DISTRIBUTED CANCELLATION-BASED MULTIPLE-SOURCE LOCALIZATION FOR WIRELESS SENSOR NETWORKS

Dimitris Ampeliotis and Kostas Berberidis

Computer Engineering and Informatics Department and CTI/R&D
University of Patras, 26500, Rio-Patras, Greece
phone: + 30 2610 996973, fax: + 30 2610 996971,
emails: {ampeliot, berberid}@ceid.upatras.gr

ABSTRACT

A low-complexity technique for multiple-source localization by wireless sensor networks is presented. The proposed technique is based on the Received Signal Strength (RSS) measurements of signals emitted by the sources. Also, all processing is performed at the sensor nodes in a decentralized fashion, and hence no "Fusion Center" is required.

The proposed cancellation-based multiple-source localization (CBMSL) technique relies upon a proper iterative application of a new dominant-source localization (DSL) algorithm. More specifically, in the initial stage, the DSL estimates the location and power of the dominant source within the area of the network. Once the parameters of the dominant source have been estimated, they are broadcast to the network. Nodes receiving this message adjust their measurements by cancelling appropriately the components due to the dominant source. This cancellation implies that another source becomes the dominant one in the area of the sensor network. Based on the adjusted measurements, the dominant-source localization algorithm can be executed once again, to estimate the next dominant source, and so forth. Thus the above "successive cancellation" procedure can be used to estimate all sources in the network area. Efficient algorithms for all the above steps have been derived.

Extensive simulation results have shown that the proposed technique could be a promising alternative for the problem at hand.

1. INTRODUCTION

The emergence of low-cost electronic devices that integrate sensing, processing and wireless communication capabilities has fostered a growing interest in developing wireless ad-hoc sensor networks. The design, implementation, and operation of such networks requires the confluence of many disciplines, including signal processing, networking, and distributed algorithms [1]. Furthermore, wireless sensor networks must operate using minimum resources: typical sensor nodes are battery powered and have limited processing ability. The aforementioned constraints impose new challenges for algorithm development, and imply that application specific algorithms should be used in the several network layers.

In many applications, ranging from environmental monitoring to manufacturing, source localization and tracking is of major importance. Indeed, the availability of location data about objects and human beings is critical in many applications, such as tracking of valuable business assets and monitoring individuals with special needs. In this work, we focus on the localization of multiple sources that emit signals whose energy is measured by the nodes of a sensor network that has been developed over a territory of interest (e.g., vehicles transmitting acoustic signals due to the operation of their engines).

Most of the source localization methods fall mainly into two broad categories. The algorithms of the first category utilize time delay of arrival (TDOA) measurements, whereas the algorithms of the second category use direction of arrival (DOA) measurements. Direction of arrival estimates are extracted by computing phase differences between signals received by spatially remote sensors, and are particularly useful for locating sources emitting narrowband signals [2]. On the other hand, TDOA measurements offer the increased capability of localizing broadband sources [3]. The above techniques, however, have two major requirements, i.e. : (a) the analog signals arriving at each sensor should be sampled in a synchronized fashion, and (b) the sampling rate at which the analog signal is being sampled should be high enough to capture the features of interest, which, in turn, implies using high frequency electronics and increased bandwidth for communicating the measurements.

Recently [4], a new approach for source localization was proposed, that utilizes received signal strength (RSS) measurements. More specifically, based on the fact that in free space, acoustic energy decays at a rate that is inversely proportional to the distance from the source, the energy measurements at individual sensors are combined to yield an estimate of the source location. In [5], the likelihood function for multiple-source localization based on RSS measurements is formulated, the Cramer-Rao bound is derived and various optimization algorithms are proposed to yield a solution. In [6], the authors proposed a distributed *incremental subgradient* algorithm to yield the source location estimate iteratively. In [7], source location estimates that are robust to erroneous modelling of the energy decay function are derived.

Similarly to the algorithms proposed in [5], the source localization algorithm proposed in this work is able to estimate the location of multiple sources. In contrast however to [5], the proposed algorithm has smaller computational complexity and can be implemented in a decentralized fashion. Furthermore, compared to the algorithms appearing in [4],[6] and [7], the proposed algorithm offers increased capability of localizing more than one sources. The idea of using a single-source localization algorithm to perform multiple-source localization, has also appeared in [9]. However the algorithm presented therein does not involve RSS measurements.

