Control and Optimization in Applied Mathematics (COAM)
DOI. 10.30473/coam.2020.50322.1131
Vol. 5, No. 1, Winter-Spring 2020 (81-101), ©2016 Payame Noor University, Iran

Research Article

An Efficient Data Collection Algorithm to Estimate Unknown Target Parameter in Wireless Sensor Networks

Amir Hosein Mohajerzadeh[1], Abbas Ali Rezaee[2*], Morteza Bigdeli[3]

[1]Computer Department, Ferdowsi University of Mashhad, Mashhad, Iran
[2]Department of Computer Engineering and Information Technology, Payame Noor University, Tehran, Iran.
[3]Electrical Engineering Department, Ferdowsi University of Mashhad, Mashhad, Iran.

Received: December 18, 2019; **Accepted:** January 8, 2021.

Abstract. Estimating the target parameter while the prior distribution function is known, and several observations which are provided by the sensor node is the main goal in this paper. In wireless sensor networks (WSN), nodes sense the environment and send data to a sink node called Fusion Center (FC). FC collects data and estimates the observed parameter with user-defined precision. The proposed algorithm increases network lifetime and has an efficient estimation process. For this purpose, the proposed algorithm schedules node's activity and determines the multihop path between nodes and FC. Simulation and performance analysis demonstrates proposed algorithm fulfills its goals.

Keywords. Estimation, Optimization problem, Random variable, Scheduling, Wireless sensor networks.

MSC. 90C26.

* Corresponding author
mohajerzadeh@um.ac.ir, a_rezaee@pnu.ac.ir, morteza_bigdeli2003@yahoo.com
http://mathco.journals.pnu.ac.ir

1 Introduction

Wireless Sensor Network (WSN) is a network that has several geographically distributed sensor nodes that sense events in an environment. The sensor node's energy, computational power, and storage capacity are limited [13]. In the recent decade, WSN has developed in many applications and environments such as monitoring, healthcare, home automation, etc. [14]. Recently WSN also is used in distributed estimation, detection, and tracking.

In [14], a distributed estimation of an unknown target parameter by a set of sensed data is performed in a distributed sensor node environment. For simplicity, we call it the parameter. In this environment, each node directly or indirectly sends its data from the environment to a central FC. FC constructs the underlying physical phenomenon according to sensors data. There are much kinds of research about estimation in computer networks [1, 18] and in WSN [2, 34]. FC process received data and estimate the parameters. In many estimation types of research WSN [3, 19], they suppose that data sending is without any distortion. In [4], different distributed estimation algorithms reviewed [34, 26]. The decentralized estimation has been introduced in distributed control [6], target tracking [31] and data fusion [7]. In [35], an optimal power scheduling problem is proposed, is used in an inhomogeneous sensor network for a noise-corrupted deterministic signal. This algorithm quantizes the power level to minimize the total data sending energy consumption with Mean Square Error (MSE) performance. The [20] minimizes the estimated MSE with an optimal tradeoff between the number of alive nodes and the quantization bit rate. The paper [12] estimate a parameter variable in a bidimensional scenario in WSN. In this work, a mathematical framework is analyzed. In [8], the energy consumption performance is reviewed in WSN distributed estimation. This paper estimates the parameter with the Best Linear Unbiased Estimation (BLUE) method. The [8, 21, 8] use an optimization method to solve the problem. The [21] explicitly considers network lifetime techniques and estimation precision in distributed estimation problems. This method has an estimation model using confidence interval is explained, which uses the user's required precision as an input. This model estimates the parameter with a defined precision (based on user-defined precision using) using the confidence interval method. It also increases network lifetime. The scheduling consists of always actual, random on-off, adaptive on-off, and periodic on-off states [9]. The proposed algorithm is an adaptive on-off scheduling algorithm in which FC creates a scheduling method, and other nodes use that method [28]. We consider hierarchical (intra- and inter-cluster routing) sensor networks, and the scheduling and routing algorithms for each cluster independently. Therefore, using a centralized algorithm in each cluster (knowing the fact the cluster area is limited) is applicable and efficient. In each cluster, Intra-cluster routing sends sensor data to the cluster head (CH). The CH then communicates with other CH to route data to the sink node. Hear single-hop is applicable for routing between clusters. The proposed algorithm schedules nodes' activity with a nonlinear programming (NLP) method to send data from a node to FC by multi-hop routes. Some of the researchers perform scheduling nodes' tasks in the MAC layer [22, 32]. The [16] reviews different techniques using simulation in a many-to-one communication paradigm. It minimizes the number of time slots required to complete a converge cast using a single frequency scheduling method. Some works are application-based. For example, in [27], a scheduling and routing algorithm guarantee the end-to-end delay. The [17] designs a lifetime-aware routing and coverage aware algorithm. The [10] is using a routing and scheduling algorithm in mesh sensor networks. This paper maximizes the lifetime of a WSN and guarantees the end-to-end delay. The other studies worked on routing and scheduling separately. They optimized just one [33].

In this paragraph, we emphasize the novelties of this paper. This paper emphasizes joint routine and scheduling algorithms. It is important to provide both routine and schedule at the same time because efficiency depends on it. Moreover, having multi-level routing helps us to provide more efficient energy consumption in nodes. We have used queuing theory to estimate delays in scheduling algorithms. In this paper, we proposed a hybrid hierarchical system with routing and scheduling for WSN. Regarding our last papers emphasized by the reviewer, it is worth mentioning that in this work, we have an optimization model efficiently adapted to the characteristics of the WSNs. By taking all similar papers in this field, including our last papers, into account, in the following, we have listed the main contributions:

- Adopted to tier hierarchical routing to the delay while the scheduling is optimized simultaneously.
- Unique confidence interval based error refinement based on HPD interval form.
- Considering both energy and delay in a mathematical model to have intra-cluster joint routing and scheduling.
- Having queue theory in delay provisioning.
- Having proposed a new solution for the proposed NLP regarding the running time.

We performed routing and scheduling algorithms jointly to achieve the highest efficiency. Also, the proposed algorithm is highly compatible with estimation process data, which is not well studied in literature before. Section 2 introduces the proposed algorithm to estimate a random variable parameter. In Section 3, proposed joint routing and scheduling algorithms are discussed in detail. The Sensors monitor the network and send their data to the CH, which estimates the parameter using the model described in Section 2. Our intra cluster routing and scheduling algorithm increases lifetime during routing from nodes to CH. The CH then estimates the required parameter and sends the results to the sink node. In Section 4, the performance of the proposed algorithm is evaluated against others. Section 5 has a conclusion.

2 Problem Statement

We define the hierarchical wireless sensor network topology consisting of different clusters. A cluster is shown in Figure 1. There are N sensor nodes and a CH in each

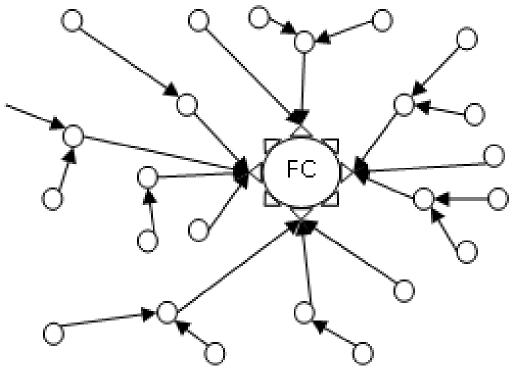

Figure 1: Sensor nodes as a cluster.

cluster to estimate the required parameter $\propto$. At first, each cluster node monitors the cluster environment and then sends events to the fusion center CH. At last, FC makes its estimation according to all the received data from cluster nodes.

