set of linear equations can be obtained. This is the basis of the so-called weighted MDS which avoids an eigenvalue decomposition (EVD) and attains the Cramer-Rao bound (CRB). This idea has been developed in [2] for multiple unknown nodes in a static network.

The problem of cooperative localization for mobile sensor networks is a challenging problem of special interest, and surprisingly it has not been efficiently solved yet. In [3], an anchorless localization algorithm for mobile (dynamic) networks has been proposed based on the theory of factor graphs. In this algorithm, each node requires knowledge about its own movement model as a probability distribution in order to do predictions, which is not so simple to be acquired in a real application and additionally it increases the computational complexity significantly. Two low-complexity anchorless algorithms have been proposed in [4] to locate and track the mobile nodes using a novel subspace tracking procedure, which attain an acceptable localization accuracy. In [5], an anchorless localization scheme based on the extended Kalman filtering (EKF) has been developed which incorporates the location of the nodes as well as their velocities in a state-space model.

However, the aforementioned algorithms (due to different inefficiencies) do not attain the achievable lower bound of estimation error (PCRB) for a mobile scenario when the problem is viewed as a discrete-time filtering problem [6]. Note that although solving the localization problem for a static network in every snapshot (e.g., using [2]) attains the CRB for that snapshot, it will not attain the PCRB. This is due to the fact that the information from the movement process is ignored. This motivated us to propose an algorithm to fill out this gap and attain the PCRB for a mobile WSN localization problem.

Inspired by the optimal filtering capability of the KF, we propose to linearize the nonlinear measurement model based on the knowledge of anchor locations by adopting similar matrix operations as in [2]. Then, instead of using the EKF, an appropriate KF can be used which can potentially attain the PCRB. The main advantages of the proposed algorithm can be described as follows. Unlike the MDS-based algorithms, the proposed algorithm does not involve EVD calculations which basically lead to high complexity as well as suboptimal performance. Furthermore, we show that the proposed algorithm attains the PCRB of the mobile scenario, which is a considerable improvement compared to the best existing algorithms. The remainder of this paper is organized as follows. In Section 2, we present the system model underlying our analysis and evaluations. Section 3 describes the proposed cooperative localization algorithm based on the KF. Section 4 explains the performance bounds under consideration in this work. Section 5 provides simulation results. Finally, concluding remarks are presented in Section 6.

2. SYSTEM MODEL

Consider a network of N wireless sensor nodes living in a 2-dimensional space (as the extension to the 3-dimensional case is straightforward) among which $l \geq 3$ are fixed anchors with known locations, and the remaining ones are mobile. Let $\{\mathbf{x}_{i,k}\}_{i=1}^{N}$ denote the actual vector coordinates of the sensor nodes, or equivalently, let $\mathbf{X}_k = [\mathbf{x}_{1,k}, \ldots, \mathbf{x}_{N,k}]$ be the matrix of coordinates at snapshot k. Then, we can write

$$\mathbf{X}_k = [\mathbf{X}_{a,k},\ \mathbf{X}_{u,k}],$$

This work was supported by NWO-STW under the VICI program (10382).

where $\mathbf{X}_{a,k} = [\mathbf{x}_{1,k}, \mathbf{x}_{2,k}, \ldots, \mathbf{x}_{l,k}]$ and $\mathbf{X}_{u,k} = [\mathbf{x}_{l+1,k}, \mathbf{x}_{l+2,k}, \ldots, \mathbf{x}_{N,k}]$ respectively represent the vector coordinates of the anchor nodes and unknown nodes in the k-th snapshot of the mobile network.

To be able to perform cooperative localization in a centralized manner, we have to collect pairwise distance measurements between all pairs of nodes. However, in practice these measurements are noisy and hence the distance measurements can be modeled as

$$r_{i,j,k} = d_{i,j,k} + q_{i,j,k}, \tag{1}$$

where $d_{i,j,k} = \|\mathbf{x}_{i,k} - \mathbf{x}_{j,k}\|$ is the noise-free Euclidean distance and $q_{i,j,k} \sim \mathcal{N}(0, \sigma^2_{i,j,k})$ is the independent and identically distributed (i.i.d.) noise. The distance measurements themselves can simply be calculated by means of time of arrival (TOA) measurements. Hence, we assume that the TOA information is already converted to noisy distance measurements.

