Localization of Wireless Sensor Nodes with erroneous anchors via EM algorithm

Mei Leng and Yik-Chung Wu

Department of Electrical and Electronic Engineering

The University of Hong Kong

Email : {meileng, ycwu}@eee.hku.hk

Abstract—**Finding locations of the sensor nodes in Wireless Sensor Network has been an active research area in recent years. One important category of approaches uses distance measurements between anchors and sensors to localize the unknown node. However, most approaches assume that the anchor positions are perfectly known, while in practice the anchor positions may not be accurate due to estimation errors as well as observation errors. In this paper, we study the localization of wireless sensor node with erroneous anchors, and propose an EM estimator which iteratively refines the anchor positions and estimates the sensor location. Simulation results shows that the EM estimator converges in a few iterations and outperforms the existing robust least squares algorithm.**

I. INTRODUCTION

Wireless Sensor Network (WSN), emerged as an important research area in recent years, consists of many small-scale miniature devices (or sensor nodes) capable of onboard sensing, computing and communications. WSNs are used in industrial and commercial applications to monitor data that would be difficult or inconvenient to monitor using wired equipment, such as monitoring the health status of environment, controlling industrial machines and home appliances, fire detection and object tracking, etc. [1] [2]. In most applications, the data collected by the sensor nodes are only meaningful with the location information of the corresponding sensors. Traditionally, Global Positioning System (GPS) provides solutions in outdoor environments [3]. However, this is a costly option and is not suitable for power-limited sensor nodes in WSNs. Therefore, instead of equipping the sensor nodes with extra GPS receivers, it is necessary to design algorithms which utilizes local information to localize the sensor nodes in WSNs.

Generally, the sensor network localization problem can be stated as follows. Assuming the knowledge of positions of certain nodes (called "*anchors*") in the network, together with pairwise distance measurements between anchors and other sensor nodes whose positions are unknown (called "*sensors*"), the localization problem is to determine positions of all sensors. In practice, the distance measurements are obtained by either timing or signal strength information between anchors and sensors, and the measurement error can be appropriately modeled as Gaussian random variables [3].

In the literature, a number of algorithms have been designed to reduce effects of the measurement error and recover sensors' positions as accurate as possible (see [4]–[7] and references therein), however most do not consider the inaccuracies in the anchors' positions. Since the locations of anchors are typically obtained by either GPS or human surveillance, both techniques introduce errors inevitably. For example, the accuracy of civilian GPS is limited to about 15 meters, while the human surveying is prone to observation errors [8]. Moreover, in a level-by-level localization scheme, any sensor can act as an anchor after being localized, and the accuracy of this kind of secondary anchors highly depends on the estimation errors of their own. The uncertainty caused by these errors will present in the anchor positions and must be taken care of.

To solve the localization problem with erroneous anchors, a distributed Second-Order Cone Programming (SOCP) and a centralized Semi-Definite Programming (SDP) based algorithms were proposed in [8] and [9]. Although these algorithms are powerful and produce good results, computational complexity of SOCP and SDP poses a challenge in implementation in WSNs. Another distributed localization scheme with erroneous anchors was proposed by Liu *et al.* in [10]. It adopts the level-by-level localization scheme, and therefore reduces the network localization problem into multiple single sensor localizations with erroneous anchors. For each single node localization, a simple robust least squares (RLS) estimator is used to handle the anchor uncertainties.

Obviously, the level-by-level localization scheme suffers from error propagation, in which the uncertainties of secondary anchors affect the localization accuracy in the next level. Therefore, developing accurate localization algorithms under erroneous anchors is an important building block and of great interest. In this paper, both measurement errors and anchor uncertainties are incorporated in the system model. An EM estimator which iteratively refines the anchor positions and estimates the sensor location is derived. Simulation results show that the EM estimator converges in a few iterations and outperforms RLS estimator in [10], and therefore it presents great potential in controlling the error propagation for localization across the whole network.

II. THE LOCALIZATION PROBLEM

In this paper, we consider the localization of a sensor with unknown location $\mathbf{s} = [x, y]^T$, where x_n and y_n indicate coordinates on the x-axis and y-axis, respectively. The location of the nth anchor is denoted as $\mathbf{a}_n^o = [x_n^o, y_n^o]^T$ and $\mathbf{a}_n^o \in \mathbb{R}^{2 \times 1}$. With a measurement error, the distance measurement between