The rest of this work is organized as follows: In Section 2, the energy decay model of [4] is presented and several assumptions are made. In Section 3, the proposed cancellation-based multiple-source localization technique is developed. In Section 4, the dominant-source localization algorithm, which is a constituent part of the CBMSL technique, is derived. Section 5 discusses the distributed implementation of the proposed technique. Section 6 presents some typical simulation results verifying the performance of the proposed algorithm and, finally, Section 7 concludes the work.

2. ATTENUATION MODEL AND ASSUMPTIONS

First, we assume that the signal strength measurements received by the sensor nodes behave according to a non-reverberant far-field model [4], [5]. More specifically, the measurement of sensor i at

This work is part of the 03ED910 research project, implemented within the framework of the "Reinforcement Programme of Human Research Manpower" (PENED) and co-financed by National and EC Funds

time instant t is assumed to be given by:

$$y_i(t) = \sum_{j=1}^{k} \frac{g_i A_j(t)}{||\mathbf{r}_i - \mathbf{x}_j(t)||^{\beta}} + w_i(t) \qquad \begin{array}{l} i = 1,\dots,n \\ t = \frac{T}{2}, \frac{3T}{2}, \dots \end{array} \tag{1}$$

where k is the number of sources/targets, n is the number of sensor nodes, $A_j(t)$ denotes the power of the j-th source at time t, as measured at 1 meter distance, $\mathbf{x}_j(t)$ is the vector of the source's coordinates, $\mathbf{r}_i$ is the vector of the coordinates of the i-th sensor and $w_i(t)$ denotes a white Gaussian noise process with positive mean σ_w^2 and variance $2\sigma_w^4/M$ [4], [5]. M denotes the number of samples averaged during the time period T before a measurement is finally acquired. Since the sensor nodes are not in general well calibrated between each other, we have introduced the parameters g_i, denoting the gains (i.e. sensitivity) of the sensors. The parameter β stands for the rate at which the energy of the signal emitted by a target decays with distance from that target. Finally, the operator $||\cdot||$ denotes Euclidean distance.

Assuming that the sources remain static at least for a period of time during the execution of the localization algorithm, we can drop the time index t in their location vectors. Similarly, we assume that the power of each source does not change significantly over a short period of time, thus, we also drop the time index from $A_j(t)$. Furthermore, the following reasonable assumptions are made:

- All sensor nodes know the location of all other nodes in their vicinity.
- There exists a mechanism for "announcing" a message to all of the nodes in the network. In the following, we will call this mechanism a "broadcast" message, although in practise it may be implemented via a flooding algorithm.
- We assume that the the number of targets k is known in advance. In [8], an algorithm for counting the targets present in the area of the network was presented.
- For the derivation of the localization algorithm, we assume that all sensors are well calibrated (i.e. $g_i = 1$) and that $\beta = 2$. In section 6 we examine how the developed algorithm operates when the above assumptions do not hold.

Furthermore, when target τ is the one to be localized, the signal component due to the other targets is considered as *interference*. The interference term at the i-th node is given by:

$$I_i^{(\tau)} = \sum_{j=1, j\neq\tau}^{k} \frac{A_j}{||\mathbf{r}_i - \mathbf{x}_j||^2} \qquad i = 1,2,\dots,n \tag{2}$$

Obviously, in a single-source localization scenario, the interference signal is equal to zero. When the locations $\mathbf{x}_j$ and powers A_j are not known exactly, and $\hat{k} \leq k$ sources have been estimated, the estimated values $\hat{\mathbf{x}}_j$ and $\hat{A}_j$ are used in (2), which is rewritten as:

$$\hat{I}_i^{(\tau,\hat{k})} = \sum_{j=1, j\neq\tau}^{\hat{k}} \frac{\hat{A}_j}{||\mathbf{r}_i - \hat{\mathbf{x}}_j||^2} \qquad i = 1,2,\dots,n \tag{3}$$

