Wireless Sensor Network, **2010, 2, 562-570**
doi:10.4236/wsn.2010.27068 Published Online July 2010 (http://www.SciRP.org/journal/wsn)

Information-Driven Collaborative Processing for Diffusive Source Estimation in Wireless Sensor Networks

Hossein Khonsari, Mohammad Hossein Kahaei
Electrical Engineering Department, Iran University of Science and Technology, Tehran, Iran
E-mail: *hossein.khonsari@gmail.com, kahaei@iust.ac.ir*

Received April 6, 2010; *revised April* 21, 2010; *accepted May* 2, 2010

Abstract

This paper discusses an accurate distributed algorithm for diffusive source localization while maintaining the low energy consumption of sensor nodes in wireless sensor networks. In this algorithm, the sensor selection scheme based on the information utility measure is used. To update the estimation in each selected node, a neighborhood radius equal to the communication range of the sensor nodes is defined and all sensors located in the neighborhood circle, whose radius is equal to the neighborhood radius and the selected node is its centre, collaborate their information. To decrease the energy consumption, the neighborhood radius is reduced gradually based on the error covariance value of the estimation. In addition, this paper includes a new method for the initial point calculation which is important in the recursive methods used for distributed algorithms in wireless sensor networks. Numerical examples are used to study the performance of the algorithms. Simulation results show the accuracy of the new algorithm becomes better while its energy consumption is low enough.

Keywords: Information-Driven Collaborative Processing, Wireless Sensor Network, Diffusive Source Localization

1. Introduction

Densely scattered low-cost sensor nodes provide a rich and complex information source about the sensed world. A major application of wireless sensor networks is to perform monitoring tasks like localization and tracking of one or more targets in their coverage field. In this paper, we focus on the localization of a diffusive source. The problem under consideration has application in the fields of security, environmental and industrial monitoring and pollution control [1]. Since in a typical wireless sensor network each sensor node has limited battery power, efficient collaborative signal processing algorithms that consume less energy for computation and less bandwidth for communication is needed [2]. In centralized estimation methods, sensor nodes transmit their row observations to a fusion centre for processing [3-5]. However, some of their innate properties such as high-energy consumption limit their use in a wireless sensor network [6]. Therefore, recent research has concentrated on developing distributed processing.

Most of the current distributed estimation methods are categorized in to two groups. In one group, the distributed estimation is developed using the sequential Bayesian method [7-9]. In this method, the state belief (posterior density function) is updated in the selected nodes and when the quality of estimation is good enough (base on the predefined threshold), the sensor selection is stopped and the location is estimated. The most important disadvantage of this type of method is that the convergence of the estimation is not easy to be proven [6]. In the other group, the distributed estimation is yielded by implementing the common centralized estimation methods like Maximum Likelihood (ML) estimation in a distributed manner [10]. As in this type the communication burden is high, the most important challenge is to develop the accuracy of algorithms while considering the total energy consumption. In this paper, we propose a new method to improve the accuracy of diffusive source estimation while maintaining the energy consumption in a reasonable level.

The recent relevant work is presented in [6] where a distributed Information-Driven Collaborative Processing (IDCP) is derived. The essential point in this algorithm is that by applying information utility measure, a sequence of sensor nodes is chosen to reduce the required data communications under the framework of ML estimation;

however, for estimation update, only selected node observations are used. In this paper, we propose the modification of the above method to improve the estimation performance. In our algorithm called Modified IDCP (MIDCP), each node calculates the new estimation of the target location by averaging over the information of its own observation and neighbors. However, the energy consumption increases because of large amount of transmission. To mitigate this energy consumption we propose Energy Efficient MIDCP (EFMIDCP) in which the neighborhood radius is decreased gradually based on the estimation covariance value. In general, Iterative methods used in distributed algorithms may not yield the ML estimate if the initial value is not selected carefully. Therefore, we propose a new method to calculate initial value for the iteration algorithms in our scenario.

This paper is organized as follows. In Section 2, we present the physical and statistical models of the released substance distribution. The IDCP algorithm is investigated in Section 3. In Section 4, we propose the MIDCP and EFMIDCP algorithms. Section 5 includes a new method to calculate initial estimation in all of these iterative algorithms. In Section 6 the numerical examples are used to demonstrate the performance of the proposed methods. Conclusions are presented in Section 7.

2. Problem Formulation

2.1. Physical Model of a Difusive Source

Let $c(\mathbf{r},t)$ denote the substance concentration diffuseion at a position $\mathbf{r}=(x,y)$ and time *t*. For a source-free volume and space-invariant diffusivity, without biomedical reaction during the transport of biochemical agents, the concentration of a diffusing substance follows the well-known diffusion equation [5]:

$$\frac{\partial c}{\partial t}=k\nabla^2 c=k\left(\frac{\partial^2 c}{\partial x^2}+\frac{\partial^2 c}{\partial y^2}\right) \tag{1}$$

Where *k* is called the diffusivity in units of kg/m^{2}s and dependence on *r* and *t* is omitted to simplify the presentation. It has been proved that in a rectangular parallelepiped space and for certain types of initial and boundary conditions, the solution of (1) is the product of the solutions of the two single spatial-variable problems [5]. That is,

$$c(\mathbf{r},t)=c_x(\mathbf{r},t)c_y(\mathbf{r},t) \tag{2}$$

where $c_x(\mathbf{r},t)$ is the solution of the diffusion equation with single spatial-variable *x:*

$$\frac{\partial c_x(x,t)}{\partial t}=k\frac{\partial^2 c(x,t)}{\partial x^2} \tag{3}$$

with the similar definition for $c_y(y,t)$. We use the distribution equation which is derived in [5] for a continuous point diffusive source (our desired source). The concentration of our source in a semi-infinite medium is obtained using the following integral:

$$c(\mathbf{r},t)=\mu\int_{t_I}^{t}\frac{2}{8\left[\pi k(t-\tau)\right]^{\frac{3}{2}}}\times\left\{\exp\left[-\frac{|\mathbf{r}\text{ - }\mathbf{r}_0|}{4k(t-\tau)}\right]\right\}d\tau \tag{4}$$

where $\mathbf{r_0}=(x_0,y_0)$ represents the exact source location, μ is the substance realizing rate and t_I is the initial time of diffusion.