A variety of movement models can be considered for the mobile nodes in the network. Here, we consider a low dynamic motion model for the mobile nodes so that the velocity can be modeled as a random walk process. This model is widely use in the literature [4, 5]. The corresponding discrete-time state equation can be given as

$$\mathbf{s}_{u,k} = \mathbf{\Phi}\mathbf{s}_{u,k-1} + \mathbf{w}_k, \tag{2}$$

where $\mathbf{s}_{u,k} = [\mathbf{x}^T_{u,k}, \dot{\mathbf{x}}^T_{u,k}]^T$ is the column vector of length $4(N-l)$ containing the locations and velocities of the unknown nodes at the k-th snapshot of the movement. We set $\mathbf{w}_k = [\mathbf{0}^T, \bar{\mathbf{w}}^T_k]^T$, where we assume that $\bar{\mathbf{w}}_k$ representing the speed variations is a vector with i.i.d. zero-mean Gaussian entries with standard deviations σ_w. Further, $\mathbf{\Phi} = \mathbf{I}_{4(N-l)} + \mathbf{F}T_s$, where T_s is the sampling period (time between two consecutive snapshots) and $\mathbf{F}$ is

$$\mathbf{F} = \begin{bmatrix} \mathbf{0}_{(2N-2l)\times(2N-2l)} & \mathbf{I}_{2N-2l} \\ \mathbf{0}_{(2N-2l)\times(2N-2l)} & \mathbf{0}_{(2N-2l)\times(2N-2l)} \end{bmatrix}.$$

Our goal is to estimate the locations of the unknown nodes.

3. KF OVER LINEARIZED MEASUREMENT MODEL

As described earlier, the distance measurements are nonlinear with respect to the location of the nodes. Thus, to be able to exploit the Kalman filter (KF) to estimate the locations, we have to take partial derivatives of the measurements $r_{i,j,k}$ and estimate the locations of the nodes by an extended Kalman filter (EKF) which leads to a sub-optimal performance [5]. To be able to use the standard KF, we take the idea from [2], and try to employ the known locations of the anchors to end up with a set of linear equations in the unknown locations. Then, taking advantage of this linear model we use a standard KF to localize the moving nodes in an adaptive manner.

Let us start by assuming that there is no noise. By collecting the squared pairwise distance measurements $d^2_{i,j,k}$ between the nodes in a distance matrix $\mathbf{D}_k$, i.e. $[\mathbf{D}_k]_{i,j} = d^2_{i,j,k}$, the double-centered distance matrix ($\mathbf{B}_k$) can be calculated as [1, 2, 4]

$$\mathbf{B}_k = -\frac{1}{2}\mathbf{J}\mathbf{D}_k\mathbf{J} = \mathbf{J}\mathbf{X}^T_k\mathbf{X}_k\mathbf{J}, \tag{3}$$

where $\mathbf{J}$ is the centering operator $\mathbf{J} = \mathbf{I}_N - \mathbf{1}_N\mathbf{1}^T_N/N$, with $\mathbf{I}_N$ the $N \times N$ identity matrix and $\mathbf{1}_N$ the $N \times 1$ vector of all ones. It is notable that, in case of a network with fixed nodes, $\mathbf{B}_k$ can be used in the classical MDS to recover the locations of the nodes $\mathbf{X}_k$ (up to a rigid transformation) by means of the EVD as described in [1]. However, we want to rewrite (3) as a linear function of the unknown locations, as done in [2], so that we can exploit the features of the standard KF for tracking. Some of the following derivations are similar to those in [2] except for the fact that we rewrite them for a mobile scenario on a per-snapshot basis (represented by the subscript k). To be able to establish the considered linearization, we assume that $\mathbf{X}_{a,k}\mathbf{1}_l = \mathbf{0}_{2\times1}$, which can always be satisfied by a simple rigid transformation. By partitioning $\mathbf{J}$ into upper and lower parts, we have

