Accurate Distributed Range-Based Positioning Algorithm for Wireless Sensor Networks

Frankie K. W. Chan and H. C. So

Abstract—**Localization of sensor nodes is a fundamental and important problem in wireless sensor networks. In this correspondence, a recursive distributed positioning algorithm is devised with the use of range measurements. Computer simulations are included to contrast the performance of the proposed approach with the conventional semi-definite relaxation positioning method as well as Cramér–Rao lower bound.**

Index Terms—**Distributed algorithm, position estimation, range-based measurements, wireless sensor networks.**

I. Introduction

The rapid advances in microelectromechanical systems and wireless communications increase the importance of wireless sensor network (WSN) for a wide variety of applications in inventory control, home and building automation as well as battlefield surveillance [1]–[3]. In WSNs, a large amount of low-cost, self-powered sensors are deployed to generate collaborative networks. Nevertheless, determining the coordinates of the sensor nodes is crucial for efficient functioning of these applications. Employing global positioning system which requires expensive hardware is not an appropriate solution for indoor sensor localization [4]. Instead, localization algorithms based on connectivity and distance measurements among sensors can produce better position estimates at lower operational cost.

Node localization methods can be generally categorized as the range-free and range-based approaches. One simple range-free method is that each sensor finds the high-connectivity anchors whose positions have already been known and then takes the centroid of the connected anchors to be its location [5]. Approximation point-in-triangulation [6] finds triangles formed by any three anchors which include the unknown sensor and then takes the center of gravity of the intersection of all these triangles to be the position estimate. On the other hand, range-based algorithms exploit distances/angles between nodes obtained from the pairwise time-of-arrival (TOA), time-difference-of-arrival (TDOA), received signal strength (RSS), and/or angle-of-arrival measurements. These algorithms can provide more reliable sensor positioning. Under white Gaussian noise assumption, nonlinear least squares estimator [7]–[9] can provide optimum estimation accuracy but sufficiently close initial guess is required due to the multimodality of the corresponding cost function. Utilizing the pairwise distance measurements constructed from the TOA or RSS information, the classical multidimensional scaling (MDS) [10], [11] is a computationally attractive range-based positioning technique but the requirement of fully connected scenario and centralized processing are the main drawbacks. Given the pairwise distances, the semi-definite relaxation (SDR) algorithm [12] transforms the sensor network localization problem to a convex optimization problem with the use of relaxed constraints. It can provide global solutions and has higher accuracy compared with MDS but the relaxation of the matrix rank affects the estimation performance. Another advantage of the SDR approach is that it can tackle partially-connected WSNs.

All these methods cannot be applied to environments in which central processor is unavailable or the computational capability is insufficient. In these situations, distributed algorithms for cooperative localization are employed and they can be classified as network mutlilateration and successive refinement [13]. In the former algorithms, sensors estimate the multi-hop range to the anchors found by, say, counting the number of hops and multiplying the average distance of a single hop. After that, multilateration is applied to estimate the sensor positions [14], [15]. On the other hand, positioning algorithms based on successive refinement recursively update the sensor positions through exchanging position information with neighbors [16]–[18]. In general, the position estimates of the latter are more reliable than the former at the expense of extra computations. However, in the above-mentioned distributed algorithms, the estimated sensor positions containing estimation errors are assumed to be exact and utilized for localization of their connected sensors or neighbors. In light of this, we develop a more suitable approach which includes the variances of positions in sensor positioning to enhance the estimation accuracy.

In this correspondence, a weighted least squares (WLS)-based algorithm which converts distance measurements obtained from TOA information into linear equations is devised for WSN node localization. The proposed algorithm belonging to the category of successive refinement has the advantages of i) distributed estimation, ii) able to work in partially-connected scenario, and iii) able to provide a position estimate as well as the covariance which shows the uncertainty of the estimate. The rest of the correspondence is organized as follows. In Section II, the problem formulation of sensor localization will be presented. Algorithm development as well as some practical considerations will be described in Section III. In Section IV, numerical examples will be provided to demonstrate the performance of the proposed algorithm by comparing with Cramér–Rao lower bound (CRLB) and the SDR algorithm [12] which can provide high estimation performance compared with existing distributed algorithms. Finally, concluding remarks are given in Section V.

II. Problem Formulation

We consider a WSN of M sensors on a two-dimensional (2-D) plane with first $k \geq 3$ of them being anchors whose positions are already known. Extension to three-dimensional (3-D) scenario is straightforward and the corresponding simulation is provided in Section IV. Let $\boldsymbol{\phi}_i = [x_i \; y_i]^T$ be the position of the ith sensor where ^{T} denotes the transpose operator. In the absence of measurement error, the one-way propagation time taken for the signal to travel from the ith sensor to the jth sensor, denoted by $t_{i,j}$, is

$$t_{i,j} = \frac{d_{i,j}}{c}, \quad i,j = 1,2,\ldots,M \tag{1}$$

where c is the speed of signal propagation. The range measurement based on $t_{i,j}$ in the presence of disturbance, denoted by $r_{i,j}$, is modeled as

$$r_{i,j} = r_{j,i} = d_{i,j} + q_{i,j} \quad i,j = 1,2,\ldots,M \tag{2}$$

where $d_{i,j} = \|\boldsymbol{\phi}_i - \boldsymbol{\phi}_j\|$ is the noise-free distance between the ith sensor and the jth sensor and $q_{i,j} \sim \mathcal{N}(0,\sigma^2)$ is the additive measurement noise in $r_{i,j}$. Here, $\|\cdot\|$ stands for the l_2 norm and $\mathcal{N}(\mu,\nu^2)$ denotes the Gaussian distribution with mean μ and variance ν^2. Based on the work of Patwari *et al.* [8], [13], the Gaussian noise model is adopted. Note that the WSN is not necessarily fully connected because

Manuscript received November 06, 2007; accepted April 06, 2009. First published May 05, 2009; current version published September 16, 2009. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Brian M. Sadler. This work was supported by the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CityU 119606).

The authors are with the Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong (e-mail: hcso@ee.cityu.edu.hk).

Digital Object Identifier 10.1109/TSP.2009.2022354

1053-587X/$26.00 © 2009 IEEE

the distance of far apart sensors will exceed the communication range. However, it is assumed that each sensor has at least three connections to uniquely determine its position. The objective is to estimate the positions of the $(M-k)$ sensors, namely, $\{\boldsymbol{\phi}_i\}_{i=k+1}^{M}$ based on the available distance measurements, $\{r_{i,j}\}$ as well as the positions of anchors, $\{\boldsymbol{\phi}_i\}_{i=1}^{k}$.