2.2. Statistical Measurement Model

We assume that a continuous point diffusive source is at position $\mathbf{r_0}=(x_0,y_0)$, from which the diffusion substance is liberated continuously at a certain rate μ and at time t_I. The sensor nodes of a WSN are deployed in which each sensor node is located at a known position and can measure the substance concentration from the diffusive sources. The measurement taking by sensor node *i* at time t_j can be written as:

$$\begin{aligned}&y(\mathbf{r}_i,t_j)=c(\mathbf{r}_i,t_j)+b+e(\mathbf{r}_i,t_j)\\&e(r_i,t_j)\sim N(0,\sigma^2),\end{aligned} \tag{5}$$

where $\mathbf{r}_i$ is the known location of sensor node *I*, the term $c(\mathbf{r}_i,t_j)$ is the concentration distribution shown in Equation (4), *b* is a constant bias term representing the sensor's response to foreign substances assumed to be a known and $e(\mathbf{r}_i,t_j)$ is the sensor measurement's Gaussian noise independent in space and correlated in time, *i.e.*,

$$\mathbf{E}\left[e\left(\mathbf{r}_{i_1},t_{j_1}\right)e\left(\mathbf{r}_{i_2},t_{j_2}\right)\right]=\begin{cases}\sigma_{j_1j_2} & i_1=i_2\\ 0 & i_1\neq i_2\end{cases}. \tag{6}$$

Therefore, for each sensor node, we have e$_i$ = $[e_{i,1},\ldots,e_{i,N}]^T$, an N-dimension Gaussian distributed random vector with the mean zero and covariance matrix Σ_i. By denoting $y_{ij}=y(\mathbf{r}_i,t_j)$, $e_{ij}=e(\mathbf{r}_i,t_j)$ and $\mu\alpha_{ij}(\boldsymbol{\theta})=c(\mathbf{r}_i,t_j)$, we can rewrite (1) as:

$$y_{ij}=\mathbf{a}_{ij}^T(\boldsymbol{\theta})\mathbf{x}+e_{ij}. \tag{7}$$

Where $\mathbf{a}_{ij}(\boldsymbol{\theta})=\left[\alpha_{ij}(\boldsymbol{\theta}),1\right]^T$ and $c(r_i,t_j)=\mu\alpha_{ij}(\boldsymbol{\theta})$,

$\mathbf{x}=[\mu,b]^T$ is called the linear parameter vector and $\boldsymbol{\theta}=[x_0,y_0]^T$ represents the source parameters.

3. Information-Driven Collaborative Processing

M sensor nodes are activated to estimate the location of the diffusive source located in the network's range. The covariance matrix, Σ_i, is known for each node because this matrix can be estimated during the calibration step in practical applications. The measurements $\{y_1,...,y_M\}$ of all sensor nodes can be written as a vector form [6]:

$$\mathbf{y}=\mathbf{A}(\boldsymbol{\theta})\mathbf{x}+\mathbf{e} \quad (8)$$

where $y=\left[y_1^T,...,y_M^T\right]^T$ is a $MN\times 1$ vector, $A(\theta)=\left[A_1^T(\theta),...,A_M^T(\theta)\right]^T$ is a $MN\times 2$ matrix and $e=\left[e_1^T,...,e_M^T\right]^T$ represents the additive Gaussian noise. Regarding to the assumptions, it follows a *MN*-dimension multiple normal distribution with mean zero and covariance matrix $\boldsymbol{\Sigma}$ with $\boldsymbol{\Sigma}=\mathrm{diag}\{\boldsymbol{\Sigma}_1,...,\boldsymbol{\Sigma}_M\}$, $\boldsymbol{\Sigma}$ is diagonal because the measurement noise of each two sensors is independent. We assume each sensor-node takes N samples.

According to the above measurement model, the log-likelihood function of the measurements vector **y** is [6]:

$$\begin{aligned}L(\mathbf{y};\boldsymbol{\theta})&\\&=-\frac{M}{N}\log 2\pi-\frac{1}{2}\sum_{i=1}^{M}\log|\Sigma_i|\\&\quad-\frac{1}{2}\sum_{i=1}^{M}\left[\mathbf{y}_i-A_i(\boldsymbol{\theta})\mathbf{x}\right]^T\Sigma_i^{-1}\left[\mathbf{y}_i-A_i(\boldsymbol{\theta})\mathbf{x}\right].\end{aligned} \quad (9)$$

By maximizing this log-likelihood function, the ML estimation of $\boldsymbol{\theta}$ is obtained:

$$\begin{aligned}\hat{\boldsymbol{\theta}}&=\arg\min_{\boldsymbol{\theta}}\\&\left\{\sum_{i=1}^{M}\left[\mathbf{y}_i-\mathbf{A}_i(\boldsymbol{\theta})\mathbf{x}\right]^T\boldsymbol{\Sigma}_i^{-1}\left[\mathbf{y}_i-\mathbf{A}_i(\boldsymbol{\theta})\mathbf{x}\right]\right\}.\end{aligned} \quad (10)$$