$$\mathbf{X}_k\mathbf{J} = \mathbf{X}_{a,k}\mathbf{J}_a + \mathbf{X}_{u,k}\mathbf{J}_u, \tag{4}$$

where $\mathbf{J} = [\mathbf{J}^T_a,\ \mathbf{J}^T_u]^T$. Now, by defining $\bar{\mathbf{X}}_{a,k} = [\mathbf{J}^T_a\mathbf{X}^T_{a,k},\ \mathbf{J}^T_u]^T$, we can calculate

$$\bar{\mathbf{X}}^\dagger_{a,k} = \bar{\mathbf{X}}^T_{a,k}(\bar{\mathbf{X}}_{a,k}\bar{\mathbf{X}}^T_{a,k})^{-1}. \tag{5}$$

Finally, by introducing $\bar{\mathbf{X}}_{u,k} = [\mathbf{X}^T_{u,k}, -\mathbf{I}_{N-l}]^T$, as described in [2], we will end up with the following linear set of equations for the noiseless pairwise distance measurements

$$\mathbf{B}_k\bar{\mathbf{X}}^\dagger_{a,k}\bar{\mathbf{X}}_{u,k} = \mathbf{0}_{N\times(N-l)}. \tag{6}$$

However, in the presence of measurement noise, instead of $\mathbf{D}_k$, we have to build the corresponding matrix $[\hat{\mathbf{D}}_k]_{i,j} = r^2_{i,j,k}$ and compute

$$\hat{\mathbf{B}}_k = -\frac{1}{2}\mathbf{J}\hat{\mathbf{D}}_k\mathbf{J}. \tag{7}$$

As a result, (6) will not hold anymore and we have to check what will be the influence of the noise. Let us therefore define

$$\mathbf{\Xi}_k = \hat{\mathbf{B}}_k\bar{\mathbf{X}}^\dagger_{a,k}\bar{\mathbf{X}}_{u,k} = \mathbf{H}_{L,k}\mathbf{X}_{u,k} - \mathbf{H}_{R,k}, \tag{8}$$

where

$$\mathbf{H}_{L,k} = \hat{\mathbf{B}}_k \begin{bmatrix} \bar{\mathbf{X}}^\dagger_{a,k} \\ \mathbf{0}_{(N-l)\times2} \end{bmatrix}, \tag{9}$$

$$\mathbf{H}_{R,k} = \hat{\mathbf{B}}_k \begin{bmatrix} -\frac{1}{l}\mathbf{1}_l\mathbf{1}^T_{N-l} \\ \mathbf{I}_{N-l} \end{bmatrix}. \tag{10}$$

Vectorizing both sides of (8) yields

$$\xi_k = \mathrm{vec}(\mathbf{\Xi}_k) = \bar{\mathbf{H}}_{L,k}\mathbf{x}_{u,k} - \mathbf{h}_{R,k}, \tag{11}$$

where

$$\bar{\mathbf{H}}_{L,k} = \mathbf{I}_{N-l} \otimes \mathbf{H}_{L,k},$$
$$\mathbf{x}_{u,k} = \mathrm{vec}(\mathbf{X}_{u,k}),$$
$$\mathbf{h}_{R,k} = \mathrm{vec}(\mathbf{H}_{R,k}),$$

with $\otimes$ representing the Kronecker product.

To match this to the state definition in (2), we define a modified version of (11) as the measurement model of our cooperative localization problem as

$$\mathbf{h}_{R,k} = \check{\mathbf{H}}_{L,k}\mathbf{s}_{u,k} - \xi_k, \tag{12}$$