III. Algorithm Development

In this section, a distributed positioning algorithm which can deal with partially-connected WSNs will be derived. The basic operation of the proposed algorithm will be firstly introduced. Communication range and boundary constraints will then be incorporated for better position estimation.

A. Basic Algorithm

The basic algorithm is based on the two-step WLS approach by Chan *et al.* [19]. Note that their algorithm utilizes TDOA measurements and assumes the positions of the anchors are noise-free while the proposed algorithm uses TOA measurements and the positions of neighbors, excluding the anchors, contain estimation errors. When $\{q_{i,j}\}$ are sufficiently small, squaring both sides of (2) and rearranging yields

$$-2x_ix_j - 2y_iy_j + x_i^2 + y_i^2 \approx r_{i,j}^2 - x_j^2 - y_j^2. \quad (3)$$

Let $j = i_1, i_2, \ldots, i_{N_i}$, be the neighbor index of the ith sensor which has N_i neighbors, (3) can be expressed in matrix form

$$\mathbf{A}_i\boldsymbol{\psi}_i \approx \mathbf{b}_i \quad (4)$$

where

$$\begin{aligned}\mathbf{A}_i &= [\,-2\boldsymbol{\Phi}_i \quad \mathbf{1}_{N_i}\,]\\ \boldsymbol{\Phi}_i &= [\,\boldsymbol{\phi}_{i_1} \quad \boldsymbol{\phi}_{i_2} \quad \cdots \quad \boldsymbol{\phi}_{i_{N_i}}\,]^T\\ \mathbf{b}_i &= [\,r_{i,i_1}^2 - \|\boldsymbol{\phi}_{i_1}\|^2 \quad \cdots \quad r_{i,i_{N_i}}^2 - \|\boldsymbol{\phi}_{i_{N_i}}\|^2\,]^T\\ \boldsymbol{\psi}_i &= [\,\boldsymbol{\phi}_i^T \quad \boldsymbol{\phi}_i^T\boldsymbol{\phi}_i\,]^T.\end{aligned}$$

Here, $\mathbf{1}_l$ denotes the $l \times 1$ vector with all elements equal to unity. The WLS cost function to be minimized is

$$e_i = (\mathbf{A}_i\boldsymbol{\psi}_i - \mathbf{b}_i)^T\mathbf{W}_{\boldsymbol{\psi}_i}(\mathbf{A}_i\psi_i - \mathbf{b}_i). \quad (5)$$

Ignoring the constraint of $[\boldsymbol{\psi}_i]_1^2 + [\boldsymbol{\psi}_i]_2^2 = [\boldsymbol{\psi}_i]_3$ where $[\boldsymbol{\psi}_i]_l$ represents the lth element of $\boldsymbol{\psi}_i$, the WLS estimate of $\boldsymbol{\psi}_i$ is

$$\hat{\boldsymbol{\psi}}_i = (\mathbf{A}_i^T\mathbf{W}_{\boldsymbol{\psi}_i}\mathbf{A}_i)^{-1}\mathbf{A}_i^T\mathbf{W}_{\boldsymbol{\psi}_i}\mathbf{b}_i \quad (6)$$

and the corresponding covariance is given by [19]

$$\mathrm{cov}\,(\boldsymbol{\psi}_i) = \left(\mathbf{A}_i^T\mathbf{W}_{\boldsymbol{\psi}_i}\mathbf{A}_i\right)^{-1} \quad (7)$$

where $^{-1}$ represents the inverse operator. The optimum weighting matrix $\mathbf{W}_{\boldsymbol{\psi}_i}$ is obtained from the Markov estimate by expressing $\mathbf{A}_i = \tilde{\mathbf{A}}_i + \boldsymbol{\Delta}\mathbf{A}_i$ and $\mathbf{b}_i = \tilde{\mathbf{b}}_i + \boldsymbol{\Delta}\mathbf{b}_i$ where $\tilde{\mathbf{A}}_i$ and $\tilde{\mathbf{b}}_i$ stand for the noise-free values of $\mathbf{A}_i$ and $\mathbf{b}_i$, respectively, while $\boldsymbol{\Delta}\mathbf{A}_i$ and $\boldsymbol{\Delta}\mathbf{b}_i$ are the corresponding perturbations. The inverse of $\mathbf{W}_{\boldsymbol{\phi}_i}$ is derived as

$$\begin{aligned}\mathbf{W}_{\boldsymbol{\psi}_i}^{-1} &= \mathbb{E}\left\{(\mathbf{A}_i\boldsymbol{\psi}_i - \mathbf{b}_i)(\mathbf{A}_i\psi_i - \mathbf{b}_i)^T\right\}\\ &= \mathbb{E}\left\{(\tilde{\mathbf{A}}_i\boldsymbol{\psi}_i + \boldsymbol{\Delta}\mathbf{A}_i\boldsymbol{\psi}_i - \tilde{\mathbf{b}}_i - \boldsymbol{\Delta}\mathbf{b}_i)\right.\\ &\quad \left.\times\,(\tilde{\mathbf{A}}_i\boldsymbol{\psi}_i + \boldsymbol{\Delta}\mathbf{A}_i\boldsymbol{\psi}_i - \tilde{\mathbf{b}}_i - \boldsymbol{\Delta}\mathbf{b}_i)^T\right\}\\ &= \mathbb{E}\left\{(\boldsymbol{\Delta}\mathbf{A}_i\psi_i - \boldsymbol{\Delta}\mathbf{b}_i)(\boldsymbol{\Delta}\mathbf{A}_i\psi_i - \boldsymbol{\Delta}\mathbf{b}_i)^T\right\}\end{aligned}$$

where

$$\boldsymbol{\Delta}\mathbf{A}_i = \begin{bmatrix} -2\Delta\boldsymbol{\phi}_{i_1} & -2\Delta\boldsymbol{\phi}_{i_2} & \cdots & -2\Delta\boldsymbol{\phi}_{i_{N_i}} \\ & \mathbf{0}_{1\times N_i} & & \end{bmatrix}^T$$

$$\boldsymbol{\Delta}\mathbf{b}_i = \begin{bmatrix} 2d_{i_1}q_{i_1} + q_{i_1}^2 - 2\tilde{\boldsymbol{\phi}}_{i_1}^T\Delta\boldsymbol{\phi}_{i_1} - \Delta\boldsymbol{\phi}_{i_1}^T\Delta\boldsymbol{\phi}_{i_1} \\ 2d_{i_2}q_{i_2} + q_{i_2}^2 - 2\tilde{\boldsymbol{\phi}}_{i_2}^T\Delta\boldsymbol{\phi}_{i_2} - \Delta\boldsymbol{\phi}_{i_2}^T\Delta\boldsymbol{\phi}_{i_2} \\ \vdots \\ 2d_{i_{N_i}}q_{i_{N_i}} + q_{i_{N_i}}^2 - 2\tilde{\boldsymbol{\phi}}_{i_{N_i}}^T\Delta\boldsymbol{\phi}_{i_{N_i}} - \Delta\boldsymbol{\phi}_{i_{N_i}}^T\Delta\boldsymbol{\phi}_{i_{N_i}} \end{bmatrix}$$

with $\mathbb{E}, \mathbf{0}_{i\times j}, \tilde{\boldsymbol{\phi}}_{i_l}^T, \Delta\boldsymbol{\phi}_{i_l}^T$ denote the expectation operator, $i \times j$ zero matrix, true position of the lth neighbor of the ith sensor and the corresponding perturbation, respectively.

