ACHIEVING HIGH-ACCURACY DISTRIBUTED LOCALIZATION IN SENSOR NETWORKS

Jose A. Costa, Neal Patwari and Alfred O. Hero III

Department of Electrical Engineering and Computer Science
University of Michigan, Ann Arbor, MI 48109
Emails: jcosta@umich.edu, {npatwari, hero}@eecs.umich.edu.

ABSTRACT

Accurate, distributed localization algorithms are needed for a wide variety of wireless sensor network applications. This paper introduces a scalable, distributed weighted-multidimensional scaling (dwMDS) algorithm that adaptively emphasizes the most accurate range measurements available and naturally accounts for communication constraints within the sensor network. For received signal-strength (RSS) based range measurements, we demonstrate via simulation that location estimates are nearly unbiased with variance close to the Cramér-Rao lower bound (CRB). Further, RSS and time-of-arrival (TOA) channel measurements are used to demonstrate performance as good as the centralized maximum-likelihood estimator (MLE) in a real-world sensor network.

1. INTRODUCTION

For monitoring and control applications using wireless sensor networks, automatic localization of every sensor will be a key enabling technology. Sensor data must be registered to its physical location to permit deployment of energy-efficient routing schemes, source localization algorithms, and distributed compression techniques. Moreover, for applications such as inventory management and manufacturing logistics, localization and tracking of sensors are the primary purposes of the wireless network. For large-scale networks of inexpensive, energy-efficient devices, it is not feasible to include GPS capability on every device or to require a system administrator to manually enter all device coordinates. In this paper, we consider the location estimation problem for which only a small fraction of sensors have *a priori* coordinate knowledge, and range measurements between many pairs of neighboring sensors permit the estimation of all sensor coordinates [1].

Two major difficulties hinder accurate sensor location estimation: first, accurate range measurements are expensive; and second, centralized estimation becomes impossible as the scale of the network increases. This paper proposes a distributed localization algorithm, based on a weighted version of multidimensional scaling (MDS), in which sensors need to exchange data only within a small set of neighbors. Its key features are: (1) a weighted cost function that allows range measurements that are believed to be more accurate to be weighted more heavily; (2) an adaptive neighbor selection method that avoids the biasing effects of selecting neighbors based on noisy range measurements; (3) a majorization method which has the property that each iteration is guaranteed to improve the value of the cost function.

This research was partially supported by the National Science Foundation under ITR grant CCR-0325571.

2. PROBLEM STATEMENT

Consider a network of $N = n + m$ devices, living in a D dimensional space ($D = 2$ or 3). Let $\{\boldsymbol{x}_i\}_{i=1}^N$, $\boldsymbol{x}_i \in \mathbb{R}^D$, be the actual vector coordinates of sensors, or, equivalently, define the matrix of coordinates $X = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]$. The last m sensors ($i = n+1, \ldots, N$), called *anchor nodes*, have perfect *a priori* knowledge of their coordinates. The first n sensors ($i = 1, \ldots, n$) have either no knowledge or some imperfect *a priori* coordinate knowledge and are called *unknown-location nodes*. Imperfect *a priori* knowledge about sensor $i \leq n$ is encoded by parameters r_i and $\overline{\boldsymbol{x}}_i$, where, with accuracy r_i, $\boldsymbol{x}_i$ is believed to lie around $\overline{\boldsymbol{x}}_i$. If no such knowledge is available, $r_i = 0$. Summarizing, three distinct sets of sensors can be considered in this formulation based on their *a priori* information: perfect ($i > n$), imperfect ($i \leq n, r_i > 0$), or zero coordinate knowledge ($i \leq n, r_i = 0$). Note that one or two of these sets might be empty, e.g., no anchor nodes available and/or no prior information on sensors locations.

The localization problem considered in this paper consists in the estimation of the coordinates $\{\boldsymbol{x}_i\}_{i=1}^n$ given the coordinates of the anchor nodes, $\{\boldsymbol{x}_i\}_{i=n+1}^N$, imperfect *a priori* knowledge, $\{(r_i, \overline{\boldsymbol{x}}_i)\}_{i=1}^n$ and many pairwise range measurements, $\{\delta_{ij}^{(t)}\}$, taken over time $t = 1 \ldots K$. The available range measurements indexes (i, j) are in some subset of $\{1, \ldots, N\}^2$. We assume that this subset of measurements results in a connected network; otherwise, each connected subset should be considered individually.