When cluster nodes send data to FC, there are two main challenges:

1. how many packets should cluster nodes send to FC (section 3.1) to estimate the parameter $\propto$
2. in order to send cluster nodes data to FC, each node should select the best route and relay nodes (section 3.2).

During environment monitoring, the event data may damage by some additive noise:

$$x_{ki} = \propto + \varepsilon_{ki} \qquad k = 1, 2, \cdots, N, \quad i = 1, 2, \cdots, n_k. \tag{1}$$

In this relation x_{ki} is the i^{th} data of node k. Each node sends its data to FC Node k provide the n_k data size. N is the sensor network nodes. $\propto$ is the parameter monitored by the nodes. FC tries to estimate $\propto$ with the least possible error. The environment has noise, so a noise ε_{ki} is added to the parameter $\propto$, which is a random variable. Most of the applications are compatible with the random variable the parameter [25, 29]. The noise variables in the sensors,ε_{ki} , is independent, and the mean zero Gaussian random variable is $var(\varepsilon_{ki}) = \sigma_k^2$ $(k = 1, 2, \cdots, N)$. So we have $x_{ik} \sim N(\partial, \sigma_k^2)$, in which X_i have Gaussian distribution with mean ∂ and variance σ_k^2. In the estimation process (because of bandwidth and energy limitations of sensors), each node at first quantizes the event analog data locally y_k into a discrete message $m_k = Q_k(y_k)$ of length L_k bits [1] in which $Q_k(y_k)$ is quantization function.

In the scheduling algorithm, the cluster head determines all the parameters such as the number of sending data, the specified route, sending time, and the state of each node (ON or OFF). Our goal is to propose a scheduling algorithm to manage cluster nodes on or off activities and increase network lifetime and decrease estimation error under the desired bound.

2.1 Estimation process

Cluster nodes send data to FC to estimate the parameter $\propto$. The more the data, the more accuracy, and lower error. We use the Bayesian method to estimate $\propto$. We use the Normal distribution function with mean μ and variance τ^2 for the density function of $\propto$ random variable. Nowadays, most of the variables behavior is like Normal distribution function (such as temperature, humidity, etc.). They can be considered in future works. The density function of $\propto$ is as follow:

$$\pi(\propto)=\frac{1}{\sqrt{2\pi\tau^2}}e^{-\frac{1}{2\tau^2}(\propto-\mu)^2} \tag{2}$$

If x is an independent random variable with normal distribution function, it's joint density function is as follows:

$$f(x\mid\propto)=\prod_{i=1}^{n} f(x_i\mid\propto)=\prod_{i=1}^{n}(2\pi{\sigma_i}^2)^{-\frac{1}{2}}.e^{-\frac{1}{2}\sum_{i=1}^{n}\left(\frac{x_i-\propto}{\sigma_i}\right)^2}. \tag{3}$$

To the equations (2) and (3), the joint density of $\propto$ and x is achieved using the equation (4):

$$\pi(\propto,x)=\frac{\prod_{i=1}^{n}\left(2\pi\sigma_i^2\right)^{-\frac{1}{2}}}{\sqrt{2\pi\tau^2}}e^{-\frac{1}{2}\left\{\sum_{i=1}^{n}\left(\frac{x_i-\propto}{\sigma_i}\right)^2+\frac{1}{\tau^2}(\propto-\mu)^2\right\}}. \tag{4}$$

By expanding the equation (4) we have:

$$\pi(\propto,\underline{x})=\frac{\prod_{i=1}^{n}\left(2\pi\sigma_i^2\right)^{-\frac{1}{2}}}{\sqrt{2\pi\tau^2}}e^{-\frac{1}{2}\left\{\sum_{i=1}^{n}\left(\frac{1}{\sigma_i^2}\left(x_i^2+\propto^2-2\propto x_i\right)\right)+\frac{1}{\tau^2}(\propto^2+\mu^2-2\mu\propto)\right\}}. \tag{5}$$

To achieve the posterior distribution function, the equation (5) can be changed as the following form:

$$\pi(\propto,\underline{x})=\frac{\prod_{i=1}^{n}\left(2\pi\sigma_i^2\right)^{-\frac{1}{2}}}{\sqrt{2\pi\tau^2}}e^{-\frac{1}{2}\left\{\propto^2\left(\sum_{i=1}^{n}\frac{1}{\sigma_i^2}+\frac{1}{\tau^2}\right)-2\propto\left(\sum_{i=1}^{n}\frac{x_i}{\sigma_i^2}+\frac{\mu}{\tau^2}\right)\right\}\times C}. \tag{6}$$

In the above equation, the terms which are not related to $\propto$ are represented by a constant coefficient C. This equation is a normal density function. Conditional density of $\propto$ is $\propto|\underline{x}\sim N(\acute{\mu},\acute{\sigma}^2)$, where $\acute{\sigma}^2$ and $\acute{\mu}$ are as follows:

$$\frac{1}{\acute{\sigma}^2}=\sum_{i=1}^{n}\frac{1}{\sigma_i^2}+\frac{1}{\tau^2}\quad =>\acute{\sigma}^2=1/\left(\sum_{i=1}^{n}\frac{1}{\sigma_i^2}+\frac{1}{\tau^2}\right), \tag{7a}$$

$$\frac{\acute{\mu}}{\acute{\sigma}^2}=\sum_{i=1}^{n}\frac{x_i}{\sigma_i^2}+\frac{\mu}{\tau^2}\quad =>\acute{\mu}=\left(\sum_{i=1}^{n}\frac{x_i}{\sigma_i^2}+\frac{\mu}{\tau^2}\right)/\left(\sum_{i=1}^{n}\frac{1}{\sigma_i^2}+\frac{1}{\tau^2}\right). \tag{7b}$$

The distribution function of $\propto|\underline{x}$ is normal and Bayesian estimator of the parameter $\propto$ is $\acute{\mu}$. So, the equation (7b) shows the final estimator [23].