Algorithm 1 Linearized Kalman Filter

1: Start with an initial location and velocity guess ($\hat{\mathbf{s}}_{u,0}$)
2: **for** $k = 1$ to K **do**
3: Next state:
$\hat{\mathbf{s}}_{u,k}^{-} = \mathbf{\Phi}\hat{\mathbf{s}}_{u,k-1}$
4: Next error covariance:
$\mathbf{P}_k^{-} = \mathbf{\Phi}\mathbf{P}_{k-1}\mathbf{\Phi}^T + \mathbf{Q}$
5: Compute the Kalman gain:
$\mathbf{K}_k = \mathbf{P}_k^{-}\check{\mathbf{H}}_{L,k}^T\big(\mathbf{R}_k^{\dagger} - \mathbf{R}_k^{\dagger}\check{\mathbf{H}}_{L,k}((\mathbf{P}_k^{-})^{-1} + \check{\mathbf{H}}_{L,k}^T\mathbf{R}_k^{\dagger}\check{\mathbf{H}}_{L,k})^{-1}\check{\mathbf{H}}_{L,k}^T\mathbf{R}_k^{\dagger}\big)$
6: Update the state:
$\hat{\mathbf{s}}_{u,k} = \hat{\mathbf{s}}_{u,k}^{-} + \mathbf{K}_k(\mathbf{h}_{R,k} - \check{\mathbf{H}}_{L,k}\hat{\mathbf{s}}_{u,k}^{-})$
7: Update the error covariance:
$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k\check{\mathbf{H}}_{L,k})\mathbf{P}_k^{-}$
8: **end for**

where $\check{\mathbf{H}}_{L,k} = [\bar{\mathbf{H}}_{L,k} \ \ \mathbf{0}_{(N^2-Nl)\times(2N-2l)}]$ is a zero-padded version of $\bar{\mathbf{H}}_{L,k}$ to account for the velocity values stacked in the state vector $\mathbf{s}_{u,k}$. Observe that we only measure the pairwise distances between the nodes, not their relative velocities. Meanwhile, note that the resulting noise term (ξ_k) does not have i.i.d entries [2]. To estimate the node locations from the described state and linearized measurement model, we propose to use the KF. The KF algorithm fitting our modified model is shown in Algorithm 1. In the algorithm, $\mathbf{P}_k$, $\mathbf{R}_k$ and $\mathbf{Q}$ are the covariance matrices of the state error, the measurement noise, and the process noise, respectively.

Assuming that we have information about the statistical properties of the movement model, we have perfect knowledge about

$$\mathbf{Q} = \mathbb{E}\{\mathbf{w}_k\mathbf{w}_k^T\} = \begin{bmatrix} \mathbf{0}_{(2N-2l)\times(2N-2l)} & \mathbf{0}_{(2N-2l)\times(2N-2l)} \\ \mathbf{0}_{(2N-2l)\times(2N-2l)} & \sigma_w^2\mathbf{I}_{2N-2l} \end{bmatrix}, \tag{13}$$

where $\mathbb{E}$ stands for the statistical expectation. The covariance matrix of the measurement noise can be defined as

$$\mathbf{R}_k = \mathbb{E}\{\xi_k\xi_k^T\}. \tag{14}$$

To compute this covariance matrix, let us rewrite (11) as

$$\xi_k = \mathbf{G}_k\text{vec}(\hat{\mathbf{D}}_k), \tag{15}$$

where

$$\mathbf{G}_k = -0.5\big((\bar{\mathbf{X}}_{a,k}^{\dagger}\bar{\mathbf{X}}_{u,k})^T \otimes \mathbf{J}\big). \tag{16}$$

By considering an appropriate deterministic selection matrix $\mathbf{\Psi}$ (similar to the one defined in [2]), we have

$$\text{vec}(\hat{\mathbf{D}}_k) = \mathbf{\Psi}\mathbf{r}_k, \tag{17}$$

where $\mathbf{r}_k = [r_{1,l+1,k}, \dots, r_{N-1,N,k}]^T$ is the column vector of pairwise distance measurements where the entries corresponding to distances between anchors are removed. Thus,

$$\mathbf{G}_k\text{vec}(\hat{\mathbf{D}}_k) = \mathbf{G}_k\mathbf{\Psi}\mathbf{r}_k. \tag{18}$$

Using (18), we can rewrite (14) as

$$\begin{aligned} \mathbf{R}_k &= \mathbb{E}\{\xi_k\xi_k^T\}, \\ &= \mathbf{G}_k\mathbf{\Psi}\mathbb{E}\{\mathbf{r}_k\mathbf{r}_k^T\}\mathbf{\Psi}^T\mathbf{G}_k^T. \end{aligned} \tag{19}$$