3. CANCELLATION BASED MULTIPLE SOURCE LOCALIZATION

In Table 1 the whole CBMSL technique is summarized. As already mentioned, the proposed technique performs a multiple-source localization task by iteratively applying a dominant-source localization algorithm. More specifically, assume that $\tau - 1 < k$ sources have already been detected and their locations and powers have been estimated. Assume also that these estimates are known by all sensor nodes. Then, before estimating the next target τ, each sensor node may adjust its energy reading by cancelling the influence of the sources already estimated, that is

$$y_i^{(e)}(t) = y_i(t) - \hat{I}_i^{(\tau,\tau-1)} \qquad i = 1,2,\dots,n \tag{4}$$

```
Input: k, y_i(t), r_i,        i = 1,2,...,n
Output: x̂_j, Â_j             j = 1,2,...,k
FOR q = 1 TO N
  FOR τ = 1 TO k
      IF (q = 1)
          y_i^(e)(t) = y_i(t) − Î_i^(τ,τ−1)
          Elect the CPA node cpa_τ at r_cpa_τ using 𝒫1
          Set-up a cluster of nodes S_min^cpa_τ using 𝒫2
      ELSE
          y_i^(e)(t) = y_i(t) − Î_i^(τ,k)
      END
      Estimate x̂_τ, Â_τ using 𝒫3
      Broadcast x̂_τ, Â_τ
  END
END
```

Table 1: Cancellation-Based Multiple-Source Localization in terms of its constituent algorithms $\mathscr{P}_1$, $\mathscr{P}_2$, and $\mathscr{P}_3$

Using the above "effective" energy readings (denoted via superscript e), a distributed algorithm, so-called $\mathscr{P}_1$, is executed in order to elect the sensor node with the maximum effective reading, assuming that this is the *Closest Point of Approach* (CPA) sensor node, i.e. the nearest node to the target of interest [4]. This node, denoted as cpa_τ, will play a leading role in estimating the target which is in the vicinity of its location.

After the election of cpa_τ, a distributed algorithm, so-called $\mathscr{P}_2$, is initiated in order to select those c nodes in the proximity of node cpa_τ which receive a relatively "clear" signal, that is a signal with small amount of interference.The nodes that are selected define the set $S_{min}^{cpa_\tau}$, which will be formally defined in subsection 4.2.

Once the cluster of nodes has been setup, another distributed algorithm, so-called $\mathscr{P}_3$, takes over in order to provide estimates $\hat{A}_\tau$ and $\hat{\mathbf{x}}_\tau$ of the target power A_τ and location $\mathbf{x}_\tau$. These estimates, once computed, are transmitted to the network, via a broadcast message. Thus, all nodes receiving this message, can compute their new effective readings.

The algorithms $\mathscr{P}_1$, $\mathscr{P}_2$ and $\mathscr{P}_3$ constitute the dominant-source localization algorithm and will be discussed in detail in Section 4. Once the above dominant-source localization - cancellation procedure has estimated all k targets, the whole procedure can be restarted. At this time, for each target τ, the effective energy readings are computed taking into account all the other $k-1$ targets that have been estimated either in the current or in the previous iteration, that is

$$y_i^{(e)}(t) = y_i(t) - \hat{I}_i^{(\tau,k)} \qquad i = 1,2,\dots,n \tag{5}$$

In general, N iterations can be executed as indicated in Table 1, where at each iteration q the interference signal is computed using the latest available estimates $\hat{\mathbf{x}}_j$ and $\hat{A}_j$ for the locations and powers of all $k-1$ targets (i.e., the estimates may come either from iteration q or the previous one). The number of iterations can be selected such that a trade-off between estimation accuracy and total execution time is achieved. Furthermore, algorithms $\mathscr{P}_1$ and $\mathscr{P}_2$ are executed only at the first iteration $(q = 1)$ and their results are used during all subsequent iterations (static targets assumed).

Of course, such an iterative interference-cancellation procedure will impose propagation of errors, since the erroneous estimation of a target may be used to localize other targets. The simulation results presented in Section 6, indicate that when the number of sensors is large enough, error propagation has limited effect to the accuracy of the algorithm.

4. DOMINANT SOURCE LOCALIZATION

Dominant-source localization is performed in three steps: (a) Coarse estimation, (b) Cluster setup and (c) Fine estimation. These steps correspond to the algorithms $\mathscr{P}_1$, $\mathscr{P}_2$ and $\mathscr{P}_3$ respectively, that are explained in the sequel. Details regarding the distributed implementation of $\mathscr{P}_1$, $\mathscr{P}_2$ and $\mathscr{P}_3$, are discussed in Section 5.

4.1 Coarse Estimation: CPA Election (algorithm $\mathscr{P}_1$)

At the first iteration of the CBMSL technique and for each target τ, the sensor node with the maximum effective reading is elected as CPA node, that is, $cpa_\tau = \arg\max_i\{y_i(t) - \hat{I}_i^{(\tau,\tau-1)}\}$. Thus, assuming that the dominant source is in the vicinity of cpa_τ, a coarse estimate for the location of this dominant source is $\mathbf{r}_{cpa_\tau}$.