We find the optimal sample number by calculating $(1-\alpha)\%$ with the quantity method

[24]. By considering $Q(\propto)=\frac{\propto-\acute{\mu}}{\acute{\sigma}}$ as pivotal quantity, the credible interval is given by the equation (8). Since our posterior density is unimodal, this credible interval is also an HPD interval form [24]:

$$P\left(\left|\frac{\propto-\acute{\mu}}{\acute{\sigma}}\right|<z_{\frac{\alpha}{2}}\right)>1-\alpha. \tag{8}$$

Here $z_{\frac{\alpha}{2}}$ is the $\left(\frac{\alpha}{2}\right)_{th}$ quantiles of standard normal distribution. By $(1-\alpha)\%$ credible interval and maximum acceptable error η, using the equation (9), data size is calculated:

$$z_{\frac{\alpha}{2}}\acute{\sigma}(n)<\eta. \tag{9}$$

Considering the equation (7a), the value of $\acute{\sigma}(n)$ is calculated as:

$$\acute{\sigma}(n)=\sqrt{1/\left(\sum_{i=1}^{n}\frac{1}{\sigma_i^2}+\frac{1}{\tau^2}\right)}. \tag{10}$$

Substituting the equation (10) in (9), we have:

$${z_{\frac{\alpha}{2}}}^2\left\{1/\left(\sum_{i=1}^{n}\frac{1}{\sigma_i^2}+\frac{1}{\tau^2}\right)\right\}<\eta^2. \tag{11}$$

By extending the equation (11), we get:

$$\sum_{i=1}^{n}\frac{1}{\sigma_i^2}>\frac{{z_{\frac{\alpha}{2}}}^2}{\eta^2}-\frac{1}{\tau^2}. \tag{12}$$

Based on the equation (12), data size (n) is given by the equation (13):

$$n=\left\{min(n)\mid\sum_{i=1}^{n}\frac{1}{\sigma_i^2}>\frac{{z_{\frac{\alpha}{2}}}^2}{\eta^2}-\frac{1}{\tau^2}\right\}. \tag{13}$$

To simplify the formula of n we consider the worse situation as $\dot{\sigma}^2=max\left(\sigma_i^2\right)\mid i\in\{1,\ldots,N\}$. Therefore by replacing $\sum_{i=1}^{n}\frac{1}{\sigma_i^2}$ with $\frac{n}{\dot{\sigma}^2}$ in the equation (13), the optimal n can be calculated as follows:

$$n=\left\lceil\dot{\sigma}^2\left\{\frac{{z_{\frac{\alpha}{2}}}^2}{\eta^2}-\frac{1}{\tau^2}\right\}\right\rceil \tag{14}$$

As observable in the equation (14), data size is easily calculable when the user determines parameters α and η. The number of samples which are needed in order to achieve desired precision based on parameters α and η is known.

3 The Proposed Routing and Scheduling Program

There are many routing protocols to send cluster data to CH in WSNs. As the radio range in WSN is limited direct and indirect routing may be used, depending on network conditions. We should know that direct transmission consumes more energy with less delay in comparison with indirect transmission, averagely.

Data size is determined based on the terms mentioned in section 2.1, to the application's required precision. The proposed algorithm regards two following challenges related to data transmission inside the cluster:

1. The route that cluster nodes can send data to FC.
2. Scheduling nodes activities and determining the way each node cooperates in the data gathering process.

In this section, methods for implementing the above issues are proposed.

3.1 Data transmission method inside the cluster

Wireless sensor network node consumes its energy for various reasons, to receive data, data collection, and data processing. Depending on the type of radio receiver, nodes have different energy consumption levels. However, in most WSNs, the same value has been considered for all network nodes. The volume of information that must be processed affects the node energy consumption. Similar to traditional networks, energy consumption due to data processing compared to other factors are negligible. Communication is the main factor of energy consumption in wireless sensor network nodes. Communication's energy consumption depends on several factors which are presented in the equation (15).

$$P_r/P_t = G_t G_r (C/4\pi d)^2. \tag{15}$$

We have studied different types of energy consumption model, and the model described in the equation (15) is the most suitable one. P_r is the signal strength received at the receiver side; P_t is the power of the signal at the transmitter side, G_t is transmitter antenna gain, G_r is receiver antenna gain strength, C is light speed, and d is the distance between sender and receiver. Note that the received signal should be greater than a specified threshold, thus based on the equation (16a), transmission energy consumption at the sender is calculated The G_r and G_T are determined according to node's characteristics, so they are constant (also C is constant). The value of d varies depending on the distance between the sender and receiver. We use the equation (16b) in order to determine transmission power. In (16b), we have: $\alpha \sim \left(G_t G_r (C/4\pi)^2\right)$.

$$E_t = F(G_t G_r (c/4\pi d)^2), \tag{16a}$$

$$E_t \sim \alpha/d^2 = F(\frac{1}{d^2}). \tag{16b}$$

Delay is also considered in the cost function. The Delay in the network is consists of propagation delay and queuing delay. The propagation delay in WSNs is minimal and can be ignored. So end-to-end delay only depends only on queuing Delay. We use $M/M/1/K$ for the nodes queue model. Modeling the delay by queuing theory is acceptable when the traffic is almost constant. We consider monitoring applications that produce data with a constant rate. By considering the IEEE 802.15.4 as MAC layer protocol, it seems that the $M/M/1/K$ would be efficient enough for the problem [30].

In $M/M/1/K$, the input process is Poisson; the service rate is exponential, one server, and system capacity is K. Queuing delay is calculated as (17):

$$W = L/\acute{\lambda}. \tag{17}$$

The parameter L is the average queue length and $\acute{\lambda}$ is packets arrival rate. These are calculated with (18) [15] and (19) equations. In the equation (18), λ is the average input rate and ρ is the data density: $\rho = \lambda/\mu$,

$$L = (\rho/(1-\rho)) - \left((K+1)\rho^{K+1}/(1-\rho^{K+1})\right) = \dot{F}(\lambda,\mu,K), \tag{18}$$

$$\acute{\lambda} = \lambda(1-P_K), \tag{19}$$

P_K is, the probability of K packets in the queue and, is calculated as:

$$P_k = \begin{cases} ((1-\rho)\rho^k/(1-\rho^{K+1})) & (\rho \neq 1) \\ ((1/K+1)\rho^k) & (\rho = 1) \end{cases} \tag{20}$$

In direct routing, the cluster nodes transmit data to FC in one hop (some of the experts do not consider direct forwarding as a routing method, but in this text, we call direct forwarding as direct routing), and an indirect routing, cluster nodes send their data to FC in multihop routing. So the end-to-end delay will be the sum of intermediate nodes queuing delay. Figure 2, shows data routing inside the cluster.

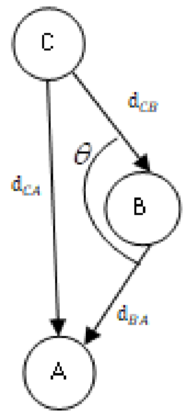

Figure 2: Data routing with three nodes.