From (1), and by assuming a sufficiently small noise power, we have

$$\begin{aligned} r_{i,j,k}^2 - d_{i,j,k}^2 &= 2d_{i,j,k}q_{i,j,k} + q_{i,j,k}^2, \\ &\approx 2d_{i,j,k}q_{i,j,k}, \end{aligned} \tag{20}$$

and therefore

$$\text{cov}(r_{i,j,k}^2) \approx 4r_{i,j,k}^2\sigma_{i,j,k}^2. \tag{21}$$

Based on (20) and (21), (19) can be rewritten as

$$\mathbf{R}_k = \mathbf{G}_k\mathbf{\Psi}\mathbf{S}_k\mathbf{\Psi}^T\mathbf{G}_k^T, \tag{22}$$

where

$$\mathbf{S}_k = 4\ \text{diag}(r_{1,l+1,k}^2\sigma_{1,l+1,k}^2, \dots, r_{N-1,N,k}^2\sigma_{N-1,N,k}^2). \tag{23}$$

It is worth mentioning that since $\mathbf{G}_k$ is a function of the unknown locations to be estimated (as is clear from (16)), $\mathbf{R}_k$ itself will be dependent on the unknown locations, which makes the estimation of $\mathbf{R}_k$ non-trivial. Since the performance of the KF is sensitive to the estimation of $\mathbf{R}_k$, to achieve a better estimate we repeat the KF loop p times for every snapshot and use the location estimates obtained in the previous iteration to compute $\mathbf{R}_k$ for the current iteration. However, both $\check{\mathbf{H}}_{L,k}\mathbf{P}_k^{-}\check{\mathbf{H}}_{L,k}^T$ and $\mathbf{R}_k$, are rank deficient according to their definitions described earlier. For this reason, we use the Kalman gain expression as shown in line 5 of the Algorithm 1 [7, 8].

In the following we briefly explain the other algorithms under consideration for comparison. One solution considering the aforementioned measurement model consists of localizing the unknown nodes in each snapshot of the movement independent of the other snapshots. The corresponding cost function to be minimized can then be given by [2]

$$\min_{\mathbf{x}_{u,k}} \ (\bar{\mathbf{H}}_{L,k}\mathbf{x}_{u,k} - \mathbf{h}_{R,k})^T\mathbf{W}_k(\bar{\mathbf{H}}_{L,k}\mathbf{x}_{u,k} - \mathbf{h}_{R,k}), \tag{24}$$

where the weighting matrix ($\mathbf{W}_k$) is chosen to result in the Markov estimate, which according to [2, 9] can be given by

$$\mathbf{W}_k = \mathbb{E}\{\xi_k\xi_k^T\}^{\dagger} = \mathbf{R}_k^{\dagger}. \tag{25}$$

The solution of (24) can be expressed as

$$\hat{\mathbf{x}}_{u,k} = (\bar{\mathbf{H}}_{L,k}^T\mathbf{W}_k\bar{\mathbf{H}}_{L,k})^{-1}\bar{\mathbf{H}}_{L,k}^T\mathbf{W}_k\mathbf{h}_{R,k}. \tag{26}$$

The WMDS algorithm starts with $\mathbf{W}_0 = \mathbf{I}_{N^2-Nl}$, which corresponds to ordinary least squares (ordinary LS), and it should be iterated p times at each snapshot to achieve a good accuracy. We also consider the anchorless algorithm based on the EKF explained in [4, 5] where similarly to the proposed KF, nodes locations and velocities are incorporated in a state-space model. Although velocity measurements of the nodes aid cooperative network localization, in practice it requires the use of Doppler sensors, which increases the implementation cost as well as the computational complexity, and hence, we avoid doing measurements of the node velocities.

