

# Efficient Minimum Cost Reactive Monitoring of Gaussian Random Field in Wireless Sensor Networks

A. Mohtasham<sup>\*†</sup>, Ahmad Khonsari<sup>\*†</sup>, and Mohammad S. Talebi<sup>†</sup>

<sup>\*</sup> *ECE Department, University of Tehran, Tehran, Iran*

<sup>†</sup> *School of Computer Science, IPM, Tehran, Iran*

*Emails: {a.mohtasham, ak, mstalebi}@ipm.ir*

## Abstract

*Monitoring is one of key applications in wireless sensor networks. In this paper, we consider Reactive Monitoring in sensor networks, which takes the advantages of very low communication overhead. In this respect, our primary concern is devising the reactive monitoring mechanism with minimal communication cost for a sensor network scattered in a Gaussian field. We cast the threshold assignment for reactive monitoring as an optimization problem based on the statistics of each sensor measurement. Then, we propose a distributed algorithm to set local thresholds on each sensor node based on Particle Swarm Optimization (PSO) technique. Through simulation, we illustrate that the proposed threshold assignment algorithm can significantly reduce the communication overhead of the monitoring mechanism in sensor networks.*

## 1. Introduction

Emergence of microsensors based on MEMS technology has made it possible to deploy a large networked system comprising of battery-operated sensor nodes. Amongst the wide variety of applications, monitoring is a subject of primary concern in current and emerging applications of wireless sensor networks. The significance of monitoring in wireless sensor network is increasing not only for accounting and management, but also for revealing anomalies and malicious attacks.

One can distinguish between two fundamental classes of monitoring mechanisms. In the *Statistical Monitoring* mechanism, each agent is required to send the raw measured data to the central node (base station)

and then the central node will be responsible for monitoring the current state of the network. On the other hand, in *Reactive Monitoring* mechanism nodes are required to send their measurements when central node sends appropriate query to them. This mechanism is essentially relying on local filters at monitoring sites that act as local constraint monitors and each filter in case of constraint violation (simply in the form of crossing a locally predefined value), sends an update message to the central node. Central node, upon receipt of an update message, queries all nodes to send their data to monitor the entire data set.

Reactive monitoring is essentially designed to alleviate the communication burden of monitoring tasks in distributed and resource constrained scenarios, and therefore is very appropriate for battery-operated wireless sensor networks [1]. In order to prolong the lifetime of the sensor network, monitoring mechanism must incur very low overhead, i.e. the communication cost of submitting measured data to the central node. Thus, our primary concern in adopting such a monitoring paradigm is adapting the monitoring scenario to the environmental conditions of the sensor networks.

The major contribution of this work is to exploit the reactive monitoring mechanism for usage in wireless sensor networks. Our primary concern is to reduce the communication burden emanated from transmitting monitoring queries from sensor nodes. The pure reactive monitoring mechanisms introduced so far, are relying on a general architecture and have not exploited the spatial distribution of sensor nodes over the target environment. We concentrate on a wide class of monitoring applications in which the essential criterion of the monitoring is to check whether an aggregate of the form  $\sum_i a_i x_i$  has exceeded a global parameter of the network, say a simple threshold  $T$ . We aim at determining the local thresholds based on

---

*A. Mohtasham is a MSc. student studying Computer Engineering at ECE Department, University of Tehran.*

the statistics of local events. Particularly, we assume that the underlying phenomenon to be monitored is a Gaussian stochastic process spatially distributed over the network. Such an environment is usually referred to as *Gaussian random field*. There exist a plethora of physical events that are spatially distributed and obey Gaussian distribution.

Towards this end, we formulate the threshold assignment problem to minimize the communication overhead of the network. This allows us to design a distributed threshold assignment algorithm which determines the optimal thresholds based on the statistics of each sensor.

The remainder of this paper is organized as follows. Section 2 reviews related work. Section 3 defines the network model and states some preliminaries regarding the reactive monitoring mechanism. Section 4 formulates the optimization problem. Section 5 provides the optimal solution to the threshold assignment problem and presents a distributed threshold assignment algorithm. Section 6 validates experimental evaluation of the proposed algorithm. Finally, Section 7 concludes the study.

## 2. Related Work

In recent years, continuous query processing for monitoring distributed data streams has received much research attention. Monitoring of aggregate of data in networks initially mentioned by the pioneering work of Dilman *et al.* [1]. They introduced installing local mathematical constraints at remote monitoring sites and present a simple approach for the efficient reactive monitoring.