Ignoring the second order terms $\{q_{i_l}^2\}$, as well as $\{\Delta\boldsymbol{\phi}_{i_l}^T\Delta\boldsymbol{\phi}_{i_l}\}$ and replacing $d_{i,i_l}, \tilde{\boldsymbol{\phi}}_{i_l}$ by r_{i,i_l}, $\boldsymbol{\phi}_{i_l}$, respectively, $\mathbf{W}_{\boldsymbol{\psi}_i}^{-1}$ can be approximated as

$$\begin{aligned}&\mathbf{W}_{\boldsymbol{\psi}_i}^{-1}\\ &\quad\approx 4\mathbb{E}\left\{\left[\left(\mathbf{I}_{N_i} \otimes \boldsymbol{\phi}_i^T\right)\boldsymbol{\Delta}\mathbf{a}_i + \mathbf{R}_i\mathbf{q}_i - \left(\mathbf{I}_{N_i} \circ \boldsymbol{\Phi}_i^T\right)^T\boldsymbol{\Delta}\mathbf{a}_i\right]\right.\\ &\qquad\left.\times\left[\left(\mathbf{I}_{N_i} \otimes \boldsymbol{\phi}_i^T\right)\boldsymbol{\Delta}\mathbf{a}_i + \mathbf{R}_i\mathbf{q}_i - \left(\mathbf{I}_{N_i} \circ \boldsymbol{\Phi}_i^T\right)^T\boldsymbol{\Delta}\mathbf{a}_i\right]^T\right\}\\ &\quad= 4\left(\mathbf{I}_{N_i} \circ \left(\boldsymbol{\phi}_i\mathbf{1}_{N_i}^T - \boldsymbol{\Phi}_i^T\right)\right)^T\mathbf{C}_{\boldsymbol{\Phi}_i}\left(\mathbf{I}_{N_i} \circ \left(\boldsymbol{\phi}_i\mathbf{1}_{N_i}^T - \boldsymbol{\Phi}_i^T\right)\right)\\ &\qquad+ 4\mathbf{R}_i\mathbf{C}_{\mathbf{r}_i}\mathbf{R}_i^T\end{aligned}$$

where

$$\begin{aligned}\boldsymbol{\Delta}\mathbf{a}_i &= [\,\Delta\boldsymbol{\phi}_{i_1}^T \quad \Delta\boldsymbol{\phi}_{i_2}^T \quad \cdots \quad \Delta\boldsymbol{\phi}_{i_{N_i}}^T\,]^T\\ \mathbf{R}_i &= \mathrm{diag}\left(r_{i,i_1}, r_{i,i_2}, \cdots, r_{i,i_{N_i}}\right)\\ \mathbf{q}_i &= [\,q_{i,i_1} \quad q_{i,i_2} \quad \cdots \quad q_{i,i_{N_i}}\,]^T\\ \mathbf{C}_{\boldsymbol{\Phi}_i} &= \mathrm{cov}\left([\,\boldsymbol{\phi}_{i_1}^T \quad \boldsymbol{\phi}_{i_2}^T \quad \cdots \quad \boldsymbol{\phi}_{i_{N_i}}^T\,]^T\right)\\ \mathbf{C}_{\mathbf{r}_i} &= \mathrm{cov}\left([\,r_{i_1} \quad r_{i_2} \quad \cdots \quad r_{i_{N_i}}\,]^T\right)\end{aligned}$$

with $\otimes$, $\circ$ and $\mathbf{I}_l$ denote the Kronecker matrix product, Khatri–Rao matrix product and the $l \times l$ identity matrix, respectively. Since the computation of $\mathbf{W}_{\boldsymbol{\psi}_i}$ requires $\boldsymbol{\phi}_i$, an initial estimate is obtained by using $\mathbf{W}_{\boldsymbol{\psi}_i}^{-1} = 4\mathbf{R}_i\mathbf{C}_{\mathbf{r}_i}\mathbf{R}_i^T$. Being non-negative, the last element of $\boldsymbol{\psi}_i$, namely, $[\boldsymbol{\psi}_i]_3$ is replaced by $|[\boldsymbol{\psi}_i]_3|$. After obtaining $\hat{\boldsymbol{\psi}}_i$, the constraint $[\boldsymbol{\psi}_i]_1^2 + [\boldsymbol{\psi}_i]_2^2 = [\boldsymbol{\psi}_i]_3$ is imposed to obtain a more accurate position estimate. The constraint is enforced by constructing

$$\mathbf{G}\boldsymbol{\beta}_i \approx \mathbf{h}_i \quad (8)$$

where

$$\begin{aligned}\mathbf{G} &= \begin{bmatrix}1 & 0 & 1\\ 0 & 1 & 1\end{bmatrix}^T\\ \mathbf{h}_i &= [\,[\hat{\boldsymbol{\psi}}_i]_1^2 \quad [\hat{\boldsymbol{\psi}}_i]_2^2 \quad [\hat{\boldsymbol{\psi}}_i]_3\,]^T\\ \boldsymbol{\beta}_i &= [\,x_i^2 \quad y_i^2\,]^T.\end{aligned}$$