4. LOCALIZATION ACCURACY

The lower bound on the variance of the estimate for discrete-time filtering problems can be computed via the PCRB. The recursive PCRB derived in [6] provides a formula for updating the posterior Fisher information matrix (FIM) from one snapshot to the next. According to [6], the sequence $\mathscr{J}_k$ of the posterior FIM for a linear process and measurement model boils down to the following recursive formula

$$\mathscr{J}_k = \left(\mathbf{Q} + \mathbf{\Phi}\mathscr{J}_{k-1}^{-1}\mathbf{\Phi}^T\right)^{-1} + \breve{\mathbf{H}}_{L,k}^T \mathbf{R}_k^{\dagger} \breve{\mathbf{H}}_{L,k}, \qquad (27)$$

where all the parameters have been defined earlier except for the fact that $\mathbf{R}_k$ should be calculated using the true locations in (16) and $\breve{\mathbf{H}}_{L,k}$ should be calculated using the noise-free double-centered distance matrix $\mathbf{B}_k$. The PCRB for the location of all unknown nodes is then given by

$$\text{PCRB}_k^{\text{KF}} = \sum_{n=1}^{2(N-l)} [\mathscr{J}_k^{-1}]_{n,n}. \qquad (28)$$

Furthermore, one way to validate the PCRB is that the trace of the upper left block of the state error covariance $\mathbf{P}_k$ of the KF, which corresponds to location estimation errors, should also attain the PCRB [10]. On the other hand, as explained in [2], the covariance matrix of the WMDS location estimation errors should attain the CRB for sufficiently small noise. Hence, we can calculate the CRB as

$$\text{CRB}_k^{\text{WMDS}} = \text{tr}\left((\bar{\mathbf{H}}_{L,k}^T \mathbf{R}_k^{\dagger} \bar{\mathbf{H}}_{L,k})^{-1}\right), \qquad (29)$$

where $\mathbf{R}_k$ should be calculated using the true locations in (16) and $\bar{\mathbf{H}}_{L,k}$ should be calculated using the noise-free double-centered distance matrix $\mathbf{B}_k$. We will investigate the performance of the explained algorithms by considering these bounds to see how good they perform.

5. SIMULATION RESULTS

In this section, we compare the performances of the explained algorithms. We consider a fully connected network of $N = 10$ sensors, living in a two-dimensional space. To be able to linearize the measurement model in the cooperative network, we consider $l = 4$ anchor nodes, which for the sake of simplicity are considered to be fixed at the locations $(0,0)$m, $(0,1000)$m, $(1000,0)$m and $(1000,1000)$m. It is notable that mobile anchor nodes with known locations can also be considered, in which case a simple rigid transformation will be required in every snapshot of the movement to recover the true locations. The mobile nodes are considered to be initially deployed in an area of 1000m $\times$1000m determined by the locations of the anchors. We consider the process model and the measurement model explained in the previous sections. To be able to quantitatively compare the performances of the algorithms under consideration, we consider the positioning mean squared error (PMSE) of the algorithms at the k-th snapshot, which is defined by

$$\text{PMSE} = \frac{\sum_{m=1}^{M} \sum_{n=l+1}^{N} e_{n,m,k}^2}{M}, \qquad (30)$$

where $e_{n,m,k}$ represents the distance between the real location of the n-th node and its estimated location at the m-th Monte Carlo (MC) trial of the k-th snapshot. All simulations are averaged over $M = 100$ independent MC runs where in each run the nodes move toward random directions starting from random initial locations.