Keralpera *et al.* [2] extended the work of [1] and presented several algorithms for static and adaptive threshold setting for monitoring thresholded counts queries and analyzed the communication complexity of each algorithm. [3] investigates a hybrid push-pull approach for monitoring global system parameters in IP networks. Authors present algorithms for selecting which elements “to push” and which “to pull” when network elements events are independent or dependent. Recent work of Kashyap *et al.* [4] considers the problem of non-zero slack threshold assignment which adaptively dedicates fraction of total threshold to monitoring node to absorb small local threshold crossing that eliminate the need for global system

polling. All the above works assume centralized threshold computation and assignment but in this paper we propose a distributed algorithm for optimal threshold setting.

In our previous studies, we addressed the problem of distributed threshold selection for reactive monitoring in wireless sensor network scenarios [5]-[7]. This work differs from those in that here we assume that the underlying monitoring event follows Gaussian distribution, thereby making the area a Gaussian random field. Since Gaussian distribution is somewhat disobedient to be addressed in dual-based solutions for this class of problems ([5]-[7]), this paper takes a quite different approach and solves the problem using Particle Swarm Optimization (PSO) technique [10].

## 3. Network Model and Monitoring Scenario

### 3.1. Network Model

We consider a sensor network comprising of  $n$  nodes scattered over an area. We assume that each sensor node  $i$  is in charge of monitoring a physical phenomenon within its coverage region, and whose measured value is denoted by  $x_i$ . We assume that there is a base station being responsible for monitoring the network. The goal of the network is to monitor when the aggregate value of all the measurements of the form  $\sum_{i=1}^n a_i x_i$  crosses a predetermined threshold  $T$ , where  $a_i, i = 1, \dots, n$  are predefined constants.

### 3.2. The Monitoring Algorithm

In this paper we focus on reactive monitoring schemes, in which for each monitoring agent  $i$  a local threshold  $t_i$  is determined. Here, we exploit the Simple-Value (*SV*) Algorithm introduced in [1]. The *SV* Algorithm consists of two parts:  $SV_N$  for monitoring nodes and  $SV_C$  for central base station. Both of these parts are described in [1] and are listed in the sequel as *SV* Algorithm.

## 4. Problem Formulation

One of the key features for a reactive monitoring algorithm in a resource-constrained sensor network is to incur low monitoring overhead. Since power is a scarce resource and the primal reason for energy

drainage is communication, our main goal is to minimize the probability of global polling. In this respect, it is shown in [8] that the probability of preserving all local constraints should be maximized. Formally, it is equivalent to saying that  $\Pr(x_1 \leq t_1, \dots, x_n \leq t_n)$  should be maximized. In sensor networks, monitoring sites are usually located far enough from each other so that correlation between nearby sites can be ignored, and thereby sensor measurements can be deemed independent. Therefore, we conclude that

$$\Pr(x_1 \leq t_1, \dots, x_n \leq t_n) = F(\mathbf{t}) = \prod_{i=1}^n F_i(t_i) \quad (1)$$

where  $F$  is the joint Gaussian CDF (Cumulative Frequency Distribution) of all sensors,  $F_i$  is the Gaussian CDF of sensor measurement  $i$ , and  $\mathbf{t} = (t_i, i = 1..n)$  is the vector representation of local thresholds. Moreover, the last equation is obtained using the fact that all sensor measurements are assumed to be independent. To find the minimal communication cost threshold assignment,  $\prod_{i=1}^n F_i(t_i)$  should be maximized such that [8]:

$$\max_{\mathbf{t}} \prod_{i=1}^n F_i(t_i) \quad \text{s.t.} \quad \sum_{i=1}^n a_i t_i \leq T \quad (2)$$

---

#### SV. Simple-Value Algorithm

---

$SV_N$  :

At each monitoring node  $i$ :

Initialize threshold  $t_i$ .

if  $x_i > t_i$

send an update to the base station.