4.1.1 *Analysis of CPA accuracy*

Let us consider that n sensor nodes are uniformly deployed over a region of area E. We may define the *spatial density* of such a network as $d = n/E$. Consider also that a target is placed in the same region, and define a circular area R of radius ρ around the target. Define also the random variable Y as the distance between the target and the closest sensor node. Then, it follows that the probability that the closest sensor node lies in R is given by:

$$\begin{aligned} \Pr\{Y < \rho\} &= \Pr\{(\text{number of nodes in } R) \geq 1\} \\ &= 1 - \Pr\{(\text{number of nodes in } R) = 0\} \\ &= 1 - \left(1 - \pi d \rho^2 \frac{1}{n}\right)^n . \end{aligned} \quad (6)$$

Thus, as n and $E = n/d$ tend to infinity, we have that

$$\Pr\{Y < \rho\} = 1 - e^{-\pi d \rho^2} . \quad (7)$$

Differentiating the above cumulative density with respect to ρ, we get the probability density function of Y as $f_Y(\rho) = 2\pi d \rho e^{-\pi d \rho^2}$. The expected value of Y, which may be viewed as the expected error of the coarse estimation algorithm for the case of single source localization, is thus found to be

$$\mu = E[Y] = \int_0^{+\infty} \rho \cdot f_Y(\rho) d\rho = \frac{1}{2\sqrt{d}}. \quad (8)$$

This parameter, will be used in the sequel to initialize the search step of the algorithm presented in subsection 4.3.

4.2 Cluster Selection (algorithm $\mathscr{P}_2$)

A simple approach to select c sensor nodes in the vicinity of node cpa_τ would be to choose the first c nodes which are closest to the leader cpa_τ. However, some of the nodes in such a cluster might receive a relatively large interfering signal from a nearby target. To avoid taking into account such nodes we suggest testing a number of different clusters around cpa_τ, each one "biased" along a certain direction around the leader, and select the one with minimum interference.

A measure of the involved interference can be derived by observing that nodes that are equally away from a target should have the same measurements, if that target was the only one in that area. On the other hand, if two nodes are equally away from a target of interest, and only one of them is close to another target, then this one should have a greater energy measurement. Assuming that the coarse estimate of the dominant source is close to the actual source location, distances from the target can be considered as distances from $\mathbf{r}_{cpa_\tau}$.

Thus, an algorithm to implement $\mathscr{P}_2$ can be: Create clusters $S_i^{cpa_\tau}$ $i = 0,\ldots,l-1$, by selecting the nodes inside the circle centered at

$$\mathbf{p}_i^{cpa_\tau} = \mathbf{r}_{cpa_\tau} + \alpha \cdot \delta \begin{bmatrix} \cos(2\pi \cdot i/l) \\ \sin(2\pi \cdot i/l) \end{bmatrix} \quad i = 0,\ldots,l-1 \quad (9)$$

with radius δ. More specifically, $S_i^{cpa_\tau} = \{j : ||\mathbf{p}_i^{cpa_\tau} - \mathbf{r}_j|| \leq \delta\}$, where δ can be selected such that the expected number of sensors in

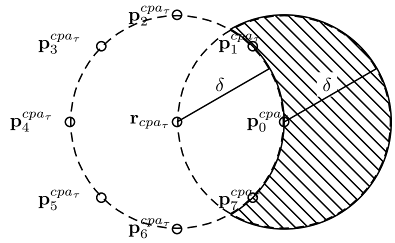

Figure 1: The sum of the measurements of the nodes in the dashed area, gives a measure of the interference in this direction, relative to the other directions. Here $\alpha = 1, l = 8$

the circle is c, that is $\delta = \sqrt{c/(d\pi)}$ and the parameter α is a positive constant in $(0,1]$ that controls the "bias" of the clusters away from $\mathbf{r}_{cpa_\tau}$ in each direction. Clearly $cpa_\tau \in S_i^{cpa_\tau}$ $\forall\, i = 0,\ldots,l-1$.