The second step WLS estimate of $\boldsymbol{\beta}_i$ is given by

$$\hat{\boldsymbol{\beta}}_i = \left(\mathbf{G}^T\mathbf{W}_{\boldsymbol{\beta}_i}\mathbf{G}\right)^{-1}\mathbf{G}^T\mathbf{W}_{\boldsymbol{\beta}_i}\mathbf{h}_i \quad (9)$$

where

$$\begin{aligned}\mathbf{W}_{\boldsymbol{\beta}_i}^{-1} &= \mathbb{E}\{(\mathbf{G}\boldsymbol{\beta}_i - \mathbf{h}_i)(\mathbf{G}\boldsymbol{\beta}_i - \mathbf{h}_i)^T\}\\ &= \mathbb{E}\{(\mathbf{G}\boldsymbol{\beta}_i - \tilde{\mathbf{h}}_i - \boldsymbol{\Delta}\mathbf{h}_i)(\mathbf{G}\boldsymbol{\beta}_i - \tilde{\mathbf{h}}_i - \boldsymbol{\Delta}\mathbf{h}_i)^T\}\\ &= \mathbb{E}\{\boldsymbol{\Delta}\mathbf{h}_i(\boldsymbol{\Delta}\mathbf{h}_i)^T\}\end{aligned}$$

where $\tilde{\mathbf{h}}_i$ is the noise-free component of $\mathbf{h}_i$ and $\mathbf{\Delta h}_i = [2\tilde{x}_i\Delta x_i + \Delta x_i^2 \; 2\tilde{y}_i\Delta y_i + \Delta y_i^2 \; \Delta[\boldsymbol{\psi}_i]_3]^T$. Excluding the terms $\Delta x_i^2, \Delta y_i^2$ and substituting $\tilde{x}_i$ and $\tilde{y}_i$ by x_i and y_i, respectively, $\mathbf{W}_{\boldsymbol{\beta}_i}^{-1}$ can be approximated as

$$\begin{aligned}\mathbf{W}_{\boldsymbol{\beta}_i}^{-1} &\approx \mathbb{E}\left\{\begin{bmatrix}2x_i\Delta x_i\\ 2y_i\Delta y_i\\ \Delta[\boldsymbol{\psi}_i]_3\end{bmatrix}\begin{bmatrix}2x_i\Delta x_i\\ 2y_i\Delta y_i\\ \Delta[\boldsymbol{\psi}_i]_3\end{bmatrix}^T\right\}\\ &= \mathrm{diag}\,(2x_i, 2y_i, 1)\;\mathrm{cov}\;(\boldsymbol{\psi}_i)\;\mathrm{diag}(2x_i, 2y_i, 1).\end{aligned}$$

The WLS estimate of $\boldsymbol{\phi}_i$ is given by

$$\hat{\boldsymbol{\phi}}_i = \mathrm{diag}\,\left(\mathrm{sgn}\,\left(\begin{bmatrix}[\boldsymbol{\psi}_i]_1\\ [\boldsymbol{\psi}_i]_2\end{bmatrix}\right)\right)\sqrt{\hat{\boldsymbol{\beta}}_i} \tag{10}$$

where $\mathrm{sgn}(l)$ is the sign of l. Since the estimated and actual positions are very close if the SNR is sufficiently high, the covariance can be approximated by substituting the position estimates and the corresponding covariance to the CRLB with anchor positions uncertainty (see the Appendix).

B. Position Estimation With Less Than Three Neighbors

In some scenarios, sensors may only have connections with one to two anchors or sensors with estimated positions. The basic algorithm which needs the positions of at least three anchors initially cannot be applied. Consider the case where the ith sensor connects to only two anchors or sensors with estimated positions. Thus, (4) becomes

$$\mathbf{A}_i\psi_i \approx \mathbf{b}_i \tag{11}$$

where

$$\begin{aligned}\mathbf{A}_i &= \begin{bmatrix}x_{i_1} & y_{i_1} & 1\\ x_{i_2} & y_{i_2} & 1\end{bmatrix}\\ \mathbf{b}_i &= [\,r_{i_1}^2 - x_{i_1}^2 - y_{i_1}^2 \quad r_{i_2}^2 - x_{i_2}^2 - y_{i_2}^2\,]^T\\ \boldsymbol{\psi}_i &= [\,\boldsymbol{\phi}_i^T \quad \boldsymbol{\phi}_i^T\boldsymbol{\phi}_i\,]^T.\end{aligned}$$

Instead of solving this underdetermined linear system, a simple geometric technique is employed to obtain the position estimates. Given two circles with centers (x_1, y_1) and (x_2, y_2) and corresponding radii r_1 and r_2, the intersection point (x, y) is calculated from

$$x = \frac{-\gamma \pm \sqrt{\gamma^2 - 4\alpha\eta}}{2\alpha} \tag{12}$$

where $\alpha = t_2^2 + t_3^2, \gamma = 2(y_1t_2t_3 - t_3^2x_1 - t_1t_2), \eta = t_3^2x_1^2 + t_1^2 - 2y_1t_1t_3 + y_1^2t_3^2 - r_1^2t_3^2, t_1 = r_2^2 - r_1^2 + x_1^2 + y_1^2 - x_2^2 - y_2^2, t_2 = 2(x_1 - x_2), t_3 = 2(y_1 - y_2)$. Similarly, y is computed using (12) by substituting x_1, x_2, y_1 and y_2 by $y_1, y_2, x_1,$ and x_2, respectively. If the two circles have no intersections, (12) will produce complex numbers. In this situation, the line passes through the two centers is first found. There are four intersection points of this line with the two circles. The midpoint of the two nearest intersection points is then chosen as the position estimate.

If the ith sensor only connects to one sensor with available position, we assume that its position lies on the perimeter of the circle with center $\boldsymbol{\phi}_{i_1}$ and diameter r_{i,i_1}. Then, we can take a set of points from the perimeter as position estimates and compute the corresponding covariance by substituting all required information to the CRLB. Note that in this case, pseudo-inverse will be used to calculate the Fisher information matrix (FIM) instead of the standard inverse since the FIM is singular.

After getting the position estimates, sensors obtain the covariances by substituting the position estimates, positions of neighbors as well as the distance and position covariances to the CRLB. Then, they transmit all position estimates and corresponding covariances to neighbors. For the ith sensor which receives several position estimates and covariances, it computes its positions by substituting all combinations of neighbors' position estimates as well as corresponding covariances into (10). After that, it calculates the error, $e_{i^{(l)}}$, for each of its own position estimate, $i^{(l)}$ and the position estimate with the smallest e_i will be the final solution.

C. Incorporation of Boundary and Communication Range Constraints

By the algorithm in Section B, we can deal with scenarios when sensors have less than three connections of sensors with available positions. However, utilizing all combinations of neighboring positions to calculate locations is computationally demanding, especially the sensors are sparsely deployed. In this situation, inclusion of boundary and communication range constraints can reduce the computations and give better estimates.

In some scenarios, the borders of the plane on which the sensors lie are known. For instance, sensors in indoor applications are confined by a room. Each sensor removes the position estimates that are outside the known boundary to reduce the computational complexity.