$SV_C$  :

Initialize  $f = 0$

while (TRUE)

if an update received

$f \leftarrow \text{POLL ALL}(x_i)$

$f > T$

report ALARM

---

#### SV. Simple-Value Algorithm

For the Gaussian sensor measurements, CDF is defined by

$$F_i(t_i) = \int_{-\infty}^{t_i} \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{-\frac{(z-\mu_i)^2}{2\sigma_i^2}} dz \quad (3)$$

For the sake of convenience in our analysis, we use the following approximation for Gaussian CDF for

( $t_i \geq \mu_i$ ) [9]:

$$F_i(t_i) \approx 1 - \frac{1}{\sqrt{2\pi}} e^{-\frac{(t_i-\mu_i)^2}{2\sigma_i^2}} \sum_{j=1}^5 b_j y_i^j \quad (4)$$

where  $(b_1, \dots, b_5) = (0.32, -0.36, 1.78, -1.82, 1.33)$  and  $y_i = \frac{1}{1+0.23t_i}$ . Therefore, we approximate the problem (2) by rewriting it as

$$\max_{\mathbf{t}} \prod_{i=1}^n \left( 1 - \frac{1}{\sqrt{2\pi}} e^{-\frac{(t_i-\mu_i)^2}{2\sigma_i^2}} \sum_{j=1}^5 b_j y_i^j \right) \quad (5)$$

subject to:

$$\sum_{i=1}^n a_i t_i \leq T \quad (6)$$

In the sequel, we conduct the relevant analysis for solving problem (5)-(6).

## 5. Optimal Threshold

In this section, we solve optimization problem (5)-(6). Problem (5)-(6) is constrained and cannot be solved directly in distributed systems such as sensor networks. On the other hand, obtaining the closed form expression for optimal threshold is cumbersome, if not impossible. To solve problem (5)-(6), we use Particle Swarm Optimization (PSO) technique [10]. PSO technique cannot be directly applied to constrained optimization problems, thus we first relax the problem by defining exterior penalty function [12] as

$$\phi(\mathbf{t}, r_k) = \prod_{i=1}^n \left( 1 - \frac{1}{\sqrt{2\pi}} e^{-\frac{(t_i-\mu_i)^2}{2\sigma_i^2}} \sum_{j=1}^5 b_j y_i^j \right) - r_k \left( \max[0, \sum_i t_i - T] \right)^2 \quad (7)$$

where  $r_k$  is a positive penalty parameter. The exterior penalty function algorithm iteratively finds the optimal solution to (5)-(6) through the following steps.

- *Step 1:* Set  $k = 1$  and choose an appropriate value for  $r_k$  and start from any initial solution  $\mathbf{t}_k$
- *Step 2:* Find  $\mathbf{t}_k^*$  that maximizes (7).
- *Step 3:* If  $\mathbf{t}_k^*$  is feasible, i.e. it satisfies (6), then the algorithm terminates and  $\mathbf{t}_k^*$  is the optimal solution to (5)-(6). Otherwise go the next step.
- *Step 4:* Choose the next value of penalty parameters  $\frac{r_{k+1}}{r_k} = c$  where  $c > 1$  is a constant. Set  $\mathbf{t}_{k+1} = \mathbf{t}_k^*$  and go to Step 2.

From Step 2 in algorithm above, it's apparent that finding the optimal  $\mathbf{t}_k^*$  is required for each penalty parameter  $r_k$ . To do so, we use the Particle Swarm Optimization technique. More details of this technique can be found in [11].

We assume that swarm size, i.e. number of particles in the swarm is  $M$ . PSO technique is governed by the two following equations

$$\mathbf{v}_m(t+1) = w\mathbf{v}_m(t) + c_1\text{rand}()(\mathbf{y}_m(t) - \mathbf{x}_m(t)) + c_2\text{rand}()(\hat{\mathbf{y}}_m(t) - \mathbf{x}_m(t)) \quad (8)$$

$$\mathbf{x}_m(t+1) = \mathbf{x}_m(t) + \mathbf{v}_m(t+1) \quad (9)$$

where  $\mathbf{v}_m = (v_{mj}, j = 1, \dots, n)$  is the velocity vector for particle  $m$  and  $v_{mj}$  is the velocity of particle  $m$  in dimension  $j$  and  $\mathbf{x}_m = (x_{mj}, j = 1, \dots, n)$  is the position vector of particle  $m$  and  $x_{mj}$  is the position of particle  $m$  at dimension  $j$ .  $\mathbf{y}_m = (y_{mj}, j = 1, \dots, n)$  and  $\hat{\mathbf{y}}_m = (\hat{y}_{mj}, j = 1, \dots, n)$  are the best position vectors found by particle  $m$  and by the entire swarm, respectively.  $\text{rand}()$  returns a random number uniformly distributed in  $[0, 1]$  interval and  $w$  is the inertia weight.