According to the discussion above, a measure of the relative interference in cluster $S_i^{cpa_\tau}$ would be

$$\mathscr{F}(S_i^{cpa_\tau}) = \sum_{j:||\mathbf{r}_j - \mathbf{r}_{cpa_\tau}|| > \delta\ ,\ j \in S_i^{cpa_\tau}} y_j(t) \quad (10)$$

that is, the sum of the measurements of the cluster nodes whose distance from $\mathbf{r}_{cpa_\tau}$ is larger than δ, provides a measure of the relative interference in that cluster. Figure 1 demonstrates the concept of cluster interference. The nodes in the cluster of minimum interference

$$S_{min}^{cpa_\tau} = \arg\min_{S_i^{cpa_\tau}} \{\mathscr{F}(S_i^{cpa_\tau})\} \quad (11)$$

will collaborate in the sequel, for estimating the target parameters.

4.3 Fine Estimation: Greedy Search (algorithm $\mathscr{P}_3$)

Assuming that the cluster of nodes (selected as in the previous step) receives measurements due to one target solely, the cost function that must be minimized is

$$J(\theta) = \sum_{j \in S_{min}^{cpa_\tau}} \left(y_j^{(e)}(t) - \sigma_w^2 - \frac{A_\tau}{||\theta - \mathbf{r}_j||^2} \right)^2 \quad (12)$$

In order to minimize the above cost function, we use a greedy search algorithm similar to the Expectation - Maximization solution proposed in [5]. The knowledge of the node cpa_τ, that is the closest node to the source of interest, can be exploited so as to avoid a costly exhaustive search.

The cost function in (12), requires knowledge of the source power A_τ and thus, cannot be evaluated directly. To circumvent this problem, one may first estimate the parameter A_τ that minimizes the cost function for fixed θ, and then evaluate the modified cost function

$$\hat{J}(\theta) = \sum_{j \in S_{min}^{cpa_\tau}} \left(y_j^{(e)}(t) - \sigma_w^2 - \frac{\hat{A}_\tau}{||\theta - \mathbf{r}_j||^2} \right)^2 \quad (13)$$

where

$$\hat{A}_\tau = \frac{\sum_{j \in S_{min}^{cpa_\tau}} \frac{y_j^{(e)}(t) - \sigma_w^2}{||\theta - \mathbf{r}_j||^2}}{\sum_{j \in S_{min}^{cpa_\tau}} \frac{1}{||\theta - \mathbf{r}_j||^4}} . \quad (14)$$

It can be easily seen that (12) and (13) have the same global minimum.

Thus, the search algorithm sets the initial estimate $\hat{\mathbf{x}}_\tau = \mathbf{r}_{cpa_\tau}$ (or, in general, the best current estimate of the location of the corresponding source, for $q > 1$) and in the sequel, at sub-iteration[1] i it

[1] The iterations of the search algorithm are denoted as sub-iterations so as not to be confused with the iterations of the CBMSL

evaluates the above cost function at the L points

$$\theta_j^{(i)} = \hat{\mathbf{x}}_\tau + a_{i,q} \begin{bmatrix} \cos(2\pi \cdot j/L) \\ \sin(2\pi \cdot j/L) \end{bmatrix} \qquad j = 0, \ldots, L-1 \qquad (15)$$

that lie on a circle of radius $a_{i,q}$ around the current estimate $\hat{\mathbf{x}}_\tau$. If for some point the cost function takes a lower value than the current minimum value, the algorithm sets this point as the current estimate $\hat{\mathbf{x}}_\tau$. The minimum cost is initially set equal to a large number. The step size $a_{i,q}$ used at sub-iteration i and iteration q, should be a monotonically decreasing sequence with respect to i and q. For the first sub-iteration of the first iteration, it is reasonable to use $a_{1,1} = \mu$, since this is the expected distance of the source from the CPA node (see eq.(8)). As for the following sub-iterations/iterations, it was experimentally observed that a step size of the form $a_{i,q} = \frac{\mu}{i \cdot q}$ yields good results.

5. DISTRIBUTED IMPLEMENTATION

5.1 Algorithm $\mathscr{P}_1$

Choosing the sensor node with maximum measurement as the leader node, is a problem closely related to the problem of "leader election" [10]. However, in this case, the sensor measurements can be exploited so as to develop a specific election algorithm. In this section, based on the theory of randomized algorithms [11], we develop a distributed algorithm which employs broadcast messages to detect the node of maximum measurement.