In Figure 2, node A is the receiver, B is the relay, and C is the transmitter. Node C sends data to A directly on link CA with length D_{CA}. If node C sends its data to A using intermediate node B with Links CB and BA, it is indirect routing, Two scenarios are considered, direct (S1) and indirect (S2) transmission. The energy consumption for direct and indirect transmission is as the equations (21a) and (21b), respectively.

$$E_{s1} = F\left(d_{CA}^2\right) = \alpha d_{CA}^2, \tag{21a}$$

$$E_{s2} = F\left(d_{CB}^2\right) + F\left(d_{BA}^2\right) = \alpha\left(d_{CB}^2 + d_{BA}^2\right). \tag{21b}$$

In the above equation, function F exists in the equation (16a). The queuing model $M/M/1/K$ is used in all nodes A, B and C, so the end-to-end delay for direct and indirect routing is calculated as follow:

$$W_{s1} = L_A/\acute{\lambda}_A, \tag{22a}$$

$$W_{s2} = L_A/\acute{\lambda}_A + L_B/\acute{\lambda}_B, \tag{22b}$$

where L_A and L_B are average queue length of nodes A and B respectively. The $\acute{\lambda}_A$ and $\acute{\lambda}_B$ are actual input rate of nodes A and B respectively. L is as in the equation (18). We have:

$$W_{s1} = \dot{F}(\rho_A, K) = \dot{F}(\lambda_A, \mu_A, K), \tag{23a}$$

$$W_{s2} = \dot{F}(\rho_A, K) + \dot{F}(\rho_B, K) = \dot{F}(\lambda_A, \mu_A, K) + \dot{F}(\lambda_B, \mu_B, K). \tag{23b}$$

We suppose, all the network nodes are homogeneous, and are the same. In the equations (23a) and (23b), the μ_A and the μ_B are the service rate of nodes A and B. The μ_A and the μ_B are as characteristics of nodes when sensors are deployed in the.

$$Cost_{s1} = \beta_E.F\left(d_{CA}^2\right) + \beta_D.\dot{F}(\lambda_A, \mu_A, K), \tag{24a}$$

$$Cost_{s2} = \beta_E.\left(F\left(d_{CB}^2\right) + F\left(d_{BA}^2\right)\right) + \beta_D.\left(\dot{F}(\lambda_A, \mu_A, K) + \dot{F}(\lambda_B, \mu_B, K)\right). \tag{24b}$$

The β_E is the weighted cost of energy compared to delay. When β_E gets larger, it has more influence on the total cost. β_E is assigned by the user (25). Total cost is calculated by subtracting directly from indirect routing cost function:

$$Cost = Cost_{s1} - Cost_{s2}. \tag{25}$$

3.2 Forwarding scheduling

FC needs the cluster's topology to run the forwarding scheduling method. The topology is given to FC in CC matrix, In this matrix the CC_{ij} is the communication cost between nodes i and j. The initial value of CC matrix elements is as:

$$CC : [\ ]_{N\times N} \rightarrow CC_{ij} = \begin{cases} F(d_{ij}^2), & i,j \quad \textit{has direct path} \\ \infty, & i,j \quad \textit{has indirect path} \end{cases} \tag{26}$$

where N is the number of cluster nodes, d_{ij} is distance between nodes i and j. At the beginning of the process, the elements of CC matrix should be initialized. When nodes i and j are not located in their sending range, then CC_{ij} is considered infinite (the value will be replaced by the other values according to the selected least-cost routes). Otherwise they are in the radio range of each other. The CC matrix is obtained from the network graph. The values of the CC matrix will be improved to the link cost. At the end of the algorithm, the CC matrix will have the best cost between each two sensor nodes.

The vector participation$_{N\times1}$ maintains a degree of each node participating in the routing process. It means how many routs a node i is in when data packets are sending to FC. Matrix Link$_{N\times N}$ have intermediate nodes located on the optimal path between every two nodes are placed in each other radio range. Matrix Links elements are a two-dimensional Vector$_{N\times2}$. If i and j are not in each other's radio range, Links$_{ij}$ is empty. The optimality of the paths is obtained from the cost function in section 3.1. The matrix CC Optimal value is obtained from Figure 3 pseudo code.

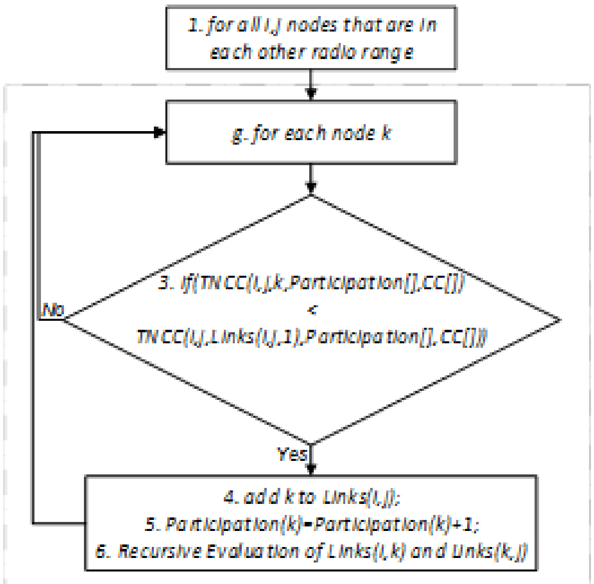

Figure 3: State diagram of determining matrix CC components.

If two nodes are in each other's radio range (each i and j), the algorithm searches the third node to send node's I data to node j, as the equation (20). Node k should be in the radio range of both nodes. TNCC function in Figure 3, compares the indirect $i-k-j$ and direct $i-j$ paths to find the best communication cost. If node k's communication cost is less than the current estimated cost, it will be selected as an intermediate node and is added to the Links$_{ij}$ vector, and the participation increases one. Then node k, as describes in the pseudo-code of Figure 3, links $i-k$ and $k-j$. After that, the algorithm is repeated. If the $i-k$ or K_J path can use another intermediate node, the node will be added to the array links.

If network conditions are modeled real, we can propose a good scheduling algorithm. Using the equation (16a), the energy consumption of each node is modeled. $EC_{N\times N\times h}$ matrix is considered to hold the node's energy consumption. EC matrix is created using the Path matrix 1's pseudo-code is used to specify the value of EC matrix elements.

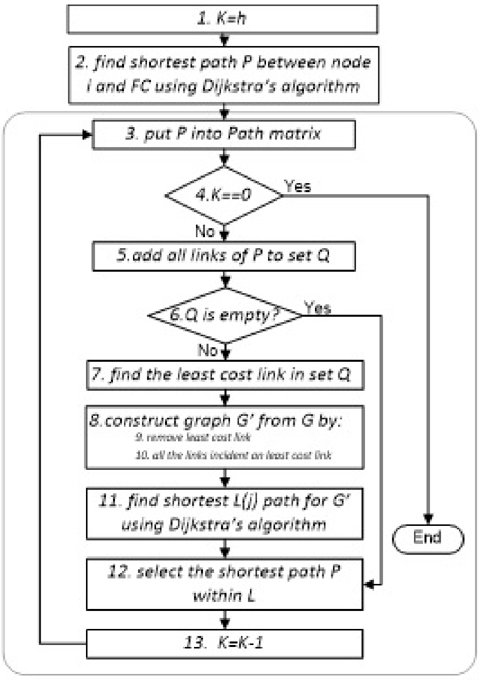

Figure 4: State diagram of finding h shortest path

Algorithm 1 The pseudo-code for EC matrix elements

for all $i \in 1,...,N$ **do**
  for all $j \in 1,...,N$ **do**
    for all $k \in 1,...,N$ **do**
      if $(Paths_{i,j,k+1<>0})$ **then**
        $EC_{i,j,k} = Econ(Paths_{i,j,k}, Paths_{i,j,k+1})$
      end if
    end for
  end for
end for