3. DISTRIBUTED WEIGHTED MULTIDIMENSIONAL SCALING

3.1. The dwMDS Cost Function

We define MDS as the solution to the minimization of the following global cost function (a.k.a. STRESS function [2]):

$$S = 2 \sum_{1 \leq i \leq n} \sum_{i < j \leq n+m} \sum_{1 \leq t \leq K} w_{ij}^{(t)} \left(\delta_{ij}^{(t)} - d_{ij}(X) \right)^2 + \sum_{1 \leq i \leq n} r_i \left\| \boldsymbol{x}_i - \overline{\boldsymbol{x}}_i \right\|^2, \quad (1)$$

where $d_{ij}(X) = \|\boldsymbol{x}_i - \boldsymbol{x}_j\| = \sqrt{(\boldsymbol{x}_i - \boldsymbol{x}_j)^T (\boldsymbol{x}_i - \boldsymbol{x}_j)}$ is the Euclidean distance, and we assume that for each pair (i, j), up to K dissimilarity measurements are available. The weight $w_{ij}^{(t)}$ ($t = 1, \ldots, K$) can be selected to quantify the predicted accuracy of measurement $\delta_{ij}^{(t)}$. If no such measurement is available between i and j, or if i and j do not consider themselves neighbors (see

subection 3.4), then $w_{ij}^{(t)} = 0$. We assume that $w_{ij}^{(t)} \geq 0$, $w_{ii}^{(t)} = 0$ and $w_{ij}^{(t)} = w_{ji}^{(t)}$, i.e., the weights are symmetric. Note that (1) differs from the standard MDS cost function in that we have added a penalty term to account for prior knowledge about node locations.

After simple manipulations, S can be rewritten as follows:

$$S = \sum_{i=1}^{n} S_i + c \,, \tag{2}$$

where functions S_i are defined for each unknown-location node (i.e. $1 \leq i \leq n$),

$$\begin{aligned} S_i = & \sum_{\substack{j=1 \\ j \neq i}}^{n} \overline{w}_{ij} \left(\overline{\delta}_{ij} - d_{ij}(X) \right)^2 \\ & + \sum_{j=n+1}^{n+m} 2\, \overline{w}_{ij} \left(\overline{\delta}_{ij} - d_{ij}(X) \right)^2 + r_i \| \boldsymbol{x}_i - \overline{\boldsymbol{x}}_i \|^2 \,, \end{aligned} \tag{3}$$

and c is a constant independent of the nodes locations X. In (3), the K weights and range measurements between i and j are summarized by a single weight $\overline{w}_{ij} = \sum_{t=1}^{K} w_{ij}^{(t)}$ and measurement $\overline{\delta}_{ij} = \sum_{t=1}^{K} w_{ij}^{(t)} \delta_{ij}^{(t)} / \overline{w}_{ij}$. As S_i only depends on the measurements available at node i and the positions of neighboring nodes, i.e., nodes for which $w_{ij}^{(t)} > 0$ (for some t), it can be viewed as the local cost function at node i. This is the key enabling factor for implementing a distributed localization procedure.

3.2. Minimizing the dwMDS Cost Function

Motivated by the additive structure of the global cost (2), we propose an iterative distributed algorithm in which each sensor updates its position estimate by minimizing the corresponding local cost function S_i, after taking measurements and receiving position estimates from its neighboring nodes.

However, unlike classical MDS, no closed form expression exists for the minimum of the cost function S_i (or S). We address this problem by minimizing $S_i = S_i(\boldsymbol{x}_i)$ iteratively using quadratic majorizing functions as in SMACOF (Scaling by MAjorizing a COmplicated Function [3]). This method has the attractive property of generating a sequence of non-increasing STRESS values. Due to space limitations, we omit the derivation of the majorization function and its minimization and present only the final update equations for the nodes positions. The corresponding details can be found in [4]. If $X^{(k)}$ is the matrix whose columns contain the position estimates for all points at iteration k, quadratic majorization of S_i results in the following update equation for the position estimate of node i:

$$\boldsymbol{x}_i^{(k+1)} = a_i \left(r_i \, \overline{\boldsymbol{x}}_i + X^{(k)} \boldsymbol{b}_i^{(k)} \right) \,, \tag{4}$$

where

$$a_i^{-1} = \sum_{\substack{j=1 \\ j \neq i}}^{n} \overline{w}_{ij} + \sum_{j=n+1}^{n+m} 2\, \overline{w}_{ij} + r_i \,, \tag{5}$$

and $\boldsymbol{b}_i^{(k)} = [b_1, \ldots, b_{n+m}]^T$ is a vector whose entries are given by:

$$\begin{array}{ll} b_j = \overline{w}_{ij} \left[1 - \overline{\delta}_{ij} / d_{ij}(X^{(k)}) \right] & j \leq n \,,\ j \neq i \\ b_i = \sum_{\substack{j=1 \\ j \neq i}}^{n} \overline{w}_{ij} \, \overline{\delta}_{ij} / d_{ij}(X^{(k)}) + \sum_{j=n+1}^{n+m} 2\, \overline{w}_{ij} \, \overline{\delta}_{ij} / d_{ij}(X^{(k)}) & \\ b_j = 2\, \overline{w}_{ij} \left[1 - \overline{\delta}_{ij} / d_{ij}(X^{(k)}) \right] & j > n \end{array} \,.$$

Inputs: $\{\delta_{ij}^{(t)}\}$, $\{w_{ij}^{(t)}\}$, m, $\{r_i\}$, $\{\overline{\boldsymbol{x}}_i\}$, ϵ, initial condition $X^{(0)}$;
Initialize: $k = 0$, $S^{(0)}$, compute a_i from equation (5);
repeat
  $k \leftarrow k + 1$;
  for $i = 1$ **to** n
    compute $\boldsymbol{b}_i^{(k-1)}$ from equation (3.2);
    $\boldsymbol{x}_i^{(k)} = a_i \left(r_i \, \overline{\boldsymbol{x}}_i + X^{(k-1)} \boldsymbol{b}_i^{(k-1)} \right)$;
    compute $S_i^{(k)}$;
    $S^{(k)} \leftarrow S^{(k)} - S_i^{(k-1)} + S_i^{(k)}$;
    communicate $\boldsymbol{x}_i^{(k)}$ to neighbors of node i
    (i.e., nodes for which $w_{ij} > 0$);
    communicate $S^{(k)}$ to node $i + 1 \pmod n$;
  end for
until $S^{(k-1)} - S^{(k)} < \epsilon$

Fig. 1. dwMDS algorithm.

As the weights $w_{ij}^{(t)}$ can be made zero except for a relative neighborhood of node i, only the corresponding entries of vector $\boldsymbol{b}$ will be nonzero, and the update rule for $\boldsymbol{x}_i$ will depend only on this neighborhood (as opposed to the whole matrix $X^{(k)}$).

3.3. Algorithm

The proposed algorithm is summarized in Figure 1. We make the following comments:

1. The choice of weighting function w_{ij} should reflect the accuracy of the range measurements, such that less accurate measurements are down-weighted in the overall cost function. Adopting a model-independent adaptive weighting scheme, we propose the following weight assignment:

$$w_{ij} = \begin{cases} \exp\left\{ -\delta_{ij}^2 / h_i^2 \right\} & \text{, if } \delta_{ij} \text{ is measured} \\ 0 & \text{, otherwise} \end{cases} \,, \tag{6}$$

   where $h_i = \max_j \delta_{ij}$. This choice of w_{ij}, which equalizes the (nonzero) weight distribution in all sensors, has robust performance as shown in the experiments reported in Section 4.