To derive a distributed estimation, it is necessary to use an incremental optimization method. Therefore, the Gauss- Newton method is used to implement a distributed estimation. When employing this method to solve the non-linear weighted least-square problem in (10), an iterative process is obtained to update the estimate of $\boldsymbol{\theta}$ [6]:

$$\begin{aligned}\boldsymbol{\theta}^{(k+1)}&=\boldsymbol{\theta}^{(k)}\\&+\left[H^T(\boldsymbol{\theta}^{(k)})\Sigma^{-1}H(\boldsymbol{\theta}^{(k)})\right]^{-1}\\&\times H(\boldsymbol{\theta}^{(k)})\Sigma^{-1}(\mathbf{y}-A(\boldsymbol{\theta}^{(k)})\mathbf{x}),\end{aligned} \quad (11)$$

where $H(\boldsymbol{\theta})=\dfrac{\partial[A(\boldsymbol{\theta})\mathbf{x}]}{\partial\boldsymbol{\theta}}$, is an $MN\times P$ Jacobian matrix, when P parameters of desired source are unknown. As it is necessary to collect all sensor node observations to update the estimation, it is not a fully distributed algorithm. So, the fully distributed information-driven collaborative processing (IDCP) investigated in the next section.

3.1. Distributed Information_Driven Maximum Likelihood Estimation

It is proven that the update stage of the Gauss-Newton method can be divided in to a sequence of updates at each sensor node. By applying this idea, instead of transmitting the messages through all the sensor nodes, at each sensor node some information is used to update the estimation and determine which sensor in the neighbor of the current node is the best one regarding information amount (more details are in [6]).

The algorithm is initialized by sensor node *i*=1. It is assumed that an initial value of $\boldsymbol{\theta}$ is available at this sensor node, denoted by $\hat{\boldsymbol{\theta}}^{(0)}$.

At sensor node *i*, the following steps are performed:

- *Data receiving*: the sensor node *i* has been activated by the previous node and has received the transmitted required data.
- *New variable calculation*: Calculate the new matrix $\mathbf{H}_i(\hat{\boldsymbol{\theta}}^{(i-1)})$ as

$$H_i(\hat{\boldsymbol{\theta}}^{(i-1)})=\left.\frac{\partial\left[A_i(\boldsymbol{\theta})\mathbf{x}\right]}{\partial\boldsymbol{\theta}^T}\right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}^{(i-1)}}; \quad (12)$$

then, update the information utility matrix, Γ_i, as

$$\begin{aligned}\Gamma_i&=\Gamma_{i-1}\\&+H_i^T(\hat{\boldsymbol{\theta}}^{(i-1)})\Sigma_i^{-1}H_i(\hat{\boldsymbol{\theta}}^{(i-1)}).\end{aligned} \quad (13)$$

- *Estimation update*: Update and obtain the current estimate $\hat{\boldsymbol{\theta}}^{(i)}$ as

$$\begin{aligned}\hat{\boldsymbol{\theta}}^{(i)}&=\boldsymbol{\theta}^{(i-1)}\\&+\Gamma_i^{-1}H_i^T\left(\hat{\boldsymbol{\theta}}^{(i-1)}\right)\times\Sigma_i^{-1}\left[\mathbf{y}_i-A_i(\boldsymbol{\theta}^{(i-1)})\mathbf{x}\right].\end{aligned} \quad (14)$$

- *Estimation quality test*: Test the quality of the updated parameter, $\hat{\boldsymbol{\theta}}^{(i)}$, according to some performance measures, e.g., the trace or determinant of the covariance matrix in each sensor-node. If the estimation of parameter is "good enough," the estimation process is terminated; otherwise, the algorithm continues with the following steps.
- *Sensor node selection*: Select a sensor node from its neighbor according to the certain information

utility criteria. The selected node should provide the information that has the most potential to decrease the estimation uncertainty and increase the performance.

In [6], the Cramer-Rao bound is chose as information utility criteria and the estimation performance for two reasons: 1)The CRB is the lower bound on the variance of any unbiased estimators; it equals the inverse of the Fisher information matrix (FIM). Hence, the measures are calculated directly from the FIM. 2) The calculation of the FIM as a performance measure is an intrinsic part of the algorithm, *i.e.*, it can be obtained without any extra computation and data transmission. In [6] the recursive equation for FIM is obtained. By denoting the matrix $F_i(\boldsymbol{\theta})$ in the sequence $\{F_i(\boldsymbol{\theta}), i=1,2,...\}$ as the FIM of the parameter $\boldsymbol{\theta}$ when collect the measurements from the first i sensor nodes, *i.e.*, $\mathbf{y}=\{\mathbf{y_1},...,\mathbf{y_i}\}$, and according to the definition of FIM [11,12],

$$\begin{aligned} F_i(\boldsymbol{\theta}) &= -E\left[\frac{\partial^2 \ln p(\mathrm{y};\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^T}\right] \\ &= F_{i-1}(\boldsymbol{\theta}) - E\left[\frac{\partial^2 \ln p(\mathbf{y}_i;\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^T}\right] \end{aligned} \tag{15}$$