We now devise a threshold assignment algorithm based on the iterative solution obtained above. First, we assume that monitoring sensors are arranged in a ring topology so that each node can update information with its two neighboring nodes. Prior to running the algorithm, all nodes are required to send their parameters, i.e.  $\mu_i$  and  $\sigma_i$ , using flooding-like algorithms or gossiping, to all other nodes.

Although only as few as 10 nodes (particles) will suffice for a swarm, for the sake of simplicity, all nodes contribute to one swarm instead of multiple swarms. Therefore, we can use node index  $i$  and particle index  $m$ , interchangeably. Based on the steps 1-4 described in this section, each node iteratively calculates its optimal threshold. Each node  $i$  receives  $\hat{y}_i$  from the network and updates  $v_i$  using (8) and consequently  $x_i$  using (9). The communication overhead of the algorithm only stems from exchanging information between two nearby nodes.

## 6. Experimental Evaluation

We have conducted simulation experiments to evaluate the performance of the proposed algorithm using MATLAB and OMNeT++. In our simulation scenario, we consider a sensor network consisting of 100 sensor

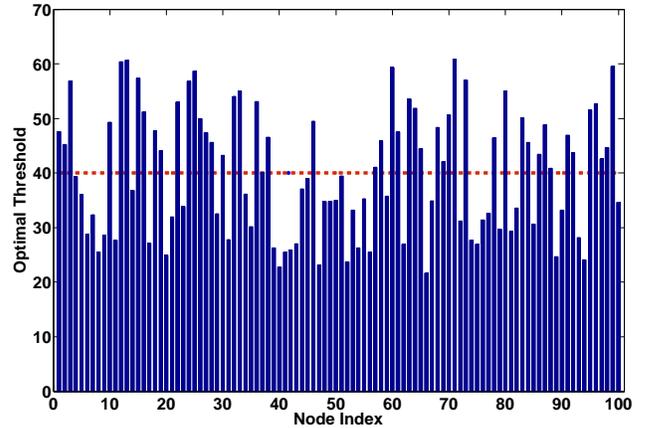


Figure 1. Optimal Threshold for Sensor Nodes (Bar Plot) and Uniform Threshold (Dashed Line)

nodes. We assume that each node  $i$  takes measurements of a normally distributed physical phenomenon with parameters  $(\mu_i, \sigma_i^2)$ . The corresponding coefficient  $a_i$  for such an illustrative experiment is set to 1.

For the first experiment  $\mu_i$  is drawn uniformly from  $[10, 40]$  and  $\sigma_i^2 = \mu_i$ . Total threshold  $T$  is set to 4000. In our simulation experiments, we have run the algorithm using the swarm size of  $M = 24$ . The inertia weight  $w$  is set to 0.4.  $c_1$  and  $c_2$  are both set to 2 and  $r_1$  is chosen to be 1. Since only one constraint exists in the optimization problem (5), once the algorithm runs through steps 1-4, obtained solution satisfies the constraint and thereby the algorithm would terminate.

In order to have a better understanding of how optimal thresholds are scattered in a heterogeneous fashion over the network, Fig. 1 depicts the bar plot of optimally-assigned thresholds for all nodes. As Fig. 1 implies, optimal thresholds for the majority of nodes are quite different from the uniform thresholding schemes, which is shown in the dashed line.

In order to have insight about the convergence behavior of the algorithm, evolution of the objective of problem (5)-(6) for all nodes toward the optimal value is depicted in Fig. 2. In this figure, evolution of  $\prod_i F_i(t_i)$  function for some sensor nodes is depicted, however, the difference is intangible. This figure divulges that the convergence of the algorithm is relatively fast and after a few hundreds of iterations, each node obtains its optimal threshold.