The idea of the algorithm, is that every node j in the network attempts to elect itself as leader by sending, with a given probability, a broadcast message containing its measurement. More specifically, we assume that sensor nodes are equipped with uniform random generators, that produce independent pseudo-random numbers in $[0,1]$. Within a predetermined time period, termed as time-slot, each sensor draws a random number, and if this number is smaller than its probability of broadcast, then it sends a relevant message. Let us denote as $p_m(y_j^{(e)}(t))$ the probability assigned to node j, where m is the time-slot. Function $p_m(\cdot)$ should give higher probabilities of transmission to the nodes of higher measurements. Such a function will be discussed later on in this subsection. Of course, in such a scenario, collisions of messages may occur, and also there is a probability that none of the sensor nodes transmits. Successful transmission occurs when only one node transmits at a time. In this case, all nodes that receive this message compare the received measurement with their own measurement, and if their measurement is smaller, they stop broadcasting by setting $p_m(y_j^{(e)}(t)) = 0$. Clearly, the algorithm stops when the node of maximum measurement is the only one succeeding to transmit.

Assuming an ordering of the nodes in the network such that $y_1^{(e)}(t) > y_2^{(e)}(t) > \ldots > y_n^{(e)}(t)$, then we can define the probability ω_m that the node of higher measurement is the only one that transmits in time-slot m as

$$\omega_m = p_m(y_1^{(e)}(t)) \prod_{j=2}^{n} (1 - p_m(y_j^{(e)}(t))) \qquad (16)$$

Obviously, since at each time-slot, some of the nodes of smaller measurement ($j \geq 2$) may set $p_m(y_j^{(e)}(t)) = 0$, we have that $\omega_m \geq \omega_1$. Thus, the probability that the algorithm stops within the first K time-slots, is given as $\Omega_K \geq 1 - (1-\omega_1)^K$ since $(1-\omega_1)^K$ is an upper bound of the probability that the "stoping message" is not transmitted within the first K time-slots.

Concerning now the function $p_m(\cdot)$, the optimal choice is the one that maximizes the probability ω_1. However, in case of single-source localization where a source of known power B is present in the network, a reasonable choice is to set it equal to the probability that each node is the CPA node. Since the power of the source is known, each node j can estimate its distance ρ_j from the source based on its energy measurement. Then, the probability that node j is the CPA node is equal to the probability that no sensor node exists in a circle of radius ρ_j, and is given by the second term of equation (7). More specifically, for a node j with measurement $y_j(t)$ the respective probability is $p_m(y_j(t)) = e^{-\pi d \rho_j^2} = e^{-\pi d \frac{B}{y_j(t) - \sigma_w^2}}$ for $y_j(t) > \sigma_w^2$. Of course, nodes with measurements $y_j(t) \leq \sigma_w^2$ are far away from the target and do not participate in CPA election.

Furthermore, in the cases where the power B is not known and/or there are multiple sources to be located, one may use a time varying value for B given by

$$B_m = \begin{cases} B_{m-1} & \text{Successful transmission at slot } m-1 \\ B_{m-1}/G & \text{No node transmitted at slot } m-1 \\ G \cdot B_{m-1} & \text{A collision occured at slot } m-1 \end{cases} \qquad (17)$$

initialized at an arbitrary value for B_1, with $G > 1$ being a constant.

5.2 Algorithm $\mathscr{P}_2$

Cluster set-up, can be implemented in a decentralized fashion, as soon as the CPA node is elected. More specifically, when the CPA node is elected, l nodes in the vicinity of the CPA node (that may be predetermined, for all possible CPA nodes) start computing the sums defined by equation (10). Each one of the l nodes that finishes the computation of the sum, reports the value computed to the CPA node. Thus, the CPA node is able to compute the minimum value according to (11).

5.3 Algorithm $\mathscr{P}_3$

It is interesting to note, that since both (13) and (14) involve summations over the nodes of the cluster $S_{min}^{cpa_\tau}$, they can be computed in a distributed fashion where each node in the cluster receives a partial sum from a neighbor, adds its own contribution, and forwards the new partial sum to the next node. Thus, two circles are required for testing one point $\theta_j^{(i)}$, one for computing the corresponding value $\hat{A}_\tau$ and one for evaluating the cost function in (13). If the algorithm performs N' sub-iterations, $N' \cdot L$ points are tested and $2 \cdot N' \cdot L$ circles are required for each target, for every iteration of the CBMSL. The value of $\hat{A}_\tau$ computed by node cpa_τ at the end of every first circle, should be communicated to all the nodes in the cluster.