The second term on the Right Hand Side (RHS) of (16) can be calculated as [6]:

$$\begin{aligned} & -E\left[\frac{\partial^2 \ln p(\mathbf{y}_i;\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^T}\right] \\ &= E\left[\frac{\partial \ln p(\mathrm{y}_i;\boldsymbol{\theta})}{\partial\boldsymbol{\theta}}\frac{\partial \ln p(\mathbf{y}_i;\boldsymbol{\theta})}{\partial\boldsymbol{\theta}^T}\right] \\ &= H_i^T(\boldsymbol{\theta})\Sigma_i^{-1}H_i(\boldsymbol{\theta}) \end{aligned} \tag{16}$$

Where $H_i(\boldsymbol{\theta}) = \left[\frac{\partial[A_i(\boldsymbol{\theta})\mathrm{x}]}{\partial\theta_1} \cdots \frac{\partial[A_i(\boldsymbol{\theta})\mathrm{x}]}{\partial\theta_P}\right].$

Hence, the recursive equation for the FIM sequence is obtained as:

$$F_i(\boldsymbol{\theta}) = F_{i-1}(\boldsymbol{\theta}) + H_i^T(\boldsymbol{\theta})\Sigma_i^{-1}H_i(\boldsymbol{\theta}) \tag{17}$$

By comparing (17) and (13) it can be easily observed that the updating formulas for the matrix $F_i(\boldsymbol{\theta})$ and $\boldsymbol{\Gamma}_i$ are the same. This equivalence represents that using FIM as the performance measure will not increase the computation complexity and the required transmission.

Consequently, if the trace of the FIM matrix is used as the information measure, the information utility function for node selecting is:

$$I(l;\boldsymbol{\theta}) = Tr\left\{H_l^T(\boldsymbol{\theta})\Sigma_l^{-1}H_l(\boldsymbol{\theta})\right\}. \tag{18}$$

Then, the sensor node l that maximizes this information utility function is selected as the next sensor node i+1:

$$i+1 = \arg\max_{l\in S} I(l;\boldsymbol{\theta}) \tag{19}$$

- *Data transmission*: The current sensor node transmits the current estimate $\hat{\boldsymbol{\theta}}^{(i)}$ and $\boldsymbol{\Gamma}_i$ to the selected node and then the current node returns to sleeping status.

4. Modified Information-driven Collaborative Processing and Energy Efficient Algorithms

4.1. Modified Information-Driven Collaborative Processing

In wireless sensor networks, good collaboration among distributed sensor nodes can be very useful to improve estimation accuracy and keep the energy consumption in a reasonable level. In the following, we propose a new approach for estimation update motivated by the above distributed estimation, *i.e.*, IDCP. Here, we assume that S is the collection of the sensor nodes located in the neighbor area of the current sensor node i and $l \in S$;

$$S = \left\{l : \left\|\mathbf{r}_l - \mathbf{r}_i\right\| \le r_0\right\}, \tag{20}$$

Where $\mathbf{r}_i$, r_0 represent the location of the ith sensor node and the neighborhood radius (here sensing range) of the sensor nodes, respectively.

Through this approach, the sensor node i sends the update information, $\hat{\boldsymbol{\theta}}^{(i-1)}$, to all of its neighbors, then all these neighbors calculate the new private Jacobian matrix, $\mathbf{H}_l(\hat{\boldsymbol{\theta}}^{(i-1)})$, and information utility matrix, $\boldsymbol{\Gamma}_l$. Based on these calculated matrixes, each node obtains the $\hat{\boldsymbol{\theta}}^{(l)}$ as:

$$\begin{aligned} \hat{\boldsymbol{\theta}}_l^{(i)} &= \boldsymbol{\theta} + \Gamma_l^{-1} H_l^T\left(\hat{\boldsymbol{\theta}}^{(i-1)}\right) \\ &\quad \times \boldsymbol{\Sigma}_l^{-1}\left[\mathbf{y}_l - A_l(\hat{\boldsymbol{\theta}}^{(i-1)})\mathbf{x}\right]. \end{aligned} \tag{21}$$

Each node transmits $\hat{\boldsymbol{\theta}}_i^{(i)}$ to the ith node when this node also performs the step 2 and 3 of IDCP algorithm to obtain $\hat{\boldsymbol{\theta}}_i^{(i)}$. Consequently, the estimation is updated as follow:

$$\hat{\boldsymbol{\theta}}^{(i)} = \frac{1}{L}\sum_{l=1}^{L}\boldsymbol{\theta}_l^{(i)}. \tag{22}$$

Note that in (22) the estimated value of current node, $\hat{\boldsymbol{\theta}}_i^{(i)}$, is included because the current node is considered in its neighborhood set. In our new algorithm, Modified

IDCP (MIDCP), for estimation update, the information utility of all neighbor sensor nodes are used instead of one node and it can be reasonable to expect that the accuracy of the algorithm increases. Generally, since every possible value of the information utility in a sensor node's neighborhood is used, the node can estimate the location of the source more accurately in comparison with IDCP. The sensor selection scheme is as same as one in IDCP. However, one of the most important challenges in designing new algorithms for wireless sensor networks is energy consumption for signal processing. Therefore, it is necessary to consider energy consumption for new algorithms.