$$\underbrace{\begin{bmatrix} r_1 - r_{1'} + \|\mathbf{a}_{1'}\|^2 - \|\mathbf{a}_1\|^2 \\ \vdots \\ r_N - r_{N'} + \|\mathbf{a}_{N'}\|^2 - \|\mathbf{a}_N\|^2 \end{bmatrix}}_{\triangleq \mathbf{b}} = \underbrace{\begin{bmatrix} 2(\mathbf{a}_{1'}^o - \mathbf{a}_1^o)^T \\ \cdots \\ 2(\mathbf{a}_{N'}^o - \mathbf{a}_N^o)^T \end{bmatrix}}_{\triangleq \mathbf{H}} \mathbf{s} + \underbrace{\begin{bmatrix} 2(\mathbf{a}_1^T \Delta \mathbf{a}_1 - \mathbf{a}_{1'}^T \Delta \mathbf{a}_{1'}) + \varepsilon_1 - \varepsilon_{1'} \\ \vdots \\ 2(\mathbf{a}_N^T \Delta \mathbf{a}_N - \mathbf{a}_{N'}^T \Delta \mathbf{a}_{N'}) + \varepsilon_N - \varepsilon_{N'} \end{bmatrix}}_{\triangleq \mathbf{e}}. \tag{4}$$

the nth anchor and the sensor is denoted as r_n and is given by [8]

$$r_n = \|\mathbf{s} - \mathbf{a}_n^o\|^2 + \varepsilon_n, \tag{1}$$

where ε_n is a Gaussian random variable with zero mean and variance σ_ε^2, i.e., $\varepsilon_n \sim \mathcal{N}(0, \sigma_\varepsilon^2)$. It is assumed that measurement errors from different anchors are independent, and we have $\mathbb{E}\{\varepsilon_i \varepsilon_j\} = 0$ for $i \neq j$.

Other than measurement errors, the uncertainty presented in the anchor position also serves as a major error source and should be alleviated. Denote the observed location of the nth anchor as $\mathbf{a}_n = \mathbf{a}_n^o - \Delta \mathbf{a}_n$, where $\Delta \mathbf{a}_n$ is the uncertainty in the nth anchor position. Since this uncertainty is due to numerous independent random processes, it can be assumed to follow a multi-variate Gaussian distribution with zero mean and covariance matrix $\mathbf{\Sigma_a} \in \mathbb{R}^{2 \times 2}$, i.e., $\Delta \mathbf{a}_n \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma_a})$. Furthermore, it is assumed that the uncertainties in different anchors are independent, i.e., $\mathbb{E}\{\Delta \mathbf{a}_i \Delta \mathbf{a}_j^T\} = \mathbf{0}$ for $i \neq j$. Taking into consideration the anchor uncertainties, the distance measurement r_n is expressed as

$$\begin{aligned} r_n &= \|\mathbf{s} - (\mathbf{a}_n + \Delta \mathbf{a}_n)\|^2 + \varepsilon_n, \\ &= \|\mathbf{s}\|^2 - 2(\mathbf{a}_n + \Delta \mathbf{a}_n)^T \mathbf{s} + \|\mathbf{a}_n + \Delta \mathbf{a}_n\|^2 + \varepsilon_n. \end{aligned} \tag{2}$$

Clearly, r_n is a non-linear function with respect to $\mathbf{s}$ due to the existence of $\|\mathbf{s}\|^2$. Since the term $\|\mathbf{s}\|^2$ is independent of anchor indices, the function (2) can be easily linearized by subtracting another distance measurement $r_{n'}$ from r_n, and the subtraction result is given by

$$\begin{aligned} r_n - r_{n'} &= 2[(\mathbf{a}_{n'} + \Delta \mathbf{a}_{n'}) - (\mathbf{a}_n + \Delta \mathbf{a}_n)]^T \mathbf{s} \\ &+ \|\mathbf{a}_n\|^2 - \|\mathbf{a}_{n'}\|^2 + 2(\mathbf{a}_n^T \Delta \mathbf{a}_n - \mathbf{a}_{n'}^T \Delta \mathbf{a}_{n'}) \\ &+ \|\Delta \mathbf{a}_n\|^2 - \|\Delta \mathbf{a}_{n'}\|^2 + \varepsilon_n - \varepsilon_{n'}. \end{aligned} \tag{3}$$

Ignoring the second order terms $\|\Delta \mathbf{a}_n\|^2$ and $\|\Delta \mathbf{a}_{n'}\|^2$, and stacking N subtraction results in the matrix form, we have equation (4) on top of this page.