Econ(i,j) function determines the required energy for transmitting a data unit from node i to j based on their relative distance, using the equation (16b). $EC_{i,j,k}$ element defines the amount of energy which is consumed by node j, where the k^{th} optimal path between node i and FC is selected. As mentioned before, we propose a scheduling algorithm and selecting an optimal path between each node and FC. Therefore matrix $SP_{R\times N\times N}$ is defined. In this matrix, N, the number of nodes in a cluster, and R is the number of algorithm estimation rounds. Each element in $SP_{i,j,k}$ shows the number of data samples sent by node j in i^{th} round in k^{th} optimal path. In the estimation process, the algorithm runs in independent rounds. Each round time duration is predefined. The network lifetime will be $R \times T$ for each round time, T, FC collects data from nodes in each round, and estimates the parameter with desired precision. Equation (27) show the optimization problem proposed:

$$Min\ F=\gamma_1 N_1(\sum_{j=1}^{R}\sum_{i=1}^{N}\sum_{k=1}^{N} SP(j,i,m).EC(j,i,k))+\gamma_2 N_2(Var(.))-\gamma_3 N_3(R) \tag{27}$$

$$\gamma_1+\gamma_2+\gamma_3=1 \tag{27a}$$

$$S.T:\ \forall k\in N,\ \sum_{j=1}^{R}\sum_{i=1}^{N}\sum_{m=1}^{N}(SP(j,i,m).EC(j,i,m))+\sum_{j=1}^{R}(E_{ac}.AP(j,k))<E_{pri} \tag{27b}$$

$$S.T:\ \forall j\in R,\ AP(j,k)=\left\lceil \sum_{i=1}^{N}\sum_{m=1}^{N}(SP(j,i,m).Path(i,k,m)) \right\rceil \tag{27c}$$

$$S.T:\ \forall i\in R,\ \sum_{j=1}^{N}\sum_{k=1}^{N} SP(i,j,k)=\left\lceil \dot{\sigma}^2\left\{\frac{z_{\frac{\alpha}{2}}{}^2}{\eta^2}-\frac{1}{\tau^2}\right\}\right\rceil \tag{27d}$$

$$S.T:\ Var(.)=\sum_{q=1}^{N}\left\{\left(\sum_{j=1}^{R}\sum_{i=1}^{N}\sum_{m=1}^{N}(SP(j,i,m).EC(i,q,m))+\sum_{j=1}^{R}(E_{ac}.AP(j,q))\right)\right.$$
$$\left.-(\sum_{r=1}^{N}\sum_{j=1}^{R}\sum_{i=1}^{N}\sum_{m=1}^{N}(SP(j,i,m).EC(i,r,m))+\sum_{j=1}^{R}(E_{ac}.AP(j,r))/N)\right\}^2.$$

In the equation (27b) we see that nodes initial energy E_{pri} is bigger than node's energy consumption. A node consumes energy every time it is active and sends packet. So in a round if node is active it consumes a constant energy. If it is inactive, it consumes a negligible energy. The term $\sum_{j=1}^{R}\sum_{i=1}^{N}\sum_{m=1}^{N}(SP(j,i,m).EC(i,k,m))$ in the equation (27b), shows the energy consumption of a sending node. E_{ac} shows node's energy consumption in a round. Therefore term $\sum_{j=1}^{R}(E_{ac}.AP(j,k))$ shows total energy consumption of node during its lifetime. If a node does not send any data, it must be inactive. In the equation (27c) the relation between matrixes P, SP and path are shown.

The proposed scheduling and routing algorithm uses an NLP to find the best paths and node activity. The object and subjects of the proposed NLP are not very complicated. We test it using Matlab[1], and the answers and the response time is acceptable. The proposed NLP is acceptable to execute in a sensor node. Furthermore, the proposed scheduling and routing (including the proposed NLP) are performed once, so its complexity is negligible. Each CH solves the NLP once (the output will be the scheduling program and the routing paths), and informs the sensor member nodes, so the overhead is acceptable. So, our algorithm is efficient because the parameter distribution function is known. This function is compatible with estimating the parameter with desired precision. All parts of the proposed algorithm are fully adjusted to the application requirements.

[1] www.matlab.com

3.2.1 Matrix SP

In the equation (27), the best values of matrix SP elements are obtained. In this form, solving optimization problems is so hard, because the parameter R is on summation bound. We should replace parameter R with $\dot{R}$ in the equation (27) to solve the problem. The new objective function will be as:

$$Min\ F = \gamma_1 N_1 (\sum_{j=1}^{\dot{R}} \sum_{i=1}^{N} \sum_{k=1}^{N} SP(j,i,m).EC(j,i,k)) + \gamma_2 N_2 (Var(.)) - \gamma_3 N_3 (R). \tag{28}$$

This optimization problem has just one difference with the equation (27). The parameter R in the equation (27) is unknown, but in (28), $\dot{R}$ is known. In the following the proposed method, we defined parameter $\dot{R}$ and solved the equation (27). In line 1 of 2, the parameter $R(0 < R < RM)$ (here it is known as $\dot{R}$) is introduced.

Algorithm 2 Proposed optimal R algorithm

1: **for** $\dot{R} \in (0 - R_M)$ **do**
2:     $LB = 0, UB = R_M$
3: **end for**
4: **for all** $i \in 1,...,\log_2 R_M$ **do**
5:     $\dot{R} = (LB + UB) \div 2$
6:     solve the equation(22b) using $\dot{R}$
7: **end for**
8: **if** Equation(22b) is solved successfully **then**
9:     $LB = \dot{R}$
10: **else**
11:     $UB = \dot{R}$
12: **end if**
13: **return** $\dot{R}$

In Section 3.2.2, we find R_M value. Parameter R is the network lifetime. if $R = 0$, it means that the network cannot perform even one round. In line 4, the problem is solved successfully. There are two points here, 1) if $\dot{R} >$ optimal (R), then the optimization problem (Equation (27)) has no answer. This is because of the contradiction between the equations (27b) and (27d). If the optimization problem is solved for the current value of $\dot{R}$, we can accept the solution.

3.2.2 Calculating R_M

To calculate R_M , we have considered the over the optimal situation which, is not practical in reality. Therefore a network with the following conditions has been considered:

1. it consists of only one node (called selected node)
2. initial energy of selected node is E_T

3. the selected node consumes E_L energy unit due to send each message to FC. Considering that only one node exists in the network, the equation (27d) is rewritten (29). Parameters E_L and E_T are calculated using the equations (30) and (31), respectively.

$$S.T\colon \forall i{\in}R,\ \sum_{j=1}^{N}\sum_{k=1}^{N} SP(i,j,k){=}\left\lceil \sigma^2\left\{\frac{{z_{\frac{\alpha}{2}}}^2}{\eta^2}-\frac{1}{\tau^2}\right\}\right\rceil, \tag{29}$$

$$E_L{=}\ min(E_{i\in(1:N)}), \tag{30}$$

$$E_T{=}\ N{\times}E_{pri}. \tag{31}$$

In the definition of $\dot{\sigma}(\dot{\sigma}^2{=}max\left(\sigma_i^2\right) \mid i{\in}\{1,\ldots,\ N\})$, we have: $\sigma^2 < \dot{\sigma}^2$. In the equation (30), E_L is the minimum energy consumption of all nodes in a cluster. In the equation (31), parameter E_T is the initial energy of all the cluster nodes. With equations (29),(30) and(31), R_M is obtined as:

$$R_M{=}\left\lfloor E_L/\left(\left\lceil \sigma^2\left\{\frac{{z_{\frac{\alpha}{2}}}^2}{\eta^2}-\frac{1}{\tau^2}\right\}\right\rceil.E_T\right)\right\rfloor. \tag{32}$$