Generally, the CBMSL algorithm requires $3 \cdot N \cdot k$ broadcast messages for announcing the coordinates and amplitude (2D case) of targets, plus those required by $\mathscr{P}_1$. $\mathscr{P}_2$ requires $k \cdot (l + l \cdot c)$ local messages, and $\mathscr{P}_3$ requires $2 \cdot N' \cdot k \cdot N \cdot L \cdot (c-1)$ local messages. The complexity of the algorithm is mainly due to $\mathscr{P}_3$, which evaluates (13) for $N \cdot k \cdot N' \cdot L$ times. As a comparison, note that Maximum Likelihood (ML) estimation [5] is based on exhaustive search over a grid. For a 2D grid of size $s \times s$, s^{2k} evaluations are required. Furthermore, in (13), only c measurements are involved whereas for ML estimation all n measurements should be used.

6. SIMULATION RESULTS

In order to assess the performance of the proposed CBMSL technique, some typical numerical simulations were conducted. In the following, we assume that the target association is done properly so that the sum of the location estimation errors is the minimum of the two possible assignments.

6.1 Discriminating two identical sources

In this experiment, a sensor network was used to localize two identical ($A_1 = A_2 = 10000$) sources. Two different sensor densities were examined, corresponding to $n = 50$ and $n = 100$ nodes over an area of 100×100 meters, and the number of sensor nodes c that collaborated in each case was set equal to 20 and 45 respectively. The two sources were placed at $(-D/2, 0)$ and $(D/2, 0)$ where point (0,0) corresponds to the center of the sensor deployment field. The mean of the noise was set equal to $\sigma_w^2 = 40$ and the rest of the parameters used are $M = 100$, $N = 5$, $l = 10$, $a = 0.6$, $N' = 10$ and

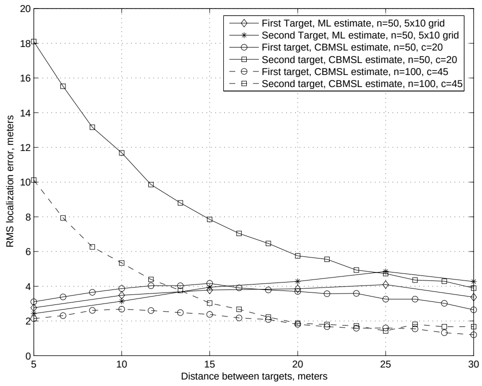

Figure 2: Localization error

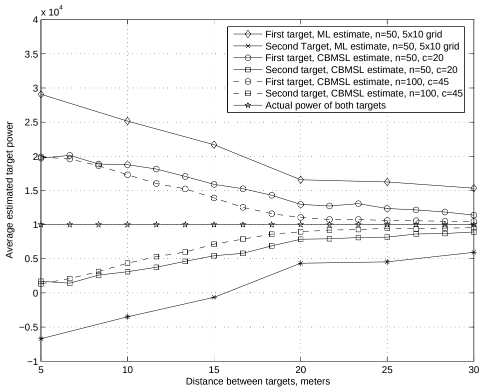

Figure 3: Estimated power

$L = 4$. Figures 2 and 3 summarize the localization errors obtained and the target power estimates respectively, as a function of the targets distance D. For comparison purposes, the performance of the Maximum Likelihood [5] estimation algorithm is also presented. To minimize experimentation time for ML estimation, we used a 5×10 grid close to the actual locations of the sources. For both of the algorithms examined, we consider the source with maximum estimated power as the first one.

From Figure 2, we note that the CBMSL technique provides quite accurate estimates for the location of the first target that are comparable to the estimates of ML. The location estimate for the second source is quite poor when the targets are close to each other, but as target spacing increases better performance is obtained, again close to the performance of ML estimation. Of course, an increase in the sensor network density improves the ability of the CBMSL algorithm to distinguish the two targets dramatically. From Figure 3, we note that when the two sources are close to each other, both algorithms estimate a source with very high power and another one with very small power.