Since all the measurement errors $\{\varepsilon_n\}_{n=1}^N$ and anchor uncertainties $\{\Delta \mathbf{a}_n\}_{n=1}^N$ are independent Gaussian variables, their linear combinations are also Gaussian, and therefore each element in the vector $\mathbf{e}$ is Gaussian with zero mean and covariance matrix $\mathbf{\Sigma_e} \in \mathbb{R}^{N \times N}$, where the $(i, j)^{th}$ element of $\mathbf{\Sigma_e}$ is given by

$$(\mathbf{\Sigma_e})_{ij} = \begin{cases} 2\sigma_\varepsilon^2 + 4(\mathbf{a}_i^T \mathbf{\Sigma_a} \mathbf{a}_i + \mathbf{a}_{i'}^T \mathbf{\Sigma_a} \mathbf{a}_{i'}) & i = j; \\ \mathbb{E}\{\varepsilon_{i'} \varepsilon_{j'}\} + 4\mathbf{a}_{i'}^T \mathbb{E}\{\Delta \mathbf{a}_{i'} \Delta \mathbf{a}_{j'}^T\} \mathbf{a}_{j'} & i \neq j. \end{cases} \tag{5}$$

in which the facts $\mathbb{E}\{\Delta \mathbf{a}_i \Delta \mathbf{a}_j^T\} = \mathbf{0}$ and $\mathbb{E}\{\varepsilon_i \varepsilon_j\} = 0$ for $i \neq j$ have been used. Notice that the subtracted measurements $\{r_{n'}\}_{n=1}^N$ can be provided by either single measurement or multiple measurements. In the former case, $r_{1'} = \cdots = r_{N'} =$

r_0, which results in correlations between different elements of $\mathbf{e}$, and (5) can be written as

$$(\mathbf{\Sigma_e})_{ij} = \begin{cases} 2\sigma_\varepsilon^2 + 4(\mathbf{a}_i^T \mathbf{\Sigma_a} \mathbf{a}_i + \mathbf{a}_0^T \mathbf{\Sigma_a} \mathbf{a}_0) & i = j; \\ \sigma_\varepsilon^2 + 4\mathbf{a}_0^T \mathbf{\Sigma_a} \mathbf{a}_0 & i \neq j. \end{cases}$$

In the latter case, we can set $r_{1'} \neq \cdots \neq r_{N'}$, and $\mathbf{\Sigma_e}$ becomes a diagonal matrix with its $(i, j)^{th}$ element as

$$(\mathbf{\Sigma_e})_{ij} = \begin{cases} 2\sigma_\varepsilon^2 + 4(\mathbf{a}_i^T \mathbf{\Sigma_a} \mathbf{a}_i + \mathbf{a}_{i'}^T \mathbf{\Sigma_a} \mathbf{a}_{i'}) & i = j; \\ 0 & i \neq j. \end{cases}$$

Compared to the former case, it can be seen that the latter case gives a simple expression for $\mathbf{\Sigma_e}$ at the expense of almost doubling the required anchors.

For the model $\mathbf{b} = \mathbf{Hs} + \mathbf{e}$ in (4), if $\mathbf{H}$ is known, it is easy to see that the minimum variance unbiased (MVU) estimator is the least square estimator and can be obtained as $\hat{\mathbf{s}}_{LS} = (\mathbf{H}^T \mathbf{\Sigma_e} \mathbf{H})^{-1} \mathbf{H}^T \mathbf{\Sigma_e} \mathbf{b}$ [11]. However, in presence of anchor uncertainties, $\mathbf{H}$ cannot be directly observed. Therefore, the localization task is to estimate $\mathbf{s}$ with observations $\mathbf{b}$ and $\mathbf{H}$ with model uncertainties.

III. EXPECTATION-MAXIMIZATION ALGORITHM FOR LOCALIZATION WITH ERRONEOUS ANCHORS

The EM algorithm is a general method of parameter estimation in the sense of maximizing the likelihood function with the knowledge of an incomplete data set [12], and here the term "incomplete" means that the data set has missing values or hidden variables embedded in the underlying distribution. In the localization problem under erroneous anchors, we are given an incomplete data set $\mathbf{b}$ with hidden variable $\mathbf{H}$, and are required to estimate the parameter $\mathbf{s}$.

Rewrite the model (4) as

$$\mathbf{b} = \left(\mathbf{I}_N \otimes \mathbf{s}^T \right) \mathbf{h} + \mathbf{e}, \tag{6}$$

where $\mathbf{I}_N$ is the $N \times N$ identity matrix, and $\mathbf{h} = \text{vec}\{\mathbf{H}^T\}$ is an $2N \times 1$ vector by stacking the N column vectors of $\mathbf{H}^T$ one after another. Since there exists anchor uncertainties in the vector $\mathbf{h}$, the log-likelihood function $p(\mathbf{b}|\mathbf{s})$ is given by

$$\log p(\mathbf{b}|\mathbf{s}) = \int_{\mathbf{h}} \log p(\mathbf{b}, \mathbf{h}|\mathbf{s}) d\mathbf{h}. \tag{7}$$

In other words, in order to maximize the likelihood function, we have to first remove the random effects of $\mathbf{h}$. Unfortunately, although the integration in (7) can be computed in closed-form under the assumption that $p(\mathbf{b}|\mathbf{h}, \mathbf{s})$ and $p(\mathbf{h})$ are Gaussian, the result is a complicated function of $\mathbf{s}$, and therefore the optimal solution for $\mathbf{s}$ cannot be obtained directly. The EM algorithm overcomes this problem by performing integration and estimating $\mathbf{s}$ iteratively, and it consists of two steps: the expectation step and the maximization step. These two steps are introduced as follows.