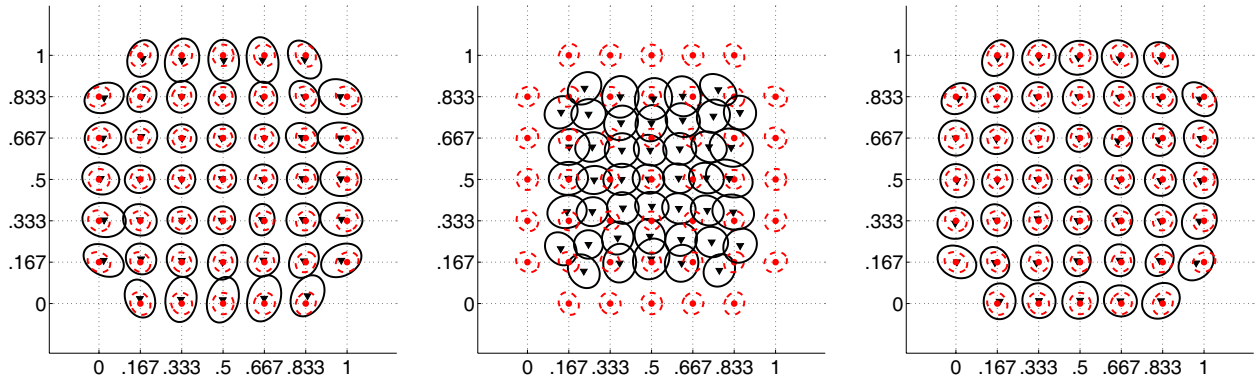
2. The question of how to adaptively choose the neighbors of each node (i.e., which weights are made positive) in order to decrease communication costs or improve localization performance is addressed in Section 3.4.

3. Regarding the initialization of the algorithm, every node requires an initial estimate of its position. This can be done using, for example, the algorithms proposed in [5] or [6], which compute a "rough" initialization of the position estimates in a distributed fashion.

4. In the description of the algorithm, it was assumed for notational convenience, that the algorithm cycles through the network in an ordered fashion (i.e., messages are passed between nodes in the order $1, 2, \ldots, n$). However, many other non-cyclic update rules are possible, e.g., parallel updates or inner iterations within local clusters of sensors.

3.4. Adaptive Neighborhood Selection

Most methods, in order to limit communication costs and computational complexity, select neighbors by choosing those devices



(a) neighborhood selection using actual distances (b) neighborhood selection using measured ranges (c) adaptive neighborhood selection

Fig. 2. Estimator mean (▾) and 1-σ uncertainty ellipse (—) for each blindfolded sensor compared to the true location ($\cdot$) and CRB on the 1-σ uncertainty ellipse (- - -).

which are closer than a threshold distance d_R. However, when ranges are measured with noise, this selection process will tend to choose devices whose measured distances are shorter than the true distances, creating a negative bias phenomenon. Motivated by this phenomenon, we propose a two-stage neighborhood selection process, based on the predicted distances between sensors.

In the first stage, the dwMDS algorithm from Fig. 1 is run with a neighborhood structure based on the available range measurements, i.e., set $w_{ij} = 0$ if $\delta_{ij} > d_R$. After convergence, this stage provides an interim estimate $\{\hat{\boldsymbol{x}}_i\}$ of the sensors locations, that, with high probability, will have negatively biased predicted distances.

In the second stage, these interim estimated sensor locations are used to compute a new neighborhood structure, by assigning $w_{ij} = 0$ if $\|\hat{\boldsymbol{x}}_i - \hat{\boldsymbol{x}}_j\| > d_R$. Some neighbors with low range measurements will be dropped, while others with possibly longer range measurements will be added. Then, using $\{\hat{\boldsymbol{x}}_i\}$ as an initial condition and the new neighborhood structure, the dwMDS algorithm is re-run, resulting in the final location estimates. Note that the predicted distances $\|\hat{\boldsymbol{x}}_i - \hat{\boldsymbol{x}}_j\|$ are used only to select neighbors (i.e., which weights are positive) – the measured ranges δ_{ij} are still used to determine the weight values.

We remark that this 2-stage algorithm does not imply twice the computation. The dwMDS algorithm is based on majorization, and each iteration brings it closer to convergence. Since the first stage only needs to provide coarse localization information, the dwMDS algorithm can be stopped quickly with a large ϵ. Next, the second stage begins with very good (although biased) coordinate estimates, so the second run of the dwMDS algorithm will likely require fewer iterations to converge.