To show the efficiency of the algorithm, the overhead due to messages for both optimal and uniform thresh-

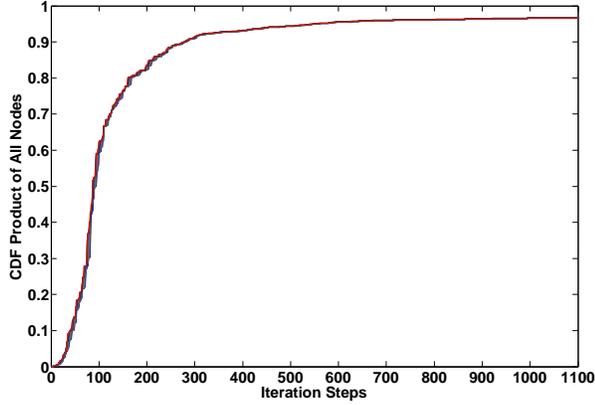


Figure 2. Evolution of CDF Product of All Sensor Nodes ( $\prod_{i=1}^n F_i(t_i)$ )

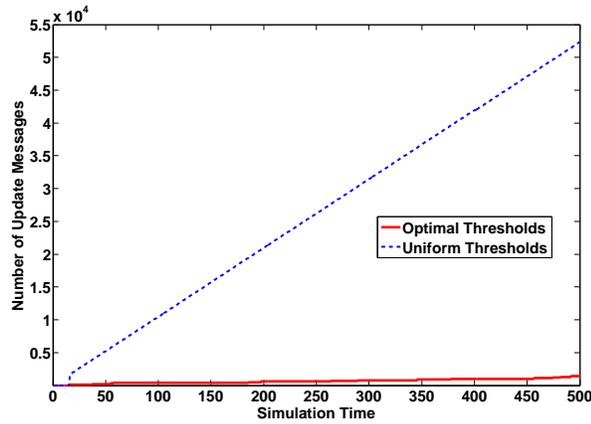


Figure 3. Communication Cost of The Monitoring Mechanism With Optimal and Uniform Thresholds

olds are compared in Fig. 3. This figure divulges that the number of messages is dramatically alleviated by a factor of 20-40. Such a tangible reduction in overhead proves quite promising especially when there are a lot of monitoring sensors in the network.

## 7. Conclusion

In this paper, we addressed the problem of threshold selection for reactive aggregate threshold monitoring in WSNs scattered in Gaussian random fields. We formulated designing the minimal-cost network monitoring scenario by casting the threshold assignment as an optimization problem. We have exploited PSO technique to solve the problem which led to a distributed solution.

The results extracted from the experimental evaluations was promising and verified that the communication cost of the algorithm is dramatically alleviated compared to the naïve threshold assignments.

## References

- [1] M. Dilman and D. Raz, "Efficient reactive monitoring," *In Proc. of INFOCOM'01*, 2001.
- [2] R. Keralapura, G. Cormode, and J. Ramamirtham, "Communication-efficient distributed monitoring of thresholded counts," *In Proc. of SIGMOD'06*, 2006.
- [3] A. Bulut, N. Koudas, A. Meka, A. K. Singh, and D. Srivastava, "Optimization Techniques for Reactive Network Monitoring," *IEEE Transactions on Knowledge and Data Engineering*, To appear.
- [4] S. Kashyap, J. Ramamirtham, R. Rastogi, and P. Shukla, "Efficient Constraint Monitoring Using Adaptive Thresholds," *In Proc. of ICDE'06*, 2006.
- [5] A. Abbasi, A. Khonsari, and M. S. Talebi, "Flooding-Assisted Threshold Assignment for Aggregate Monitoring in Sensor Networks," *In Proc. of ICDCN'09*, 2009.
- [6] M. S. Talebi, A. Khonsari, and A. Abbasi, "Distributed Threshold Selection for Aggregate Threshold Monitoring in Sensor Networks," *In Proc. of CCNC'09*, 2009.
- [7] M. S. Talebi, A. Khonsari, and R. Jabarvand, "Cost-Aware Reactive Monitoring in Resource-Constrained Wireless Sensor Networks," *In Proc. of WCNC'09*, 2009.
- [8] S. Agrawal, S. Deb, K. V. M. Naidu, and R. Rastogi, "Efficient Detection of Distributed Constraint Violations," *In ICDE'07*, 2007.
- [9] S. L. Miller and D. G. Childers, *Probability and random processes: with applications to signal processing and communications*, Academic Press, 2004.
- [10] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proc. IEEE Conf. Neural Networks IV*, Piscataway, NJ, 1995.
- [11] J. Robinson and Y. R. Samii, "Particle swarm optimization in electromagnetism," *IEEE Trans. Antennas Propagat.*, vol. 52, no. 2, pp. 397407, Feb. 2004.
- [12] S. S. Rao, *Optimization: Theory and Applications*. New Delhi, India: Wiley Eastern Limited, 1995.