6.2 Robustness against perturbation of parameters

In this experiment, 50 sensors were placed randomly (uniformly) over an area of 100×100 meters and two unequal power sources ($A_1 = 10000, A_2 = 5000$) were randomly (uniformly) placed over the central 50×50 area to avoid boundary effects. Here, the source with power A_1 is considered as the first one. Table 2 presents the results of the CBMSL algorithm in terms of the localization error for the first source e_1, the error for the second source e_2 and the respective estimated standard deviations σ_1 and σ_2, as the average of 1000 independent runs. $\Delta g = x$ means that the sensor gains g_i were uniformly chosen to lie in $[1-x, 1+x]$, and $\Delta \mathbf{r} = x$ means that the sensor coordinates contain uniform error in $[-x, x]$. The rest of the

Configuration	e_1 (m)	σ_1	e_2 (m)	σ_2
$\sigma_w^2 = 0, \Delta g = 0, \Delta \mathbf{r} = 0, \beta = 2$	3.53	5.67	4.48	6.70
$\sigma_w^2 = 40, \Delta g = 0, \Delta \mathbf{r} = 0, \beta = 2$	4.52	7.12	7.09	10.68
$\sigma_w^2 = 80, \Delta g = 0, \Delta \mathbf{r} = 0, \beta = 2$	6.25	10.44	9.51	13.54
$\sigma_w^2 = 0, \Delta g = 0.2, \Delta \mathbf{r} = 0, \beta = 2$	3.99	5.24	5.48	6.80
$\sigma_w^2 = 0, \Delta g = 0.6, \Delta \mathbf{r} = 0, \beta = 2$	5.54	5.61	8.28	8.64
$\sigma_w^2 = 0, \Delta g = 0, \Delta \mathbf{r} = 1, \beta = 2$	3.90	5.22	5.76	7.48
$\sigma_w^2 = 0, \Delta g = 0, \Delta \mathbf{r} = 2, \beta = 2$	4.74	5.59	6.69	7.64
$\sigma_w^2 = 0, \Delta g = 0, \Delta \mathbf{r} = 0, \beta = 2.5$	4.19	5.24	5.86	8.01
$\sigma_w^2 = 0, \Delta g = 0, \Delta \mathbf{r} = 0, \beta = 3$	5.16	5.66	7.28	9.26

Table 2: CBMSL accuracy under various parameters

algorithm parameters were as in the previous experiment. From Table 2, we note that the proposed CBMSL algorithm is quite robust with respect to various perturbations of the energy decay model. Localization error for the first target is smaller compared to e_2, mainly due to the fact that the first target emits more power.

7. CONCLUSION

In this work, a multiple-source localization algorithm for sensor networks, amenable to distributed implementation, has been proposed. Extensive simulation experiments have been conducted in order to assess the performance of the new technique, showing that it exhibits promising performance characteristics.

Further work will focus on aspects such as the effects of error propagation, analysis of the messages required for algorithm $\mathscr{P}_1$, as well as a study on how the various parameters of the algorithm can be selected to guarantee certain accuracy.

REFERENCES

[1] F. Zhao and L. J. Guibas. *Wireless Sensor Networks: An Information Processing Approach*. Morgan Kaufmann, 2004.

[2] S. Haykin. *Array Signal Processing*. Englewood-Cliffs, NJ: Prentice-Hall, 1985.

[3] N. Owsley and G. Swope. ”Time delay estimation in a sensor array”, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29(3):519–523, June 1981.

[4] D. Li and Y.-H. Hu. ”Energy-based collaborative source localization using acoustic microsensor array”, *EURASIP Journal on Applied Signal Processing*, 4:321–337, 2003.

[5] X. Sheng and Y.-H. Hu. ”Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks”, *IEEE Transactions on Signal Processing*, 53(1):44–53, January 2005.

[6] M. G. Rabbat and R. D. Nowak. ”Decentralized source localization and tracking”, In Proc. of *ICASSP 2004*, pages 921–924, Montreal, Quebec, Canada.

[7] M. G. Rabbat, R. D. Nowak, and J. Bucklew. ”Robust decentralized source localization via averaging”, In Proc. of *ICASSP 2005*, pages 1057–1060, Philadelphia, PA, USA.

[8] Q. Fang, F. Zhao, and L. Guibas. ”Counting targets: Building and managing aggregates in wireless sensor networks”, Technical report, Palo Alto Research Center (PARC), June 2002.

[9] J. C. Chen, R. E. Hudson, and K. Yao, ”Maximum-Likelihood source localization and unknown sensor location estimation for wideband signals in the near field”, *IEEE Transactions on Signal Processing*, Vol. 50, No. 8, August 2002

[10] H. Garcia-Molina. ”Elections in a distributed computing system”, *IEEE Transactions on Computers*, C-31(1):47–59, January 1982.

[11] R. Motwani and P. Ragavan. *Randomized Algorithms*. Cambridge University Press, 1995.