4. EXPERIMENTAL RESULTS

4.1. Simulations

In this subsection, all the simulated data were generated according to the log-normal model for RSS range measurements (see [7] for details). If the received power in mW at sensor i transmitted by sensor j, $P_{ij}(\text{mW})$, is log-normal, then received power in decibels, $P_{ij} = 10\log_{10} P_{ij}(\text{mW})$, is Gaussian. Typically P_{ij} is modeled as

$$\begin{aligned} P_{ij} &\sim \mathcal{N}(\bar{P}_{ij}, \sigma_{dB}^2) \\ \bar{P}_{ij} &= P_0 - 10 n_p \log_{10}(d_{ij}/d_0)\,, \end{aligned}$$

where $\bar{P}_{ij}$ is the mean power in decibel milliwatts at distance d_{ij}, σ_{dB}^2 is the variance of the shadowing and $P_0(\text{dBm})$ is the received power at a reference distance d_0. Typically $d_0 = 1\,\text{m}$, and P_0 is calculated from the free space path loss formula. The path loss exponent n_p is a parameter determined by the environment. This leads to the following expression for the range measurements:

$$\delta_{ij} = d_0 10^{(P_0 - P_{ij})/(10 n_p)}. \tag{7}$$

In all the simulations presented in this subsection, $\sigma_{dB}/n_p = 1.7$.

We first demonstrate the performance of the proposed algorithms on a network of 7×7 sensors arranged on a uniform grid of unit area, in which the 4 corner devices are anchor nodes and the remaining 45 are unknown location devices. For all experiments on this configuration, we use $d_R = 0.4\,\text{m}$ (yielding an average of 14 neighbors per device). We ran 200 Monte Carlo simulation trials to determine confidence ellipses, root-mean-square error (RMSE) and bias performance (per sensor) of the location estimates. The results are displayed in Figure 2, where we plot the mean and 1-σ uncertainty ellipse of the estimator, and compare it to the actual device location and the CRB lower bound on the uncertainty ellipses [7]. We remark that the CRB shown is calculated assuming full connectivity (all devices measure range to all other devices), and as such provides only a loose lower bound on the best performance achievable by any unbiased estimator.

In the first experiment, we provide a baseline best-case scenario by using noise-free distance measurements to select neighborhoods, i.e., we have an oracle to tell us when the true distance between i and j is less than a threshold, $\|\boldsymbol{x}_i - \boldsymbol{x}_j\| < d_R$ (note that measurements δ_{ij} are still noisy – the oracle only tells us how to select neighbors). The simulation shown in Figure 2(a) has a RMSE of the location estimates of 0.090 m and a bias of 0.019 m.

For the second experiment, the (noisy) RSS measurements are used to select neighbors, i.e., devices i and j are neighbors if $\delta_{ij} \leq$

Table 1. RMSE of location estimates in experimental network

	Classical MDS	MLE [7]	dwMDS
RSS	4.30 m	2.18 m	2.48 m
TOA	1.96 m	1.23 m	1.12 m

d_R. This corresponds to applying only the first stage of the neighborhood selection method from subsection 3.4. The results are shown in Figure 2(b). The estimates are strongly pulled towards the center, due to the negative bias of the range estimates which are 'selected' by the connectivity condition. Now, the RMSE is 0.162 m and the bias is 0.130 m.

A third experiment uses the adaptive neighborhood selection method proposed in Section 3.4. The results are displayed in Figure 2(c), where it can be seen that this method succeeds in removing the negative bias effect. The bias has gone back down to 0.012 m, while the RMSE is 0.092m, just slightly higher than the baseline experiment using the oracle.

4.2. Localization in an Experimental Sensor Network

To test the performance of the proposed algorithm on real-world channel measurements, we used the RSS and TOA measurements presented in [7]. This data set includes the RSS and TOA range measurements from a network of 44 devices (4 of which are anchor nodes) using a wideband direct-sequence spread-spectrum transmitter and receiver pair operating at a center frequency of 2.4 GHz. The measurements were made in an open plan office building, within a 14 m square area.