$$p(\mathbf{b}|\mathbf{h},\mathbf{s}) = \frac{1}{\sqrt{2\pi\|\mathbf{\Sigma_e}\|}}\exp\left\{-\frac{1}{2}\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\mathbf{h}\right]^T\mathbf{\Sigma_e}^{-1}\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\mathbf{h}\right]\right\}, \tag{10}$$

$$\mathcal{Q}(\mathbf{s},\mathbf{s}^{k-1}) \propto \left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\hat{\mathbf{h}}\right]^T\mathbf{\Sigma_e}^{-1}\left[\mathbf{b}-\left(\mathbf{I}_N\otimes s^T\right)\hat{\mathbf{h}}\right]$$
$$-2\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\hat{\mathbf{h}}\right]^T\mathbf{\Sigma_e}^{-1}\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\int_\mathbf{h}(\mathbf{h}-\hat{\mathbf{h}})p(\mathbf{h}|\mathbf{b},\mathbf{s}^{k-1})d\mathbf{h}$$
$$+\sum_{i=1}^N\mathbf{c}_i^T\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\left[\int_\mathbf{h}(\mathbf{h}-\hat{\mathbf{h}})(\mathbf{h}-\hat{\mathbf{h}})^Tp(\mathbf{h}|\mathbf{b},\mathbf{s}^{k-1})d\mathbf{h}\right]\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)^T\mathbf{c}_i. \tag{14}$$

E-Step: In order to remove the random effects of $\mathbf{h}$ and still give a tractable expression for $\mathbf{s}$, the E-step involves finding the expected value of $\log p(\mathbf{b},\mathbf{h}|\mathbf{s})$ with respect to $\mathbf{h}$ given $\mathbf{b}$ and the current parameter estimates $\mathbf{s}^{k-1}$ [12]. Denote this expected value as $\mathcal{Q}(\mathbf{s},\mathbf{s}^{k-1})$, we have

$$\mathcal{Q}(\mathbf{s},\mathbf{s}^{k-1}) = \int_\mathbf{h}\left[\log p(\mathbf{b},\mathbf{h}|\mathbf{s})\right]p(\mathbf{h}|\mathbf{b},\mathbf{s}^{k-1})d\mathbf{h}$$
$$= \int_\mathbf{h}\left[\log p(\mathbf{b}|\mathbf{h},\mathbf{s})\right]p(\mathbf{h}|\mathbf{b},\mathbf{s}^{k-1})d\mathbf{h}$$
$$+ \int_\mathbf{h}\left[\log p(\mathbf{h}|\mathbf{s})\right]p(\mathbf{h}|\mathbf{b},\mathbf{s}^{k-1})d\mathbf{h}. \tag{8}$$

Since $\mathbf{h}$ and $\mathbf{s}$ are independent, the second term in (8) does not depend on $\mathbf{s}$ and therefore can be neglected, and the function $\mathcal{Q}(\mathbf{s},\mathbf{s}^{k-1})$ is simplified as

$$\mathcal{Q}(\mathbf{s},\mathbf{s}^{k-1}) \propto \int_\mathbf{h}\left[\log p(\mathbf{b}|\mathbf{h},\mathbf{s})\right]p(\mathbf{h}|\mathbf{b},\mathbf{s}^{k-1})d\mathbf{h}. \tag{9}$$

According to the model (6), the probability density function (PDF) $p(\mathbf{b}|\mathbf{h},\mathbf{s})$ is given in equation (10) on top of this page, where $\mathbf{\Sigma_e}$ is given in (5). Putting PDF $p(\mathbf{b}|\mathbf{h},\mathbf{s})$ into equation (9), we have

$$\mathcal{Q}(\mathbf{s},\mathbf{s}^{k-1})$$
$$\propto \int_\mathbf{h}\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\mathbf{h}\right]^T\mathbf{\Sigma_e}^{-1}\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\mathbf{h}\right]$$
$$\times p(\mathbf{h}|\mathbf{b},\mathbf{s}^{k-1})d\mathbf{h}. \tag{11}$$