4 Performance Evaluation

We used MATLAB and OPNET software in investigating the performance of the proposed algorithm[1]. Calculations are implemented using MATLAB software, and network simulations are performed using OPNET software. Both software are based on C compiler, and we connected them. We call the MATLAB engine in OPNET environment. The main contribution of the proposed algorithm is to estimate the parameter based on user-defined precision. Also, joint routing and scheduling algorithms are provided to send the estimation output to FC. The proposed algorithm is fully adjusted to the proposed estimation model requirements. As it is presented in the rest of this section, by considering two heuristic algorithms, the proposed algorithm achieves its goals. We used heuristic algorithms because there is no previous work that performs the estimation and data gathering similar to our proposed algorithm (this model of evaluation is expected in this field [11, 5]).

We evaluate our algorithm efficiency with the following similar algorithms:

1. The proposed algorithm with single-hop routing, called $M1$.

2. The proposed algorithm with one route between each node and FC, called $M2$.

In this section, we will call our proposed algorithm, $M3$. Figure 5, shows the relation between precision parameters and data size.

[1] www.Opnet.com

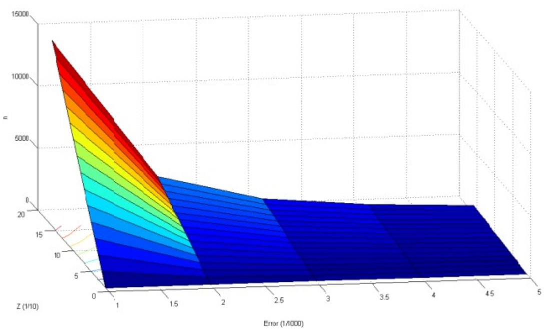

Figure 5: The relationship between data size and precision parameters.

The data size is calculated by the equation (14). The parameter $s\dot{\sigma}$, η, $z_{\frac{\alpha}{2}}$, and τ has an effect on data size. The values of $\dot{\sigma}$ and τ are determined based on network conditions. However, parameters $z_{\frac{\alpha}{2}}$ and η determine estimation precision, which is directly specified by the user. The X-axis in Figure 5, shows the acceptable error limitations. The scale of the x-axis is 1/1000. Y-axis shows the degree of certainty. Y-axis maps $z_{\frac{\alpha}{2}}$. The z-axis presents data size (n). It is evident in Figure 5 that, for more precision, smaller acceptable error limitations and a high sample size are required.

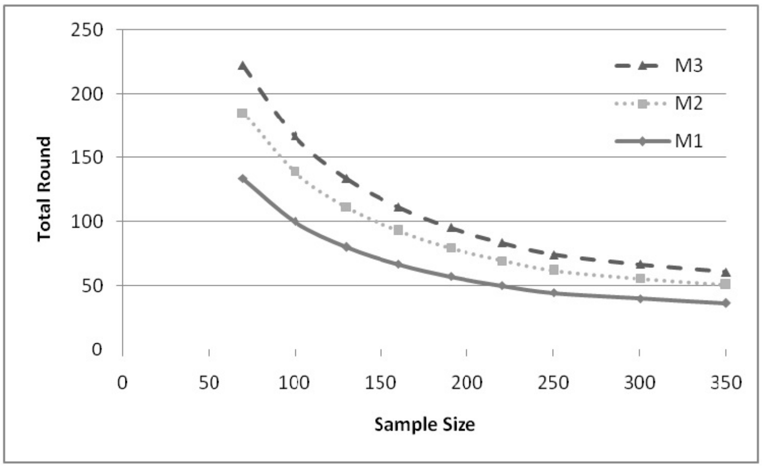

Figure 6: Total performed rounds versus different sample size

In Figure 6, we see the total performed rounds with different required sample sizes. In each round, if nodes don't have enough energy to run the estimation, they cannot finish that round. When network nodes send required data samples to FC, the estimation is complete. So, in Figure 6, it's clear that the lifetime of $M3$ is more than the other two algorithms because it can complete more rounds. The efficiency of $M3$ is more distinct in lower data size. The horizontal axis in Figure 6 shows the data size that is determined based on $\dot{\sigma}$, η, $z_{\frac{\alpha}{2}}$ and τ values.

Figure 7 shows the average packet transmission energy consumption for all algorithms. The X-axis is the number of network nodes, and the y-axis is the average energy consumption. Algorithms $M1$, $M2$, and $M3$ are compared with each other.

Since $M1$ uses direct routing, average energy remains constant. As shown in Figure 7, the average energy consumption of $M2$ is less than $M3$; this happens, because $M3$ has a fair and more efficient routing algorithm. As it is clear, if the network consumes energy at a fair, the result will be efficient. Therefore algorithms in which some nodes corporate in data sending more than other nodes. They lose energy faster. In some situations, $M3$ may use non-optimal routes to provide fairness in the node's energy consumption. However, $M2$ always selects the least cost route. Therefore, as is shown in Figure 8, the number of estimation rounds for $M3$ is more than two other algorithms.

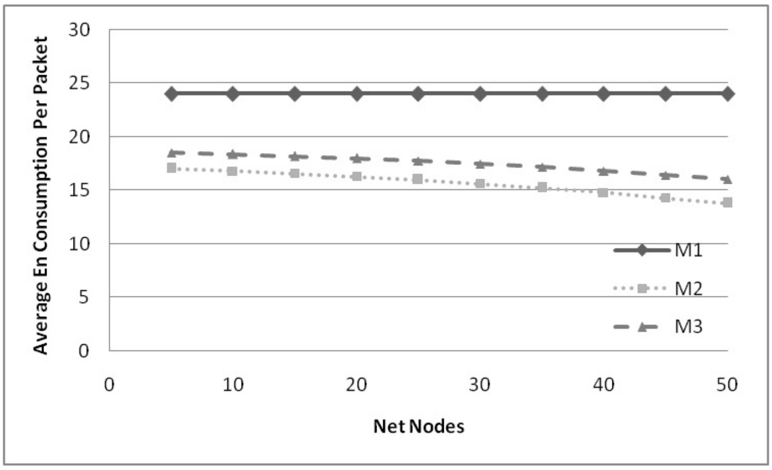

Figure 7: The average packet energy consumption versus the number of nodes.