Besides, the communication range of sensors can be known *a priori*. Sensors can transmit their neighbors' identity numbers and their own position estimates as well as covariances to neighbors. On receiving the information, every sensor calculates the pairwise distances among its neighbors to check whether any two of its neighbors are connected. If this result contradicts with the information sent from the two neighbors, then all the connection combinations involving these two sensors will be deleted. In this way, the computational complexity will be reduced by calculating just the distances instead of the performing the positioning estimation algorithm. In practice, tolerance, which depends on the distance measurements and position uncertainty, is needed because the information received from neighbors contains errors. The algorithm is summarized as follows.

1) Anchors broadcast their positions to neighbors.
2) Sensors connecting to anchors estimate their positions and delete those outside the boundary and calculate the corresponding covariances.
3) Sensors with estimated locations become anchors with position uncertainty and they broadcast their locations, covariances as well as the neighbors' identity numbers.
4) After receiving the neighbors' information, sensors calculates the mutual distances among neighbors and delete those contradictory combinations of neighbors' positions with the use of the identity numbers of neighbors and communication range.
5) Sensors then estimate their positions and delete those outside the boundary and broadcast the one with the smallest error, together with the corresponding covariance.
6) By successive propagation and update of positions using steps 3) to 5), each sensor gets its final position estimate and covariance.

IV. Numerical Examples

Computer simulation has been conducted to evaluate the performance of the proposed algorithm using a PC with 2-GHz Pentium dual core CPU and 1-GB RAM. Given the distance measurement model of (2), the proposed method is compared with CRLB and SDR algorithm with regularization [12]. The boundary and communication range constraints are incorporated in the proposed and SDR schemes where they appear as inequality constraints in the SDR program. There is no gradient-descent based refinement step [12] in the latter which indicates that it is a purely semi-definite programming. Note that the unconstrained CRLB is not affected by these inequality constraints [20], [21].

Before proceeding to the numerical results, we follow [18], [22] to analyze the communication and computational complexity of the proposed and SDR algorithms. Assuming that packet-based multi-hop communication model is adopted, the total energy consumed is

$$\theta \times \zeta \times \tau \tag{13}$$

where θ, ζ and τ are the number of transmitted packets, average number of hops for communication and average transmission energy of one packet over one hop, respectively. For the centralized SDR algorithm, $\theta = \kappa\epsilon(M-k), \zeta = O((M-k)^{1/d})$, where κ is the ratio of packet size to measurement size, ϵ denotes the average node degree and $d=2$ and $d=3$ for 2-D and 3-D spaces, respectively. For the proposed distributed algorithm, $\theta = \kappa\epsilon(M-k)T$, where T is the number of iterations and $\zeta = 1$ because sensors only communicate with their one-hop neighbors. Furthermore, the average hop distance is the same and hence τ for both approaches as multihop communication model is assumed. As a result, the energy dissipation ratio of the SDR program to distributed algorithm is $O((M-k)^{1/d}/T)$. It means that when T is fixed, the required communication energy of the centralized algorithm is more than that of the distributed algorithm by an order of $(M-k)^{(1/d)}$.

In the proposed algorithm, each sensor obtains its WLS position estimate using (6). The average size of $\mathbf{A}_i$ is $\epsilon \times d$, which means that the computational complexity of all unknown-position sensors with T iterations equals $O(\epsilon^3(M-k)T)$. Comparing with that of the SDR scheme, namely, at least $O((M-k)^3)$ [23], the proposed algorithm is more computationally attractive particularly when the sensor connectivity of the WSN is low.

In the following, the mean square position errors (MSPEs), defined as

$$\frac{\sum_{i=1}^{\varrho}\sum_{j=k+1}^{M}\left\{\left(\hat{x}_j^{(i)}-\tilde{x}_j\right)^2+\left(\hat{y}_j^{(i)}-\tilde{y}_j\right)^2\right\}}{(M-k)\varrho},$$

are plotted in all scenarios where ϱ denotes number of runs. The average CPU time which is the mean of the time spent in all the independent runs and noise power range is used to measure the computational complexity. In our simulations, there are $M = 54$ sensors of which $k = 4$ are anchors. The proposed algorithm is iterated for 30 times to ensure convergence for all noise power ranges. All results are averages of $\varrho = 100$ independent runs and we assume that the WSN is localizable [24] in each run.

In the first scenario, the sensors are located on a 100 m × 100 m plane enclosed by (0, 0) m, (0, 100) m, (100, 0) m and (100, 100) m. The anchor positions are (100/3, 100/3) m, (100/3, 200/3) m, (200/3, 100/3) m, and (200/3, 200/3) m. The maximum communication range is set to 50 m. The unknown-position sensors have random positions within the $10^4\,\mathrm{m}^2$ area while the anchor positions are fixed in each independent run. As the sensor geometry is different in each trial, the corresponding CRLB is not unique even when the noise condition is kept identical. As a result, we take the average CRLB value based on all runs as the performance benchmark and denote it as mean CRLB. The MSPEs for the 50 unknown-position sensors are plotted in Fig. 1 to show the overall performance. It is observed that the proposed approach has smaller MSPEs than those of the SDR algorithm. The MSPEs of the proposed algorithms deviate from the mean CRLB by about 0.5 dB when $\sigma^2 \leq 0$ dB. The proposed method outperforms the SDR approach and is able to attain the mean CRLB for sufficiently small noise power. The CPU times of the proposed and SDR methods are 49 s and 176 s, respectively, which agrees with our computational complexity analysis. Note that the overall computational time of the 50 unknown-position sensors is counted in the proposed algorithm, but the localization of each sensor is performed simultaneously.

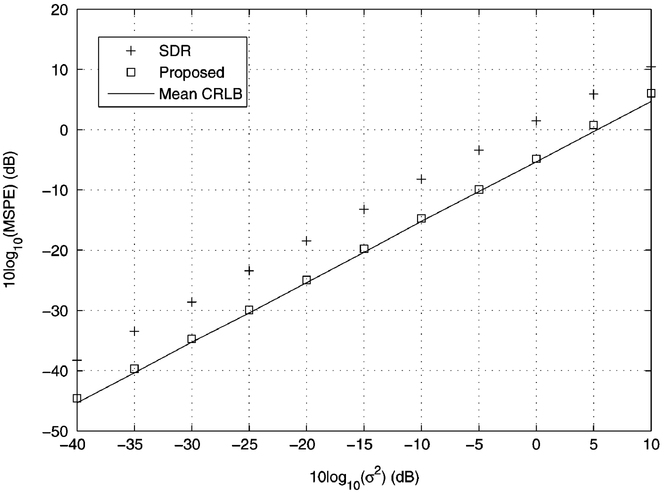

Fig. 1. Mean square position error versus σ^2 in 2-D geometry.