Notice that $\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\mathbf{h}\right]^T\mathbf{\Sigma_e}^{-1}\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\mathbf{h}\right]$ can be approximated using the second order Taylor's series as follow

$$\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\mathbf{h}\right]^T\mathbf{\Sigma_e}^{-1}\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\mathbf{h}\right]$$
$$\approx \left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\hat{\mathbf{h}}\right]^T\mathbf{\Sigma_e}^{-1}\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\hat{\mathbf{h}}\right]$$
$$-2\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\hat{\mathbf{h}}\right]^T\mathbf{\Sigma_e}^{-1}\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)(\mathbf{h}-\hat{\mathbf{h}})$$
$$+(\mathbf{h}-\hat{\mathbf{h}})^T\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)^T\mathbf{\Sigma_e}^{-1}\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)(\mathbf{h}-\hat{\mathbf{h}}), \tag{12}$$

where $\hat{\mathbf{h}}$ represents any vector at which $\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\mathbf{h}\right]^T\mathbf{\Sigma_e}^{-1}\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\mathbf{h}\right]$ is differentiable with respect to $\mathbf{h}$. Moreover, since the covariance matrix $\mathbf{\Sigma_e}$ must be positive semi-definite, its inverse is also positive semi-definite and therefore can be

decomposed as $\mathbf{\Sigma_e}^{-1} = \mathbf{C}\mathbf{C}^T$, where $\mathbf{C}$ is a $N\times N$ lower triangular matrix obtained by Cholesky decomposition. Denote the ith column vector of $\mathbf{C}$ as $\mathbf{c}_i$, the quadratic term $\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\mathbf{h}\right]^T\mathbf{\Sigma_e}^{-1}\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\mathbf{h}\right]$ can be further written as

$$\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\mathbf{h}\right]^T\mathbf{\Sigma_e}^{-1}\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\mathbf{h}\right]$$
$$\approx \left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\hat{\mathbf{h}}\right]^T\mathbf{\Sigma_e}^{-1}\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\hat{\mathbf{h}}\right]$$
$$-2\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\hat{\mathbf{h}}\right]^T\mathbf{\Sigma_e}^{-1}\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)(\mathbf{h}-\hat{\mathbf{h}})$$
$$+\sum_{i=1}^N\mathbf{c}_i^T\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)(\mathbf{h}-\hat{\mathbf{h}})(\mathbf{h}-\hat{\mathbf{h}})^T\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)^T\mathbf{c}_i, \tag{13}$$

Bring this approximated term into (11), we have equation (14) on top of this page. It is easy to see that if $\hat{\mathbf{h}} = \int_\mathbf{h}\mathbf{h}p(\mathbf{h}|\mathbf{b},\mathbf{s}^{k-1})d\mathbf{h}$, the second term in (14) equals zero. Notice that this represents the standard minimum mean squared estimation (MMSE) of $\mathbf{h}$ [11]. With

$$\hat{\mathbf{h}}_{MMSE}^{k-1} = \int_\mathbf{h}\mathbf{h}p(\mathbf{h}|\mathbf{b},\mathbf{s}^{k-1})d\mathbf{h}, \tag{15}$$

$$\hat{\mathbf{\Sigma}}_\mathbf{h}^{k-1} = \int_\mathbf{h}(\mathbf{h}-\hat{\mathbf{h}}_{MMSE}^{k-1})(\mathbf{h}-\hat{\mathbf{h}}_{MMSE}^{k-1})^Tp(\mathbf{h}|\mathbf{b},\mathbf{s}^{k-1})d\mathbf{h}, \tag{16}$$

the function (14) can be simplified as

$$\mathcal{Q}(\mathbf{s},\mathbf{s}^{k-1}) \propto$$
$$\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\hat{\mathbf{h}}_{MMSE}^{k-1}\right]^T\mathbf{\Sigma_e}^{-1}\left[\mathbf{b}-\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\hat{\mathbf{h}}_{MMSE}^{k-1}\right]$$
$$+\sum_{i=1}^N\mathbf{c}_i^T\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)\hat{\mathbf{\Sigma}}_\mathbf{h}^{k-1}\left(\mathbf{I}_N\otimes\mathbf{s}^T\right)^T\mathbf{c}_i. \tag{17}$$

The closed-form expressions of $\hat{\mathbf{h}}_{MMSE}^{k-1}$ and $\hat{\mathbf{\Sigma}}_\mathbf{h}^{k-1}$ are derived in the Appendix and given by (27) and (28) .

M-Step: The maximization step involves finding the optimal $\mathbf{s}$ which maximizes the function $\mathcal{Q}(\mathbf{s},\mathbf{s}^{k-1})$ over all the possible values of $\mathbf{s}$. That is,

$$\mathbf{s}^k = \arg\max_\mathbf{s}\mathcal{Q}(\mathbf{s},\mathbf{s}^{k-1}). \tag{18}$$

Since the function in (17) is convex with respect to $\mathbf{s}$, its peak corresponds to the unique and globally optimal estimate of $\mathbf{s}$.