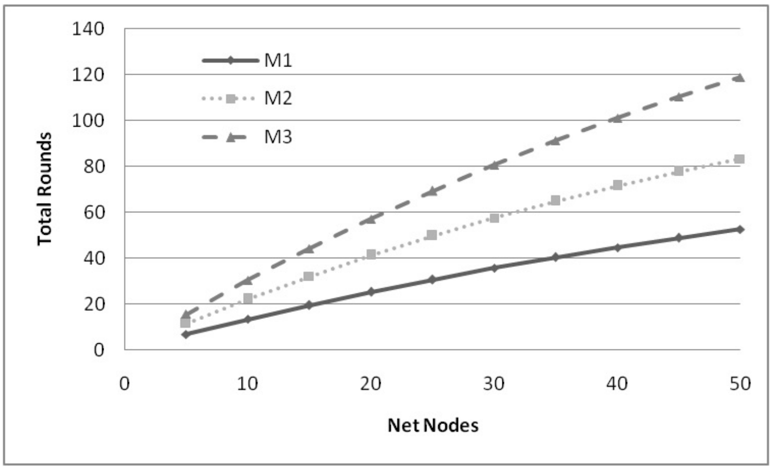

Figure 8: The number of performed estimation rounds versus the number of nodes.

As mentioned earlier, the main reason behind M3 efficiency in comparison with $M2$ is providing fairness. Like the $M2$ algorithm, $M3$ considers multiple least cost routes instead of one route. The Variance is obtained as follows:

$$Var = \sum_{i=1}^{N} (RE_i - AE)^2 \tag{33}$$

where RE_i is the remained energy of i^{th} node and, AE is all network node's average remained energy.

In Figure 9, energy variance of nodes for $M1$, $M2$ and, $M3$ algorithms are shown. Variance shows fairness for each algorithm. As can be seen in Figure 9, the $M3$ algorithm has a lower Variance. It is near zero. In this regard, $M1$ has the least efficiency because it uses direct forwarding.

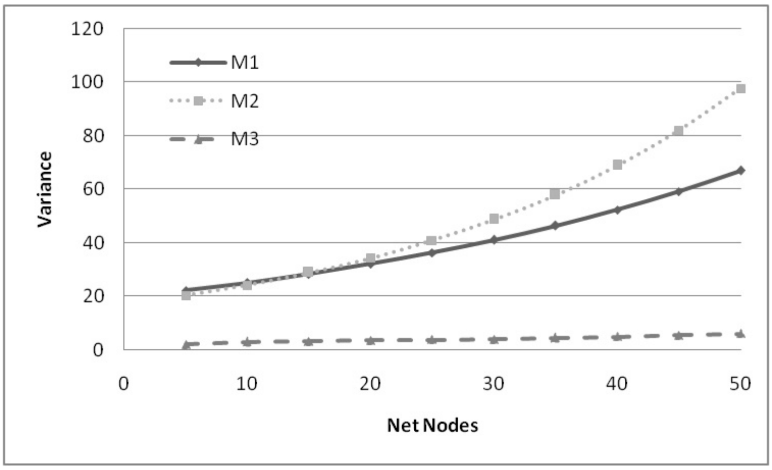

Figure 9: The variance of the remained network energy versus the number of nodes.

Figure 10 shows the packet's average end-to-end delay. Due to short distance limitations in sensor networks, we ignore propagation delay. Therefore, a delay only depends on the queuing delay. Note that delay is computed by the application layer.

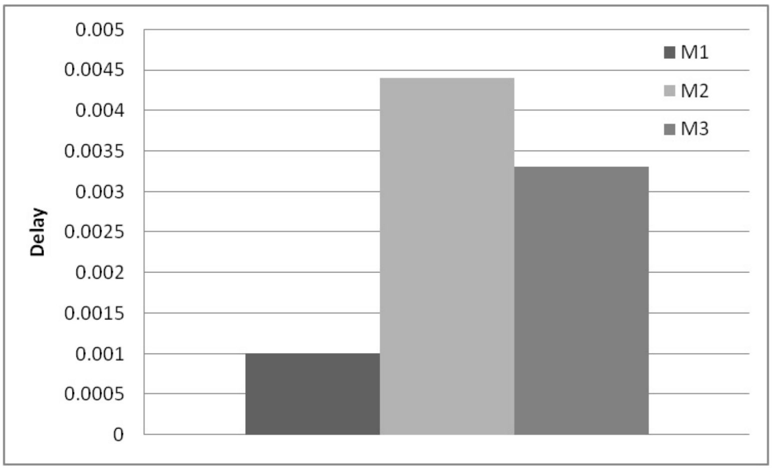

Figure 10: End-to-end delay in three algorithms.

Figure 10 shows that the M1 algorithm has a lower delay than the other algorithms, because in this algorithm, nodes send their data to the FC node directly. Also delay of the $M3$ algorithm is less than $M2$, because the number of intermediate nodes in this algorithm is lower than $M2$ because of energy consumption fairness.

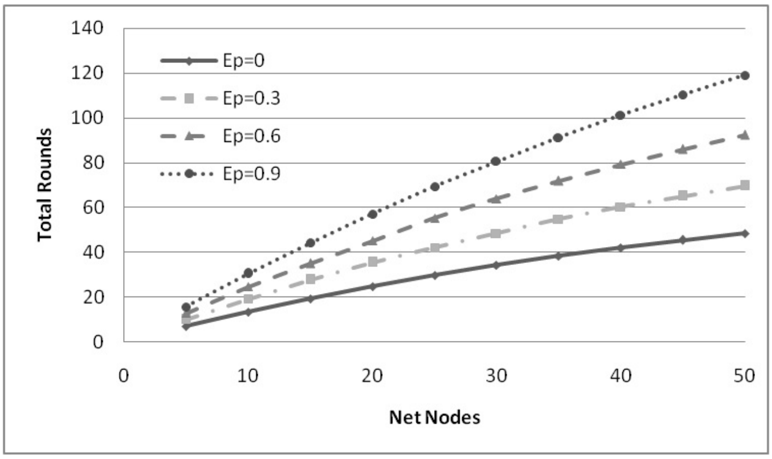

Figure 11: Total rounds versus the number of nodes.

Figure 11 shows the total round versus the number of nodes for $M3$. $E_p(\beta_E)$ parameter, as introduced in equation (14), determines energy priority. When E_p gets closer to 1, the $M3$ algorithm gets more energy-efficient, and the network lifetime increases.

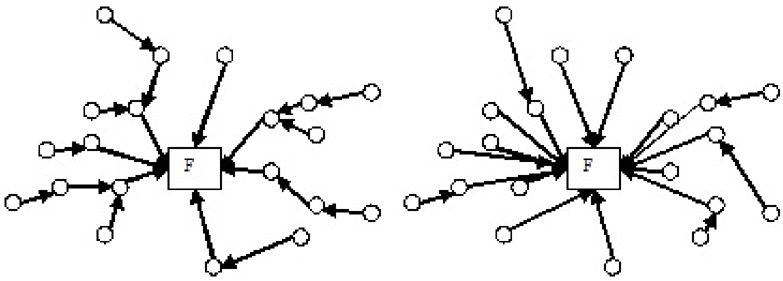

Figure 12: Intra-cluster routing tree, A-left) β_E= 1, B-right) β_E= 0.1

In Figure 12, intra cluster routing tree is presented to different values of $E_p(\beta_E)$. 14-A presents the routing tree when $\beta_E = 1$ and 14-B presents the routing tree when $\beta_E = 0.1$.