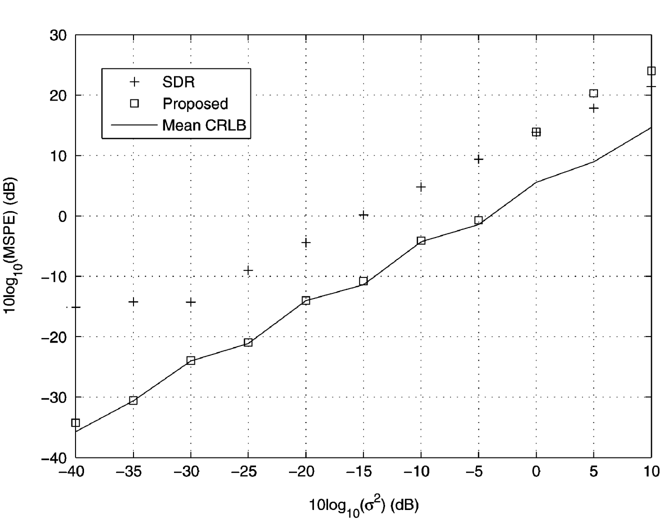

Fig. 2. Mean square position error versus σ^2 in 3-D geometry.

In the second scenario, the sensors are located on a 100 m × 100 m × 100 m space enclosed by (0, 0, 0) m, (0, 0, 100) m, (100, 0, 0) m, (100, 0, 100) m, (0, 100, 0) m, (0, 100, 100) m, (100, 100, 0) m and (100, 100, 100) m and the anchor positions are (100/3, 100/3, 100/3) m, (200/3, 200/3, 100/3) m, (200/3, 100/3, 200/3) m and (100/3, 200/3, 200/3) m. Similar to the previous simulation, the maximum communication range is 50 m and the unknown-position sensors have random positions within the 10^6 m^3 space while the anchor positions are fixed in each independent runs. Fig. 2 shows that the proposed approach has a threshold noise power of −5 dB and the MSPEs of the SDR algorithm deviate the mean CRLB by at least 6 dB. It shows the superiority of the proposed method over the SDR approach in 3-D WSN localization. The proposed method spends 119 s and the SDR approach needs 380 s.

In the third scenario, lower-connectivity WSNs are considered. The simulation setting is the same as the first one except the communication range is set to 30 m. The average node degree is now 11.7 which is 22%

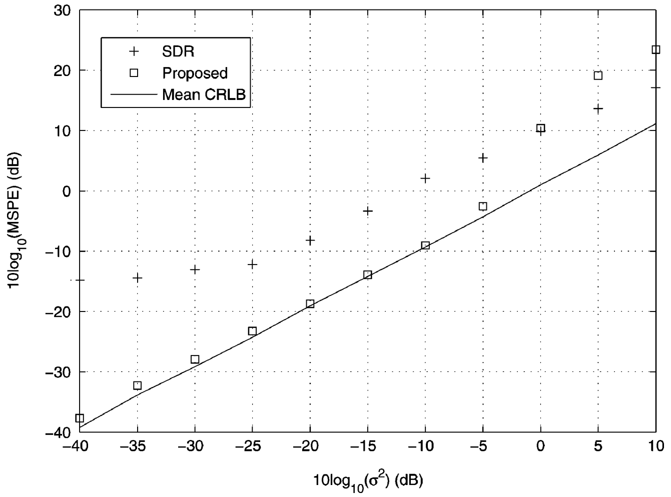

Fig. 3. Mean square position error versus σ^2 with lower connectivity.

of a fully connected sensor network. Fig. 3 shows that the threshold noise power of the proposed method is -10 dB and the SDR approach deviates from the mean CRLB by at least 5 dB. This means that the proposed method outperforms the SDR approach in low-connectivity WSNs in terms of estimation accuracy. The CPU times of the proposed and SDR approaches are 27 and 126 s, respectively, indicating the computational superiority of the former in low-connectivity WSNs.

V. Concluding Remarks

A distributed positioning approach is developed based on weighted least squares. Based on limited numerical experiments, its performance can attain Cramér–Rao lower bound when the noise power is sufficiently small and is superior to the semi-definite relaxation estimation method.

Appendix

In this Appendix, the Cramér–Rao lower bound (CRLB) for the position estimates with anchor position uncertainty is derived. The development is an extension of [25] and [26] where the former considers CRLB for WSN localization without anchor uncertainty while the latter derives the CRLB for a single unknown position with receiver position uncertainty using TDOA and frequency-difference-of-arrival measurements. The joint likelihood function of the available distance measurements and the noise-corrupted anchor positions are given by

$$p\left(\mathbf{a}|\boldsymbol{\phi}_{i_1}^T,\ldots,\boldsymbol{\phi}_{i_{N_i}}^T,\boldsymbol{\phi}_i^T\right)=\mathcal{N}(\mathbf{a};\boldsymbol{\mu},\boldsymbol{\Sigma}) \tag{14}$$

where

$$\begin{aligned}
\mathbf{a}&=\left[\,r_{i,i_1}\quad r_{i,i_2}\quad\cdots\quad r_{i,i_{N_i}}\quad \boldsymbol{\phi}_{i_1}^T\quad \boldsymbol{\phi}_{i_2}^T\quad\cdots\quad \boldsymbol{\phi}_{i_{N_i}}^T\,\right]^T\\
\boldsymbol{\mu}&=\left[\,\|\tilde{\boldsymbol{\phi}}_i-\tilde{\boldsymbol{\phi}}_{i_1}\|\quad\cdots\quad \|\tilde{\boldsymbol{\phi}}_i-\tilde{\boldsymbol{\phi}}_{i_{N_i}}\|\quad \tilde{\boldsymbol{\phi}}_{i_1}^T\quad\cdots\quad \tilde{\boldsymbol{\phi}}_{i_{N_i}}^T\,\right]^T\\
\boldsymbol{\Sigma}&=\mathrm{blkdiag}\ \left(\boldsymbol{\Sigma}_{\mathbf{r}},\boldsymbol{\Sigma}_{\boldsymbol{\phi}}\right)\\
\boldsymbol{\Sigma}_{\mathbf{r}}&=\mathrm{diag}\left(\sigma_{i,i_1}^2,\sigma_{i,i_1}^2,\ldots,\sigma_{i,i_{N_i}}^2\right)\\
\boldsymbol{\Sigma}_{\boldsymbol{\phi}}&=\mathrm{blkdiag}\left(\boldsymbol{\Sigma}_{\boldsymbol{\phi}_{i_1}},\boldsymbol{\Sigma}_{\boldsymbol{\phi}_{i_2}},\ldots,\boldsymbol{\Sigma}_{\boldsymbol{\phi}_{i_{N_i}}}\right).
\end{aligned}$$