To proceed, we take the first order derivative of $\mathcal{Q}(\mathbf{s}, \mathbf{s}^{k-1})$ with respect to $\mathbf{s}$ and set the result to zero, which gives

$$-2\left(\hat{\mathbf{H}}^{k-1}\right)^T \boldsymbol{\Sigma}_{\mathbf{e}}^{-1}(\mathbf{b} - \hat{\mathbf{H}}^{k-1}\mathbf{s})$$
$$+ 2\sum_{i=1}^{N}\left(\mathbf{c}_i^T \otimes \mathbf{I}_2\right)\hat{\boldsymbol{\Sigma}}_{\mathbf{h}}^{k-1}\left(\mathbf{c}_i \otimes \mathbf{I}_2\right)\mathbf{s} = 0, \qquad (19)$$

where $\hat{\mathbf{H}}^{k-1}$ is an $N \times 2$ matrix obtained as

$$\hat{\mathbf{H}}^{k-1} = \begin{bmatrix} \left(\hat{\mathbf{h}}_{MMSE}^{k-1}\right)_1, \left(\hat{\mathbf{h}}_{MMSE}^{k-1}\right)_2 \\ \cdots \\ \left(\hat{\mathbf{h}}_{MMSE}^{k-1}\right)_{2N-1}, \left(\hat{\mathbf{h}}_{MMSE}^{k-1}\right)_{2N} \end{bmatrix}, \qquad (20)$$

where $\left(\hat{\mathbf{h}}_{MMSE}^{k-1}\right)_i$ is the ith element of $\hat{\mathbf{h}}_{MMSE}^{k-1}$, and notice that $\hat{\mathbf{H}}^{k-1}$ actually gives refined positions for the erroneous anchors. Finally, from (19), the estimated sensor location in the kth iteration is given by

$$\mathbf{s}^k = \frac{\left(\hat{\mathbf{H}}^{k-1}\right)^T \boldsymbol{\Sigma}_{\mathbf{e}}^{-1}\mathbf{b}}{\left(\hat{\mathbf{H}}^{k-1}\right)^T \boldsymbol{\Sigma}_{\mathbf{e}}^{-1}\hat{\mathbf{H}}^{k-1} + \sum_{i=1}^{N}\left(\mathbf{c}_i^T \otimes \mathbf{I}_2\right)\hat{\boldsymbol{\Sigma}}_{\mathbf{h}}^{k-1}\left(\mathbf{c}_i \otimes \mathbf{I}_2\right)}. \qquad (21)$$

The EM algorithm iterates between (17) and (21) with the least square estimate of $\mathbf{s}$ as initialization.

IV. SIMULATION RESULTS

In this section, simulation results are presented to validate the effectiveness of the proposed EM estimator. The unknown sensor location is set to be $[0,0]^T$, and 4 anchors with co-ordinates $\{[x_n^o, y_n^o]^T\}_{n=1}^4$ are uniformly distributed in an area $(-10, 10) \times (-10, 10)$. For simplicity, we define $\boldsymbol{\Sigma}_{\mathbf{a}} = \sigma_a^2 \mathbf{I}_2$ and the anchor uncertainty is measured in the scale of dB, i.e., $10 \log(1/\sigma_a^2)$. For each point in the figures, 10000 simulation runs were performed to obtain the average performance. Since the results for x-coordinate are similar to that on the y-axis, only the results on the x-axis are shown in the following.

Fig. 1 shows the mean squared error (MSE) for the estimated x-coordinate of the sensor position versus the number of iterations in the proposed EM estimator, where the variance of the measurement error is set to be 1. As shown in the figure, the EM estimator converges within 5 iterations. Furthermore, when the anchor uncertainty is small, it requires only one or two iterations to converge.

Next, we compare the performance of the proposed EM estimator with the robust least squares (RLS) estimator [10], which is expressed as $\hat{\mathbf{s}}_{RLS} = \left(\mathbf{H}^T\boldsymbol{\Sigma}_{\mathbf{e}}\mathbf{H} + \boldsymbol{\Sigma}_{\mathbf{H}}\right)^{-1}\mathbf{H}^{-1}\boldsymbol{\Sigma}_{\mathbf{e}}\mathbf{b}$ with $\boldsymbol{\Sigma}_{\mathbf{H}}$ as the covariance matrix of $\mathbf{H}$. Fig. 2 shows the MSE of the estimated x-coordinate of the sensor position versus the anchor uncertainty. As shown in the figure, the proposed EM estimator performs better than the RLS estimator after two iterations, especially when the anchor uncertainty is large.