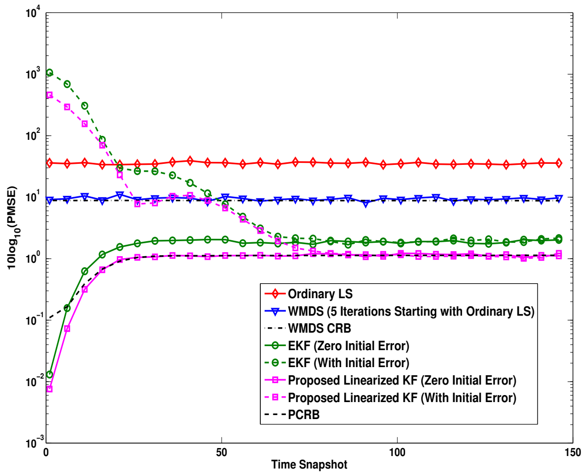

Figure 1: PMSE versus time snapshots for $N = 10$ and $l = 4$

As explained earlier, the distance measurements are impaired by additive noise. The CRB of range estimation in an additive white Gaussian (AWGN) channel with attenuation is derived in [11], where the range information is obtained from both the time delay and the amplitude of the received signal. The results show that the CRB is inversely proportional to the signal to noise ratio (SNR) of the transmissions and directly proportional to the distance powered by the path loss exponent (κ). Therefore, for a free space model ($\kappa = 2$), we consider a constant $\gamma = d_{i,j,k}^2/\sigma_{i,j,k}^2$, which acts like the SNR and punishes the longer distances with larger measurement errors. Note that this additive noise model is widely used in the literature [2, 5]. In addition, for all the simulations $T_s = 0.1$sec and $\sigma_w = 0.1$.

In order to acquire a complete picture of the performance of the cooperative algorithms (ordinary LS, WMDS with 5 iterations per snapshot starting with ordinary LS, EKF as explained in Section 3 and the proposed linearized KF), we plot the three following complementary figures. Fig. 1 illustrates the PMSE performance for $\gamma = 50$dB during a time span of 150 consecutive snapshots. The figure basically illustrates the tracking capability of the algorithms under consideration. To this aim, the PMSE performance of the EKF and the proposed KF is plotted once initialized with the true locations and once initialized with highly erroneous locations. As is clear from the figure, the EKF and the proposed KF converge to their achievable accuracy after some iterations even when initialized with a large error. For the LS and the WMDS, the PMSE is calculated independently from the previous snapshots and hence the initialization has no effect. As can be seen from the figure, the proposed linearized KF achieves a lower PMSE compared to all the other algorithms and attains the PCRB. The WMDS attains the CRB; however, it cannot attain the PCRB and there is a performance gap between the WMDS and the proposed KF.

Fig. 2 depicts the PMSE performance of the algorithms versus γ at snapshot $k = 100$, where according to Fig. 1 all the

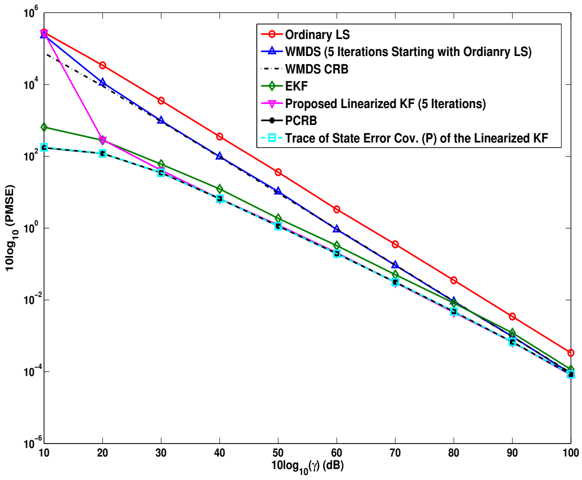

Figure 2: PMSE versus γ for $N = 10$ and $l = 4$

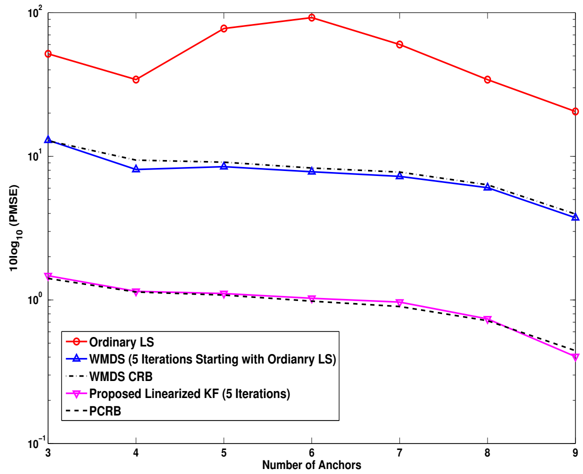

Figure 3: PMSE versus number of anchors for for $N = 10$

algorithms have converged to their best achievable accuracy. As can be seen from the figure, the proposed KF attains the PCRB and outperforms all the other algorithms. The performance improvement is significant compared to WMDS over a large span of γ which shows that an optimal adaptive process leads to a better localization accuracy. This is due to the fact that the proposed KF, similar to the EKF, continuously monitors the movement process and exploits this knowledge to improve the accuracy. The improvement over EKF is also considerable especially for the γ values over 30dB. Moreover, the trace of the upper left block of $\mathbf{P}_k$ also attains the PCRB, which in a sense validates that the KF is performing optimally in the current setup.