4.2. Energy Efficient Algorithm

In this paper we consider Rough order of Magnitude (ROM) model [13]. Technically, the model provides the energy required for a node to reach another node in one hop. Given l bits of data, the model states that the energy to transmit the data to a distance of d meters is:

$$E_t = l\varepsilon_{elec} + l\varepsilon_{amp} d^4 \tag{23}$$

and the energy to receive the data is

$$E_r = l\varepsilon_{elec} \tag{24}$$

The energy per bit to run the electronics such as the filters is represented by ε_{elec}, and the energy to run the power amplifier is presented by ε_{amp}. The values for these parameters are ε_{elec} = 50 nJ/bit and ε_{amp} = 0:0013 pJ/bit/m4. The model assumes $1/d^4$ for propagation loss in the channel because of the complicated transmission medium. It is predicted that the node antenna will be low to ground, possibly hidden under the grass. Finally, we set the packet size to be $l = 384$ bits to encompass the information required to perform distributed localization along with the overhead for error correcting and security [13]. It is important to note that the energy consumption in sensor nodes for computational tasks is much less than energy consumption for receiving and transmitting. As a result, we do not consider the energy amount used for computational processes.

It can be seen from (22) that the energy consumption in each node is related to distance between destination node and receiver node with the power of four. Therefore, one of the most efficient ways to reduce the energy consumption can be decrease the distance between transmitter and receiver nodes. This fact motivates us to design a new algorithm in which the neighborhood radius of the nodes decreases gradually. This new algorithm is called Energy Efficient MIDCP (EFMIDCP). The flowchart of EFMIDCP is shown in **Figure 1**.

As **Figure 1** shows, at first, the selected node receives the required parameters, $\hat{\boldsymbol{\theta}}^{(i-1)}$ and $F_{i-1}\left(\hat{\boldsymbol{\theta}}^{(i-1)}\right)$ from the previous node which has already updated the estimation. In fact, the inversion matrix of FIM is Cramer-Rao Bound (CRB) and it is a lower bound for unbiased estimators like ML. As a result, by calculating determinant of CRB the covariance error of the ML estimation is obtained at each iteration.

According to **Figure 1,** the next step is to decide whether the neighborhood radius is required to decrease or not. The Information utility of nodes located near the desired target is very high; therefore, saturation state of information is more likely to occur, *i.e.*, change in amount of energy is trivial from one node to another. Consequently, it can be expected that the performance of estimation is not change dramatically by reducing the number of nodes in this area.

It can lead us to decreasing the neighborhood radius for update estimation to abate energy consumption. Based on the above definitions, we use predefined thresholds and their correspondence neighborhood radius to decrease this radius at each iteration. According to (22), by applying this algorithm, the energy consumption decreases due to elimination of transmitting the data to further distance.

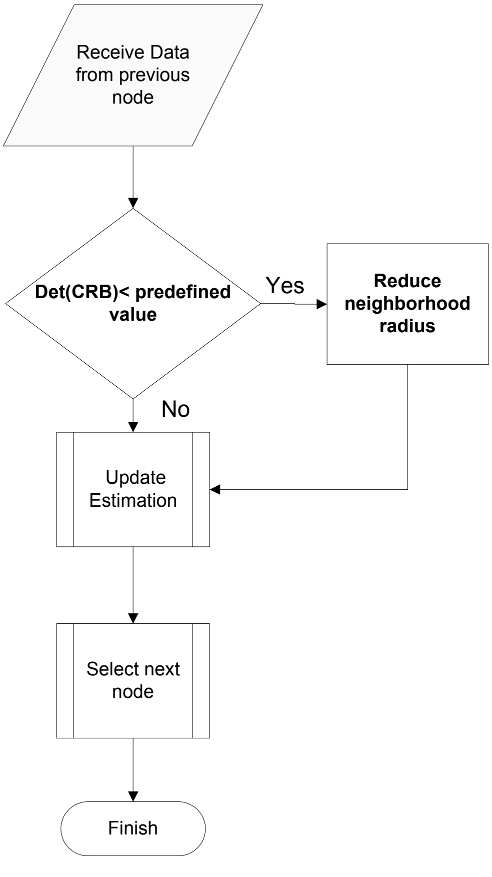

Receive Data from previous node
Det(CRB)< predefined value
Yes
Reduce neighborhood radius
No
Update Estimation
Select next node
Finish

Figure 1. Flowchart of MIDCP algorithm.

5. New Method for Initial Target Location Estimation

All iterative algorithms, such as Gauss-Newton method, are sensitive to the initial point of estimation. For better conception, a sample function for the estimation error is shown in **Figure 2**. It can be seen that many sub-optimum points exist in error function. Thus, if the initial point is not estimated accurately, the algorithm requires much iteration to diverge; even in some cases the divergence is probable.

The method that we will propose in this section needs at least three sensor nodes. As we mentioned before, the distribution model for a diffusive source is:

$$c(\mathbf{r},t)=\frac{\mu}{4\pi k\left|\mathbf{r}-\mathbf{r}_0\right|}erf\left(\frac{\left|\mathbf{r}-\mathbf{r}_0\right|}{2\sqrt{k\left(t-t_0\right)}}\right) \tag{25}$$

Where $erf\left(x\right)=\frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}dt$ If the observation of one node can satisfy $t-t_0 \gg \left(\left|\mathbf{r}_i-\mathbf{r}_0\right|^2\right)/4k$, the *erf* function is approximately one and with renunciation of noise, the observation of this node is equal to:

$$y\left(\mathbf{r}_i,\infty\right)\approx\frac{\mu}{4k\pi\left|\mathbf{r}_i-\mathbf{r}_0\right|}+b \tag{26}$$

Therefore,

$$\begin{aligned}&\frac{\mathrm{y}\left(\mathbf{r}_i,\infty\right)-\mathrm{y}\left(\mathbf{r}_1,\infty\right)}{\mathrm{y}\left(\mathbf{r}_2,\infty\right)-\mathrm{y}\left(\mathbf{r}_1,\infty\right)}\\&\approx\frac{\left|\mathbf{r}_i-\mathbf{r}_0\right|^{-1}-\left|\mathbf{r}_1-\mathbf{r}_0\right|^{-1}}{\left|\mathbf{r}_2-\mathbf{r}_0\right|^{-1}-\left|\mathbf{r}_1-\mathbf{r}_0\right|^{-1}}\quad 3\le i\le n\end{aligned} \tag{27}$$