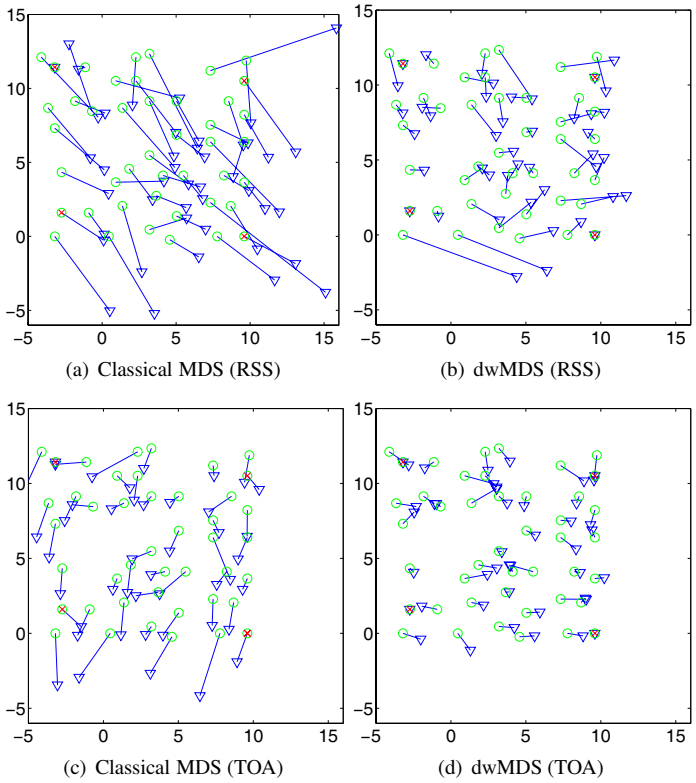
We compare the performance of the dwMDS algorithm with adaptive neighborhood selection to classical MDS and the MLE based solutions from [7]. Table 1 summarizes the RMSE of the location estimates.

Figures 3(a) and 3(b) show the location estimates using classical MDS (which used all the pairwise range measurements between sensors) and the dwMDS algorithm, for the RSS measurement data set. The true and estimated sensor positions are marked by 'o' and '▽', respectively, where the lines represent the estimation errors. It can be observed that the dwMDS algorithm does much better than classical MDS. On the other hand, the RMSE of the dwMDS algorithm is slightly higher than the RMSE of the centralized MLE reported in [7]. However, that method, unlike the dwMDS, not only uses all pairwise range measurements, but also relies on previously estimating the channel parameters. If we allow d_R to increase at the expense of increasing communication costs, the dwMDS algorithm can reach an RMSE as low as 2.27 m for $d_R = 8.5$ m.

Figure 3(c) and 3(d) show the same location estimates for the TOA measurement data set. From Table 1, it can be seen that the dwMDS algorithm outperforms all other location estimators.

5. CONCLUSION

The proposed distributed weighted-MDS algorithm is well-suited for sensor localization, because it is both nonparametric, thus applicable to a variety of distance measurements (eg. TOA, RSS) without the need to manually adjust parameters; and communication-constrained such that sensors need information only from a small set of neighbors. This paper has demonstrated the accuracy of dwMDS via simulation and experimental measurements.



(a) Classical MDS (RSS) (b) dwMDS (RSS)

(c) Classical MDS (TOA) (d) dwMDS (TOA)

Fig. 3. Experimental sensor network location estimates from RSS and TOA measurements via MDS method and dwMDS method (using adaptive neighborhood selection w/ $d_R = 6$ m). Anchor nodes 'x', true sensor locations 'o' and location estimates '▽' are shown.

6. REFERENCES

[1] N. Patwari, J. Ash, S. Kyperountas, R. L. Moses, A. O. Hero III, and N. S. Correal, "It takes a network: Cooperative geolocation of wireless sensors," *subimtted to IEEE Signal Proc. Mag., Special Issue on Positioning and Navigation*, 2005.

[2] T. Cox and M. Cox, *Multidimensional Scaling*, Chapman & Hall, London, 1994.

[3] P. Groenen, *The majorization approach to multidimensional scaling: some problems and extensions*, DSWO Press, 1993.

[4] J. A. Costa, , N. Patwari, and A. O. Hero III, "Distributed multidimensional scaling with adaptive weighting for node localization in sensor networks," *submitted to ACM Trans. on Sensor Networks*, 2004.

[5] C. Savarese, J. M. Rabaey, and J. Beutel, "Locationing in distributed ad-hoc wireless sensor networks," in *Proc. of IEEE Int. Conf. on Acoust. Speech and Signal Proc.*, May 2001.

[6] S. Čapkun, M. Hamdi, and J. P. Hubaux, "GPS-free positioning in mobile ad-hoc networks," in 34^{th} *IEEE Hawaii Int. Conf. on System Sciences*, Jan. 2001.

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