Here, $\tilde{\boldsymbol{\phi}}_i$ stands for the ith noise-free anchor position, $\mathcal{N}(\mathbf{a};\boldsymbol{\mu},\boldsymbol{\Sigma})$ denotes Gaussian density function with $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ represent the mean and covariance and $\mathrm{blkdiag}(\,\cdot\,)$ represents the block diagonal matrix. The CRLBs of position estimates are given by the diagonal elements of the inverse of Fisher information matrix (FIM):

$$\mathrm{FIM}=\mathbf{L}\boldsymbol{\Sigma}^{-1}\mathbf{L}^T \tag{15}$$

where $\mathbf{L}$ is the Jacobian with the following form:

$$\begin{aligned}
\mathbf{L}&=\left[\,\mathbf{L}_{\mathbf{r}}\quad \mathbf{L}_{\boldsymbol{\phi}}\,\right]\\
\mathbf{L}_{\mathbf{r}}&=\begin{bmatrix}\mathrm{blkdiag}\left(\dfrac{\tilde{\boldsymbol{\phi}}_{i_1}-\tilde{\boldsymbol{\phi}}_i}{\|\tilde{\boldsymbol{\phi}}_{i_1}-\tilde{\boldsymbol{\phi}}_i\|},\dfrac{\tilde{\boldsymbol{\phi}}_{i_2}-\tilde{\boldsymbol{\phi}}_i}{\|\tilde{\boldsymbol{\phi}}_{i_2}-\tilde{\boldsymbol{\phi}}_i\|},\ldots,\dfrac{\tilde{\boldsymbol{\phi}}_{i_{N_i}}-\tilde{\boldsymbol{\phi}}_i}{\|\tilde{\boldsymbol{\phi}}_{i_{N_i}}-\tilde{\boldsymbol{\phi}}_i\|}\right)\\
-\dfrac{\tilde{\boldsymbol{\phi}}_{i_1}-\tilde{\boldsymbol{\phi}}_i}{\|\tilde{\boldsymbol{\phi}}_{i_1}-\tilde{\boldsymbol{\phi}}_i\|}\cdots-\dfrac{\tilde{\boldsymbol{\phi}}_{i_{N_i}}-\tilde{\boldsymbol{\phi}}_i}{\left\|\tilde{\boldsymbol{\phi}}_{i_{N_i}}-\tilde{\boldsymbol{\phi}}_i\right\|}\end{bmatrix}\\
\mathbf{L}_{\boldsymbol{\phi}}&=\left[\,\mathbf{I}_{2N_i}\quad \mathbf{0}_{2N_i\times 2}\,\right]^T.
\end{aligned}$$

The covariance of $\hat{\boldsymbol{\phi}}_i$ is given by the 2×2 block of the bottom right corner of the FIM.

References

[1] A. Hać, *Wireless Sensor Network Designs*. New York: Wiley, 2003.

[2] M. Ilyas and I. Mahgoub, *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*. London, U.K.: CRC Press, 2005.

[3] H. Karl and A. Willig, *Protocols and Architectures for Wireless Sensor Networks*. New York: Wiley, 2005.

[4] R. Poovendran, C. Wang, and S. Roy, *Secure Localization and Time Synchronization for Wireless Sensor and Ad Hoc Networks*. New York: Springer, 2007.

[5] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low cost outdoor localization OFR very small devices," *IEEE Pers. Commun. Mag.*, vol. 7, pp. 28–34, Oct. 2000.

[6] T. He, C. Huang, B. Blum, J. Stankovic, and T. Abdelzaher, "Range-free localization schemes in large scale sensor networks," in *Proc. 9th Annu. Int. Conf. Mobile Computing Networking*, San Diego, CA, Sep. 2003, pp. 81–95.

[7] R. L. Moses, D. Krishnamurthy, and R. M. Patterson, "A self-localization method for wireless sensor networks," *EURASIP J. Appl. Signal Process.*, vol. 4, pp. 348–358, Mar. 2003.

[8] N. Patwari, A. O. Hero III, M. Perkins, N. S. Correal, and R. J. O'Dea, "Relative location estimation in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 51, no. 8, pp. 2137–2148, Aug. 2003.

[9] M. Meurer, P. W. Baier, T. Weber, C. A. Jotten, and S. Heilmann, "SPIDER: Enhanced distance based localization of mobile radio terminals," in *Proc. IEEE 60th Vehicular Technology Conf.*, Los Angeles, CA, Sep. 2004, vol. 5, pp. 3521–3525.

[10] X. Ji and H. Zha, "Robust sensor localization algorithm in wireless ad-hoc sensor networks," in *Proc. 12th Int. Conf. Computer Communications Networks*, Dallas, TX, Oct. 2003, pp. 527–532.

[11] Y. Shang, W. Ruml, Y. Zhang, and M. Fromherz, "Localization from connectivity in sensor networks," *IEEE Trans. Parallel Distrib. Netw.*, vol. 15, no. 11, pp. 961–974, Nov. 2004.

[12] P. Biswas, T.-C. Liang, K.-C. Toh, Y. Ye, and T.-C. Wang, "Semidefinite programming approaches for sensor network localization with noisy distance measurements," *IEEE Trans. Autom. Sci. Eng.*, vol. 3, no. 4, pp. 360–371, Oct. 2006.

[13] 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.*, pp. 54–69, Jul. 2005.

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

[15] J. J. Caffery, Jr., *Wireless Location in CDMA Cellular Radio Systems*. Boston, MA: Kluwer, 2000.

[16] C. Savarese, J. M. Rabaey, and J. Beutel, "Location in distributed ad-hoc wireless sensor network," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, Salt Lake City, UT, May 2001, vol. 4, pp. 2037–2040.

[17] A. Savvides, C. C. Han, and M. B. Srivastava, "Dynamic fine-grained localization in ad hoc networks of sensors," in *Proc. 7th ACM Int. Conf. Mobile Computing Networking*, Rome, Italy, Jul. 2001, pp. 166–179.