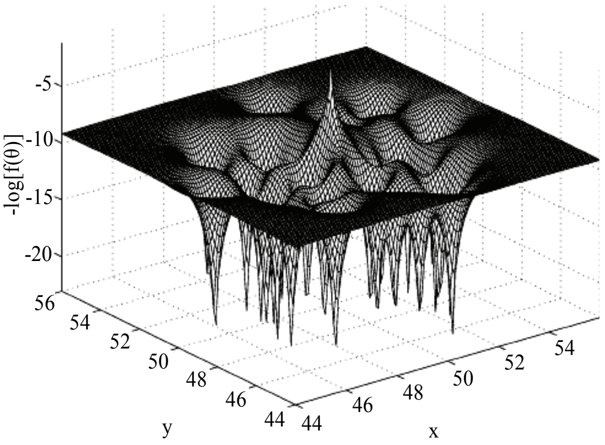


Figure 2. Logarithm of the error function of a distributed ML algorithm [14].

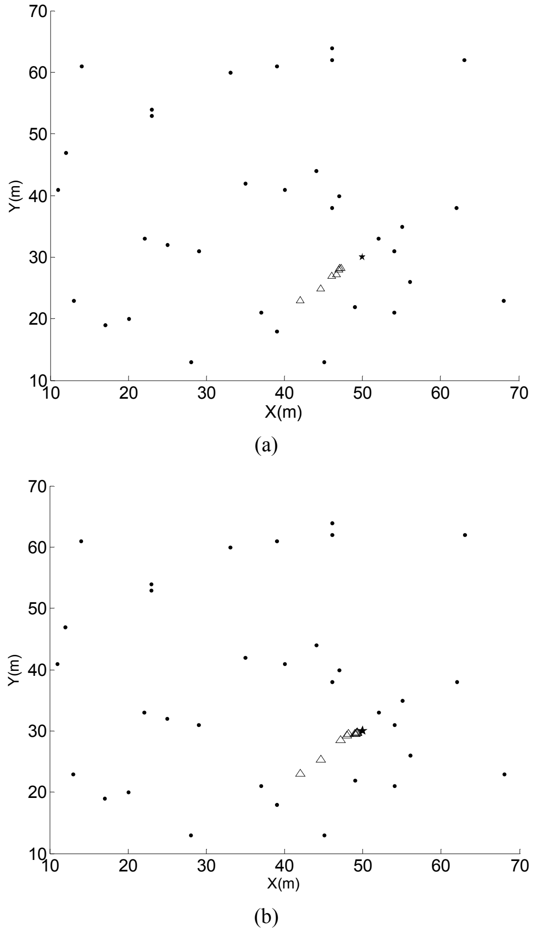
If n = 4, two unknown variable, $\mathbf{r}_0=\left(x_0,y_0\right)$, exist and they are obtained by using numerical methods to solve the set of equations and if n > 4 these are obtained by solving set of equations and then averaging over the results. In our proposed algorithm, we choose one node arbitrary and this node is considered as the central processor. It gathers all observation of its neighbor nodes and solves (27) to achieve the initial point for estimation. In addition, this node is the first node in our sequence of selected node.

6. Numerical Examples

This section presents a simulation of a sensor network with M = 100 sensor nodes, distributed randomly in a 100? 00m field. At each sensor, the measurement of the diffusive source is generated according to (5). Here, source position $\mathbf{r}_0=(x_0,y_0)$ is the unknown parameter we need to estimate. We define a pair of neighbor sensor nodes whose distance is less than 20m, *i.e.*, the communication range (neighborhood radius) is 20 m. For the diffusion model, we consider the environment as homogenous semi-infinite with an impermeable boundary which can represent dispersion in air above the ground. We use a scenario of a stationary impulse source located at $\mathbf{r}_0=(50,30)$ meter. The bias term in (5) is $b=10^{-5}$ g/m^2, and the noise standard deviation is $\sigma=6\times10^{-6}$ g/m^2. We take 10 temporal samples at each sensor node with a sampling interval of 5 seconds. The other parameters μ , k, and t_1 are taken to be 1 g/s, 20 m^2/s and 0 s, respectively.

In **Figure 3,** the trajectory of target estimation for IDCP and MIDCP is shown. The trajectory is depicted with 10 iterations of both algorithms. This figure is the result of averaging over 200 independent experiments in which the location of sensor nodes and target are assumed constant but the observations and measurement noise of each sensor changes. In addition, the initial value for estimation is obtained based on our proposed algorithm in Section 5. In this figure, the "star" denotes the exact location of the desired target and "triangles" denote estimated location at each iteration. We observe that although the initial bias is relatively high, the trajectory nearly reach to the exact source location in MIDCP. This means the algorithm can converge accurately and nearly in comparison with IDCP.

For better conception, we compare bias estimation and covariance error of these two algorithms in the following. In **Figures 4 and 5** we compare the estimation bias $\sqrt{\left\|\theta-\hat{\theta}\right\|^2}$ of $\boldsymbol{\theta}$ and the logarithm of determinant of CRB for IDCP and MIDCP with respect to the number of iterations. These figures are the result of averaging over 200 independent experiments in which the location of the sensor nodes and target are constant, but the observations and measurement noise of each sensor change.