Finally, Fig. 3 shows the PMSE performance of the anchored algorithms versus the number of anchors for $\gamma =$ 50dB at snapshot $k = 100$. As can be seen from the figure, both WMDS and proposed KF attain their corresponding CRBs (WMDS CRB and PCRB) and the proposed KF has approximately 10 times lower PMSE compared to WMDS for different number of anchors. As well, the slight slope of the curves and their corresponding CRBs depicts the fact that the PMSE performance of the anchored algorithms under consideration can slightly be improved by increasing the number of anchors.

6. CONCLUSIONS

Most of the recently proposed algorithms for cooperative mobile network localization are suboptimal in estimation accuracy. This is due to the fact that they are either based on a suboptimal approach (MDS) or based on an algorithm which is only optimal for one snapshot of the mobile scenario. Thus, they can attain the CRB of every snapshot, but they cannot attain the PCRB of the dynamic filtering problem. To resolve this inefficiency, we have proposed to employ the location of the anchors to linearize the measurements ending up with a set of linear equations in the unknown locations. This allows us to use KF over this modified measurement model. It has been illustrated that the proposed algorithm attains the PCRB of mobile location estimation and outperforms comparable anchorless and anchored mobile localization algorithms. As future work, we will analytically prove that the estimation error of the proposed algorithm attains the PCRB. Moreover, we will develop this algorithm for the case where there is no knowledge about the statistical properties of the process model.

REFERENCES

[1] J. A. Costa, N. Patwari, and A. O. Hero, "Distributed Weighted MDS for Node Localization in Sensor Networks," *ACM Trans. on Sensor Networks*, vol. 2, pp. 39–64, Feb. 2006.

[2] F. K. W. Chan, and H. C. So, "Efficient Weighted Multidimensional Scaling for Wireless Sensor Networks," *IEEE Trans. on Signal Proc.*, vol. 57, pp. 4548–4553, Nov. 2009.

[3] U. Ferner, H. Wymeersch, and M. Z. Win, "Cooperative Anchorless Localization for Large Dynamic Networks" in *Proc. Intl. Conf. on Ultra-Wideband. ICUWB*, Sept. 2008, pp. 181-185.

[4] H. Jamali Rad, A. Amar, and G. Leus, "Cooperative Mobile Network Localization Via Subspace Tracking" to appear in *Proc. of IEEE ICASSP*, May. 2011.

[5] L. Dong, "Cooperative Network Localization via Node Velocity Estimation," in *Proc. IEEE WCNC*, April 2009, pp. 1–6.

[6] P. Tichavski, H. Muravchik, and A. Nehorai, "Posterior Cramer-Rao Bounds for Discrete-Time Nonlinear Filtering," *IEEE Trans. on Signal Proc.*, vol. 46, pp. 1386–1396, May 1998.

[7] F. H. Nieto and V. M. Guerrero, "Kalman Filter for Singular and Conditional State-Space Models When the System State and Observational Error Are Correlated," *Stat. and Prob. Letters*, vol. 22, pp. 303–310, 1995.

[8] H. Ogawa, "An Operator Pseudo-Inversion Lemma," *SIAM J. of Appl. Math.*, vol. 48, No. 6, pp. 1527–1531, Dec. 1988.

[9] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Address: Englewood Cliffs, NJ: Prentice-Hall, 1989.

[10] R. M. Taylor, B. P. Flanagan, and J. A. Uber, "Computing the Recursive Posterior Cramer-Rao Bound for a Nonlinear Nonstationary System" in *Proc. IEEE ICASSP*, April. 2003, pp. 673-676.

[11] Y. Wang, G. Leus, A.-J. van der Veen, "Cramer-Rao Bound for Range Estimation," in *Proc. IEEE ICASSP*, May 2009, pp. 3301–3304.