V. CONCLUSIONS

In this paper, we studied the localization problem for a single sensor with erroneous anchors. A novel EM estimator
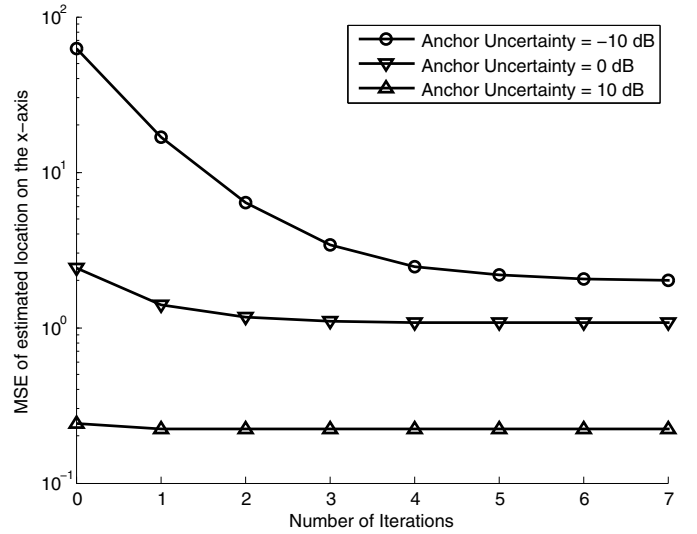

Fig. 1. MSE of the estimated location versus the number of iterations conducted by the EM estimator.
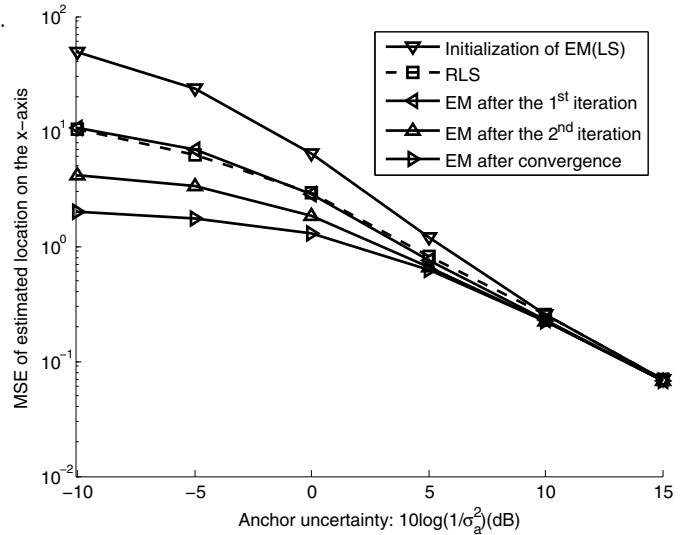

Fig. 2. MSE of the estimated location v.s. the anchor uncertainty.

which iteratively refines anchor positions and estimates the unknown sensor location was proposed. Simulation results show that this estimator converges in a few iterations and out-performs the existing algorithm given in [10]. Our future work will focus on extension of this estimator to the localization of the whole network.

APPENDIX
FINDING $\hat{\mathbf{h}}_{MMSE}^{k-1}$ AND $\hat{\boldsymbol{\Sigma}}_{\mathbf{h}}^{k-1}$

Notice that the two terms $\hat{\mathbf{h}}_{MMSE}^{k-1}$ and $\hat{\boldsymbol{\Sigma}}_{\mathbf{h}}^{k-1}$ in (15) and (16) represent mean value and covariance matrix of $p(\mathbf{h}|\mathbf{b}, \mathbf{s}^{k-1})$, respectively, we should first find $p(\mathbf{h}|\mathbf{b}, \mathbf{s}^{k-1})$. According to Bayes' formula, we have

$$p(\mathbf{h}|\mathbf{b}, \mathbf{s}^{k-1}) = \frac{p(\mathbf{b}|\mathbf{h}, \mathbf{s}^{k-1})p(\mathbf{h})}{p(\mathbf{b}|\mathbf{s}^{k-1})}. \qquad (22)$$

From (10), it is easy to see that $p(\mathbf{b}|\mathbf{h}, \mathbf{s}^{k-1})$ is Gaussian. On the other hand, for $p(\mathbf{h})$, we have $\mathbf{h} = \text{vec}\{\mathbf{H}\}$ according to (6), i.e.,

$$\mathbf{h} = [2(\mathbf{a}_{1'} - \mathbf{a}_1 + \Delta\mathbf{a}_{1'} - \Delta\mathbf{a}_1)^T,$$
$$\cdots, 2(\mathbf{a}_{N'} - \mathbf{a}_N + \Delta\mathbf{a}_{N'} - \Delta\mathbf{a}_N)^T]^T. \quad (23)$$