(a)

(b)

Figure 3. Trajectory of target estimation in a square area with 100 sensor nodes randomly placed for a) IDCP and b) MIDCP.

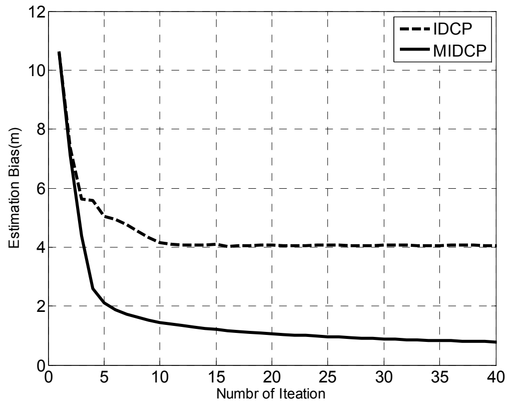


Figure 4. Estimation bias versus number of iterations for IDCP and MIDCP algorithms.

According to these two above figures, we find that the estimation performance of MIDCP is much better than IDCP and this means that using all information utilities of neighbor sensors can greatly improve the accuracy of the estimation.

Before we discuss the simulation results for EF-MIDCP, we provide the estimation bias and energy consumption of MIDCP with the different neighborhood radiuses. At first, we calculate the energy consumption for neighborhood radiuses equal to 5, 10, 15, 20 for MIDCP based on (23) and (24). The experiment conditions are similar to the previous one. The results are shown in **Table 1**.

As can be seen, by increasing the neighborhood radius (communication range) the energy consumption increase and the bias estimation decreases. Therefore, it is a tradeoff between the accuracy of the algorithm and the bias estimation. It motivates us to decrease the energy consumption without dramatic change in the accuracy of the algorithm; we decrease the neighborhood range with regard to the error covariance of MIDCP calculated at each iteration. (See Section 4.)

In the second example, we compare the performance of MIDCP with EFMIDCP. In **Figures 6 and 7** the estimation bias $\sqrt{\left\|\theta-\hat{\theta}\right\|^2}$ of $\boldsymbol{\theta}$ and logarithm of determinant of CRB with respect to the number of iterations are shown. These figures are the result of averaging over 200 independent experiments in which the location of sensor

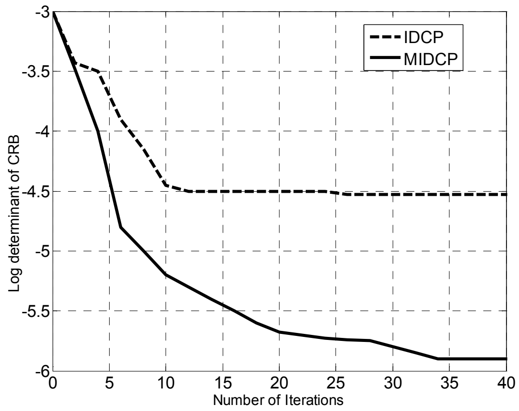


Figure 5. Log determinant of CRB versus number of iterations for IDCP and MIDCP algorithms.

Table 1. Energy consumption and estimation bias for MIDCP with different neighborhood radiuses.

Neighborhood radius	5	10	15	20
Bias Estimation(m)	5.25	0.85	0.88	0.67
Energy Used (μJ)	0.14	1.56	4.9	24.38

nodes and the target are assumed constant but the observations and measurement noise of each sensor change. Other conditions are as same as our first example. In this scenario, we change the neighborhood radius with the following pattern:

- If $\log(|CRB|) \geq -4.5$ then the neighborhood radius is set to 20.
- If $-5 \leq \log(|CRB|) \leq -4.5$ then the neighborhood radius is set to 15.
- If $\log(|CRB|) \leq -5$ then the neighborhood radius is set to 10.

According to these two figures, we find that the estimation performance of MIDCP is somewhat better than EFMIDCP. In addition, in this example we compare energy consumption of these tow algorithms and the result is shown in **Table 2**. As it shows, the energy consumption decreases dramatically and it can be observed that the EFMIDCP reduces the energy consumption by half while duplicates the accuracy of the algorithm.

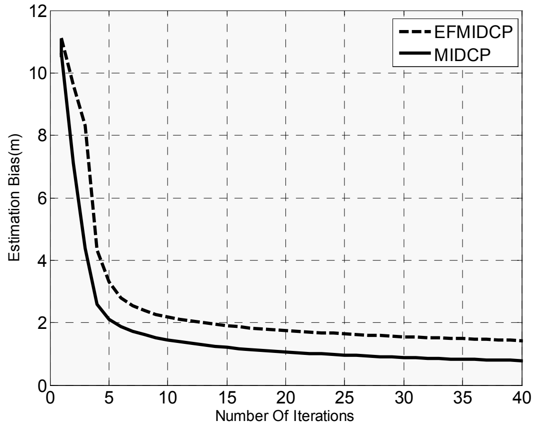


Figure 6. Estimation bias versus number of iterations for MIDCP and EFIDCP algorithms.

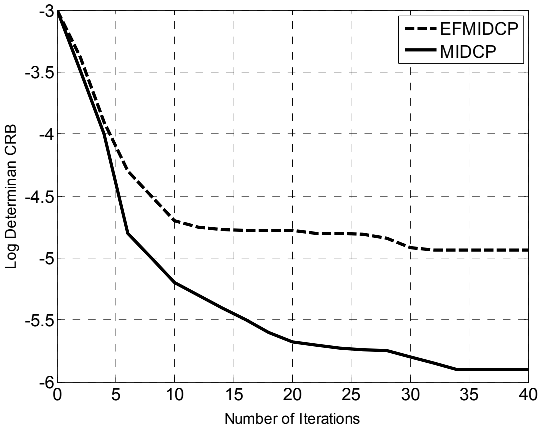


Figure 7. Log determinant of CRB versus number of iterations for MIDCP and EFMIDCP algorithms.