5 Conclusion

The proposed algorithm estimates the parameter with desired precision and increases network lifetime. This algorithm consists of a routing and scheduling algorithm to send data to FC and estimate a parameter in WSN. The network lifetime is calculated by the number of successful estimation rounds. The network will be non-functional if nodes don't have sufficient energy to provide enough samples for FC. To achieve the goals, the scheduler uses a nonlinear optimization problem (NPL). Based on the required accuracy of estimation, the proposed NLP not only determines the routes (multihop routing) but also determines the state of each sensor node (active or inactive). Regarding the environment characteristics, the parameter is considered a random variable, which is estimated using a Bayesian confidence interval based on the user's desired precision. The cost function of the multihop routing considers both delay and energy parameters. The proposed algorithm was compared with the other two algorithms that showed more efficiency. Without loss of generality, calculations are performed central, and links are considered loss-less. Finally, in future works, we will consider a fully distributed method to solve the optimization problem based on the link's error rate.

References

[1] Ayanoglu E. 1990. "On optimal quantization of noisy sources", IEEE Trans. Inf. Theory, 36(6), 1450–1452.

[2] Aysal T.C., Barner K.E. 2008. "Constrained decentralized estimation over noisy channels for sensor networks", IEEE Transaction on Signal Processing, 56(4), 1398-1410.

[3] Akyildiz I.F., Su W., Sankarsubramaniam Y., Cayirci E. 2002. "Wireless sensor networks: A survey", Computer Network, 38, 393–422.

[4] Akyildiz I.F., Su W., Sankarasubramaniam W., Cayirci E. 2002. "A survey on sensor networks.", IEEE Communication Magazine, 102–114.

[5] Bertrand A. Moonen M. 2011. "Distributed adaptive estimation of node-specific signals in wireless sensor networks with a tree topology", IEEE Transaction on Signal Processing, 59(5).

[6] Cui S., Goldsmith A., Bahai A. 2004. "Joint modulation and multiple access optimizations under energy constraints", Proc. IEEE Global Telecomm. Conf., Dallas, Texas, 151–155.

[7] Chair Z., Varshney P.K. 1998. "Distributed Bayesian hypothesis testing with distributed data fusion", IEEE Trans. Syst., Man, Cybern., 18, 695–699.

[8] Chen H. 2010. "Performance-energy tradeoffs for decentralized estimation in a multihop sensor network", IEEE Sensors Journal, 10(8).

[9] Cheng C.T., Tse C.K., Lao F.C.M. 2010. "An energy-aware scheduling scheme for wireless sensor networks", IEEE Transactions on Vehicular Technology, 59(7).

[10] Cohen R., Kapchits B. 2009. "An optimal wake-up scheduling algorithm for minimizing energy consumption while limiting maximum delay in a mesh sensor network", IEEE/ACM Transactions on Networking, 17(2).

[11] Chaudhary M.H. Vandendorpe L. 2012. "Power constrained linear estimation in wireless sensor networks with correlated data and digital modulation", IEEE Transaction on Signal Processing, 60(2).

[12] Dardari D., Conti A., Burrati C., Verdone R. 2007. "Mathematical evaluation of environmental monitoring estimation error through energy-efficient wireless sensor networks", IEEE Transaction on Mobile Computing, 6(7).

[13] Feng W. 2020. "Joint energy-saving scheduling and secure routing for critical event reporting in wireless sensor networks", IEEE Access, 8.

[14] Fitzgerald E. 2018. "Energy-optimal data aggregation and dissemination for the internet of things", IEEE Internet of Things Journal.

[15] Gross D., Shortle J.F., Thompson J.M., Harris C.M. 2008. "Fundamentals of queuing Theory", Fourth Edition, John Wiley and Sons.

[16] Incel Z.D., Ghosh A., Krishnamachari B., Chintalapudi K. 2012. "Fast data collection in tree-based wireless sensor networks", IEEE Transactions on Mobile Computing, 11(1).

[17] Karkvandi H. R., Pecht E., Yadid-Pecht O. 2011. "Effective lifetime-aware routing in wireless sensor networks", IEEE Sensors Journal, 11(12).

[18] Lam A., Reibman, A. 1993. "Quantizer design for decentralized systems with communication constraints", IEEE Trans. Commun., 41, 1602–1605.

[19] Luo Z.Q. 2005. "Universal decentralized estimation in a bandwidth constrained sensor network", IEEE Transaction on Information Theory, 51(6).

[20] LI J., AlRegib G. 2007. "Rate-constrained distributed estimation in wireless sensor networks", IEEE Transaction on Signal Processing, 55(5).

[21] LI J., AlRegib G. 2009. "Network lifetime maximization for estimation in multihop wireless sensor networks", IEEE Transactions on Signal Processing, 57(7).

[22] Liu F., Tsui C-Y, Zhang Y.J. 2010. "Joint routing and sleep scheduling for lifetime maximization of wireless sensor networks", IEEE Transactions on Wireless Communications, 9(7).

[23] Lehman E.L., Casella G. 1983. "Theory of point estimation", 2nd Edition, Springer.

[24] Lehman E.L. 1997. "Testing statistical hypothesis", 2nd Edition, Springer.

[25] Noori M. Ardakani M. 2011. "Energy efficient of universal decentralized estimation in random sensor networks", IEEE Transactions on Wireless Communication, 10(12).

[26] Ribeiro A., Giannakis G.B. 2006. "Bandwidth-constrained distributed estimation for wireless sensor networks-part II: unknown probability density function", IEEE Transaction on Signal Processing, 54(7).

[27] Ryu J., Lee C.G., Kwon T.T., Han J. 2009. "Combined scheduling and routing for deterministic guarantee of end-to-end deadlines in cell structured sensor networks", IEEE Sensors Journal, 9(10).

[28] Sichitiu M.L. 2004. "Cross-layer scheduling for power efficiency in wireless sensor networks", Proceedings of INFOCOM 2004, Hong Kong, 2, 266-276.

[29] Shi L. 2010. "Resource optimization in a wireless sensor network with guaranteed estimator performance", IET Control Theory and Applications, 4 (5).

[30] Tang S. Li W. 2006. "QoS supporting and optimal energy allocation for a cluster based wireless sensor network", Computer Communications.

[31] Willsky A.S., Bello M., Castanon D.A., Levy B.C., Verghese G. 1982. "Combining and updating of local estimates and regional maps along with sets of one-dimensional tracks", IEEE Trans. Autom. Control, AC-27, 799–813.

[32] Wu Y., Li X-Y, Liu Y., Lou W. 2010. "Energy-efficient wake-up scheduling for data collection and aggregation", IEEE Transactions on Parallel and Distributed Systems, 21(2).

[33] Wang H., Yang Y., Ma M., He J., Wang X. 2008. "Network lifetime maximization with cross-layer design in wireless sensor networks", IEEE Transactions Wireless Commun., 7(10), 3759-3768, Oct.

[34] Xiao J.J., Ribeiro A., Luo Z.Q., Giannakis G.B. 2006. "Distributed compression-estimation using wireless sensor networks", IEEE Signal Processing Magazine, 23(4), 27–41.

[35] Xiao J.J., Cui S., Luo Z.Q., Goldsmith A.J. 2006. "Power scheduling of universal decentralized estimation in sensor networks", IEEE Transaction on Signal Processing, 54(2).