Define

$$\bar{\mathbf{h}} = [2(\mathbf{a}_{1'} - \mathbf{a}_1)^T, \cdots 2(\mathbf{a}_{N'} - \mathbf{a}_N)^T]^T, \quad (24)$$

and under the assumption $\Delta\mathbf{a}_n \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma_a})$, the vector $\mathbf{h}$ also follows Gaussian distribution and $p(\mathbf{h})$ can be written as

$$p(\mathbf{h}) = \frac{1}{\sqrt{2\pi\|\mathbf{\Sigma_h}\|}} \exp\{-\frac{1}{2}\left(\mathbf{h} - \bar{\mathbf{h}}\right)^T \mathbf{\Sigma_h}^{-1} \left(\mathbf{h} - \bar{\mathbf{h}}\right)\}, \quad (25)$$

where $\mathbf{\Sigma_h}$ is the covariance matrix of $\mathbf{h}$ with its $(i,j)^{th}$ element given by

$$(\mathbf{\Sigma_h})_{ij} = \begin{cases} 4\mathbb{E}\{(\Delta\mathbf{a}_i - \Delta\mathbf{a}_{i'})^T(\Delta\mathbf{a}_i - \Delta\mathbf{a}_{i'})\} & i = j; \\ 4\mathbb{E}\{(\Delta\mathbf{a}_i - \Delta\mathbf{a}_{i'})^T(\Delta\mathbf{a}_j - \Delta\mathbf{a}_{j'})\} & i \neq j. \end{cases}$$
$$= \begin{cases} 8\text{Tr}\{\mathbf{\Sigma_a}\} & i = j; \\ 4\mathbb{E}\{\Delta\mathbf{a}_{i'}^T\Delta\mathbf{a}_{j'}\} & i \neq j. \end{cases} \quad (26)$$

Similar to (5), there exists two cases: either $i' = j'$ or $i' \neq j'$. According to these cases, different expressions for $\mathbf{\Sigma_h}$ can be obtained, however the general form of $p(\mathbf{h})$ remains the same in (25).

As shown in [11], when both $p(\mathbf{b}|\mathbf{h}, \mathbf{s}^{k-1})$ and $p(\mathbf{h})$ follow Gaussian distributions, the posterior PDF $p(\mathbf{h}|\mathbf{b}, \mathbf{s}^{k-1})$ is also Gaussian, and we have

$$\hat{\mathbf{h}}_{MMSE}^{k-1} = \hat{\mathbf{\Sigma}}_{\mathbf{h}}^{k-1} \left[\left(\mathbf{I}_N \otimes \mathbf{s}^{k-1}\right) \mathbf{\Sigma_e}^{-1}\mathbf{b} + \mathbf{\Sigma_h}^{-1}\bar{\mathbf{h}}\right], \quad (27)$$
$$\hat{\mathbf{\Sigma}}_{\mathbf{h}}^{k-1} = \left[\mathbf{\Sigma_h}^{-1} + \left(\mathbf{I}_N \otimes \mathbf{s}^{k-1}\right) \mathbf{\Sigma_e}^{-1} \left(\mathbf{I}_N \otimes \mathbf{s}^{k-1}\right)^T\right]^{-1}. \quad (28)$$

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey", *Cumputer Networks*, 38(4), pp.393-422, Mar. 2002.
[2] N. Bulusu and S. Jha, *Wireless Sensor Networks: A Systems Perspective*. Artech House, 2005.
[3] N. Bulusu, John Heidenmann and Deborah Estrin, "GPS-less low-cost outdoor localization for very small devices", *IEEE Personal Communications*. pp. 28-34. Oct. 2000.
[4] R. Szewczyk, E. Osterweil, J. Polastre, M. Hamilton, A. Mainwaring, and D. Estrin, "Habitat monitoring with sensor networks", *Commun, ACM.*, vol. 47, pp.34-40, June 2004.
[5] J. Highrower and G. Borriello, "Location systems for ubiquitous computing", *Computers*. vol. 34, pp.57-66. Oct. 2001.
[6] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero III, R. L. Moses, and N. S. Correal, "Locating the nodes: Cooperative localization in wireless sensor networks", *IEEE Signal Process. Mag.*, 22(4), pp.54-69, July 2005.
[7] Y. T. Chan and K. C. Ho, "A simple and efficient estimator for hyperbolic location", *IEEE Trans. on Signal Processing*. 42(8), pp. 1905-1914, Aug. 1994.
[8] S. Srirangarajan, A. H. Twefic, and Zhi-Quan Luo, "Distributed sensor network localization using SOCP relaxation", *IEEE Trans. on Wireless Communications*. 7(12), pp. 4886-4894, Dec. 2008.
[9] Kenneth W. K. Lui, Wing-Kin Ma, H. C. So and Frankie K. W. Chan "Semi-Definite Programming Algorithms for Sensor Network Node Localization with Uncertainties in Anchor Positions and/or Propagation Speed", *IEEE Trans. on Signal Processing*. 57(2), pp. 752-763, Feb. 2009.
[10] J. Liu, Y. Zhang and F. Zhao, "Robust distributed node localization with error management", In *Proc. MobiHoc.*, Florence, Italy, May. 2006.
[11] S. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 1993.
[12] J. A. Blimes. "A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models", Technical report, University of California at Berkeley, April 1998.