Table2. The energy and bias estimation for MIDCP and EFMIDCP.

	MIDCP	EFMIDCP
Bias Estimation (m)	0.67	1.45
Energy Used (μJ)	24.38	13.05

7. Conclusions

In this paper, we addressed the problem of developing an accurate and energy efficient distributed algorithm for a diffusive source location estimation in wireless sensor networks. At first, we defined the neighborhood radius that was equal to the communication range of the sensor nodes. Then, we used the information of every node located within the neighborhood circle. This method improved the accuracy of estimation. However, based on ROM model, we observed that the energy consumption in our new algorithm was somewhat high. As our primate goal was to reduce energy consumption and increase the accuracy, we proposed another algorithm in which the neighborhood range decreased gradually based on the covariance error of the estimation. In addition, we proposed a new algorithm for the initial point calculation which was necessary for iterative methods like Gauss-Newton method. Finally, we used numerical examples to compare the performance of our proposed methods and found that EFMIDCP was the most suitable algorithm to apply in wireless sensor networks when we considered both energy consumption and the accuracy of the estimation. In future work, a mathematical relationship between the neighborhood radius and the number of covariance error can be derived; therefore the neighborhood radius can be changed adaptively instead of changing it with regard to predefined value. In addition, the distributed implementation of other recursive methods such as Newton method can be derived.

8. Acknowledgements

This research has been supported by Iran Telecommunication Research Center, Tehran, Iran, which is appreciated.

5. References

[1] T. Zhao and A. Nehorai, “Detecting and Estimating Biochemical Dispersion of a Moving Source in a Semi-Infinite Medium,” *IEEE Transactions on Signal Processing*, Vol. 54, No. 6, June 2006, pp. 2213-2225.

[2] X. Sheng and Y. H. Hu, “Maximum Likelihood Multiple-Source Localization Using Acoustic Energy Measurements with Wireless Sensor Networks,” *IEEE Transac-*

tions on Signal Processing, Vol. 53, No. 1, January 2005, pp. 44-53.

[3] A. Pardo, S. Marco and J. Samitier, "Nonlinear Inverse Dynamic Models of Gas Sensing Systems Based on Chemical Sensor Arrays for Quantitative Measurements," *IEEE Transactions on Instrumentation and Measurement*, Vol. 47, No. 3, June 1998, pp. 644-651.

[4] H. Ishida, T. Nakamoto and T. Moriizumi, "Remote sensing and localization of gas/odor source and distribution using mobile sensing system," *Proceedings of the International Conference on Solid-State Sensors and Actuators*, Chicago, Vol. 1, June 1997, pp. 16-19.

[5] A. Nehorai, B. Porat and E. Paldi, "Detection and Localization of Vaporemitting Sources," *IEEE Transactions on Signal Processing*, Vol. 43, No. 1, January 1995, pp. 243-253.

[6] T. Zhao and A. Nehorai, "Information-Driven Distributed Maximum Likelihood Estimation Based on Gauss-Newton Method in Wireless Sensor Networks," *IEEE Transactions on Signal Processing,* Vol. 55, No. 9, September 2007, pp. 4669-4682.

[7] A. T. Ihler, J. W. Fisher, R. L. Moses and A. S. Willsky, "Nonparametric Belief Propagation for Self-Localization of Sensor Networks," *IEEE Journal on Selected Areas in Communications*, Vol. 23, No. 4, April 2005, pp. 809-819.

[8] J. L. Williams, J. W. Fisher and A. S. Willsky, "Optimization Approaches to Dynamic Routing of Measurements and Models in a Sensor Network Object Tracking Problem," *IEEE International Conference on Acoustics*, *Speech*, *and Signal Processing* (*ICASSP*), Philadelphia, Vol. 5, March 2005, pp. 1061-1064.

[9] J. Liu, J. Reich and F. Zhao, "Collaborative In-Network Processing for Target Tracking," *EURASIP Journal on Applied Signal Processing*, Vol. 2003, No. 4, 2003, pp. 378-391.

[10] M. G. Rabbat and R. D. Nowak, "Decentralized Source Localization and Tracking," *IEEE International Conference on Acoustics*, *Speech*, *and Signal Processing* (*ICASSP*), Montreal, May 2004, pp. 921-924.

[11] P. Stoica and A. Nehorai, "MUSIC, Maximum Likelihood, and Cramer-Rao Bound," *IEEE Transactions on Acoustics Speech and Signal Processing*, Vol. 37, No. 5, 1989, pp. 720-741.

[12] S. F. Yau and Y. Bresler, "A Compact Cramer-Rao Bound Expression for Parametric Estimation of Superimposed Signals," *IEEE Transactions on Signal Processing*, Vol. 40, No. 5, 1992, pp. 1226-1230.

[13] L. Kaplan, "Global Node Selection for Localization in a Distributed Sensor Network," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 42, No. 1, January 2006, pp. 113-135.

[14] D. Blatt and A. O. Hero, "Energy-Based Sensor Network Source Localization via Projection onto Convex Sets," *IEEE Transactions on Signal Processing*, Vol. 54, No. 9, September 2006, pp. 3614-3619.