[18] J. A. Costa, N. Patwari, and A. O. Hero III, "Distributed weighted-multidimensional scaling for node localization in sensor networks," *ACM Trans. Sensor Netw.*, vol. 2, no. 1, pp. 39–64, Feb. 2006.

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

[20] J. D. Gorman and A. O. Hero, "Lower bounds for parametric estimation with constraints," *IEEE Trans. Inf. Theory*, vol. 26, no. 6, pp. 1285–1301, Nov. 1990.

[21] T. J. Moore, B. M. Sadler, and R. J. Kozick, "Maximum likelihood estimation, the Cramer–Rao bound, and the method of scoring with parameter constraints," *IEEE Trans. Signal Process.*, vol. 56, no. 3, pp. 895–908, Mar. 2008.

[22] M. Rabbat and R. Nowak, "Distributed optimization in sensor networks," in *Proc. 3rd Int. Symp. Information Processing in Sensor Networks*, Berkeley, CA, Apr. 2004, pp. 20–27.

[23] Z. Wang, S. Zheng, Y. Ye, and S. Boyd, "Further relaxations of the SDP approach to sensor network localization," *SIAM J. Optimiz.*, vol. 19, no. 2, pp. 655–673, Jul. 2008.

[24] A. M. C. So and Y. Ye, "Theory of semidefinite programming for sensor network localization," *Math. Programm.*, vol. 109, no. 2–3, pp. 367–384, Mar. 2007.

[25] S. Gezici, T. Zhi, G. B. Giannakis, H. Kobayashi, A. R. Molisch, H. V. Poor, and Z. Sahinoglu, "Localization via ultra-wideband radios: A look at positioning aspects for future sensor networks," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 70–84, Jul. 2005.

[26] K. C. Ho, X. Lu, and L. Kovavisaruch, "Source localization using TDOA and FDOA measurements in the presence of receiver location errors: Analysis and solution," *IEEE Trans. Signal Process.*, vol. 55, no. 4, pp. 684–696, Apr. 2007.

On Error-Saturation Nonlinearities in NLMS Adaptation

Neil J. Bershad

***Abstract*—The effect of a saturation-type error nonlinearity in the weight update equation in normalized least mean-square (NLMS) adaptation is investigated for system identification for a white Gaussian data model. Nonlinear recursions are derived for the weight mean error and mean-square deviation (MSD) that include the effect of an error function (erf) saturation-type nonlinearity on the error sequence driving the algorithm. The nonlinear recursion for the MSD is solved numerically and shown in excellent agreement with Monte Carlo simulations, supporting the theoretical model assumptions. The theory is extended to tracking a Markov channel and accurately predicts the tracking behavior as well. The saturation behavior of the algorithm is easily studied by varying a single parameter in the error function, varying from a linear device to a hard limiter. For the white data case, the excess mean square-error (EMSE) is simply related to the MSD. The tradeoff between the extent of error saturation, steady-state EMSE, and algorithm convergence rate is studied using these results.**

***Index Terms*—Adaptive filters, analysis, NLMS, nonlinear systems, stochastic algorithms.**

I. Introduction

Adaptive signal processing algorithms have found wide application in situations where the statistics of the input are unknown or time-varying. The least mean-square (LMS) algorithm was most often selected in many situations because of its simplicity and robustness. The LMS algorithm often operated in a nonlinear manner because of saturation or quantization acting upon the various signals in the adaptation loop. Mathematical models were developed to study the nonlinear behavior. Reference [1] is an early example of such analysis. Subsequently, the effects of a saturation type non-linearity on LMS adaptation for Gaussian inputs and Gaussian observation noise were studied in a series of papers [2]–[4] in the middle of the 1980s. The saturation nonlinearity was modeled as an erf function where $\mathrm{erf}(x) = (2/\sqrt{\pi}) \int_0^x \exp(-t^2)dt$. This shape, coupled with Gaussian data model, simplified the stochastic analysis of the nonlinear LMS weight update recursion. The tradeoff between algorithm speed and degree of saturation was studied in detail for these Gaussian models. More recently the LMS algorithm has been replaced by the normalized least mean-square (NLMS) algorithm [5], [6] for various reasons, the most important being that the NLMS algorithm does not require knowledge of the input power level for design purposes. Stochastic analysis of the behavior of the NLMS algorithm is given in [7]–[9] for example. However, for theoretical reasons, the stochastic analysis of the effects of error nonlinearities in the NLMS weight update have not been studied in very much detail. [10] has investigated the case of nonlinear NLMS adaptation for a Huber function nonlinearity operating on the error signal for the case of additive impulse noise.

This paper studies the stochastic behavior of the NLMS algorithm with an erf function type nonlinearity operating on the error signal. The erf function can model different degrees of saturation by varying a single parameter, varying from a linear device to a hard limiter. Thus, the analysis results are in terms of this parameter. The stochastic analysis evaluates certain conditional expectations in the NLMS weight update recursion by combining 1) the approach in [7] for differentiating under the expectation sign to remove the NLMS normalization factor and 2) the approach in [4] to evaluate expectations of nonlinear Gaussian functionals using Price's theorem [11]. The approach of this paper can be applied to other nonlinearities operating on the error signal in the NLMS algorithm. In particular, this approach can be used to correct the analysis errors for the Huber type saturation nonlinearity, studied in [10].

The paper is organized as follows. Section II defines the system identification problem and states the assumptions needed for the analysis. Section III derives deterministic recursions for the mean weight error and the mean-square deviation (MSD). Section IV extends the theory to tracking a Markov channel. Section V compares the theory derived in Sections III and IV with Monte Carlo simulations for a wide variety of system parameters. This section verifies the assumptions needed for the theoretical model. Section VI contains the conclusions.

II. System Identification Problem

This paper focuses on the system identification problem shown in Fig. 1. Initially, the unknown channel is assumed time-invariant. The NLMS weight update for the weight vector $W(n)$ with the erf function nonlinearity is given by

$$W(n+1) = W(n) + \mu \frac{g\left(e(n)X(n)\right)}{X^T(n)X(n)} \tag{1}$$

where $0 < \mu \leq 1$ is the algorithm step size

$$e(n) = d(n) - X^T(n)W(n) = n_o(n) + X^T(n)\left(W_o(n) - W(n)\right) \tag{2}$$

Manuscript received May 27, 2008; accepted April 09, 2009. First published May 02, 2009; current version published September 16, 2009. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. William A. Sethares.

The author is with the Henry Samueli School of Engineering, University of California, Irvine, CA 92660 USA (e-mail: bershad@ece.uci.edu).

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

Digital Object Identifier 10.1109/TSP.2009.2021917

1053-587X/$26.00 © 2009 IEEE