

## Research Article

# Bio-Inspired Distributed Sensing Using a Self-Organizing Sensor Network

Indu Sreedevi,<sup>1</sup> Shubham Mankhand,<sup>2</sup> Santanu Chaudhury,<sup>2</sup> and Asok Bhattacharyya<sup>1</sup>

<sup>1</sup> *Electronics and Communication Engineering Department, Delhi College of Engineering, India*

<sup>2</sup> *Electrical Engineering Department, IIT Delhi, India*

Correspondence should be addressed to Indu Sreedevi; [s.indu@rediffmail.com](mailto:s.indu@rediffmail.com)

Received 22 August 2012; Revised 22 March 2013; Accepted 14 April 2013

Academic Editor: Piero Castoldi

Copyright © 2013 Indu Sreedevi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Nature offers several examples of self-organizing systems that automatically adjust to changing conditions without adversely affecting the system goals. We propose a self-organizing sensor network that is inspired from real-life systems for sampling a region in an energy-efficient manner. Mobile nodes in our network execute certain rules by processing local information. These rules enable the nodes to divide the sampling task in a manner such that the nodes self-organize themselves to reduce the total power consumed and improve the accuracy with which the phenomena are sampled. The digital hormone-based model that encapsulates these rules, provides a theoretical framework for examining this class of systems. This model has been simulated and implemented on cricket notes. Our results indicate that the model is more effective than a conventional model with a fixed rate sampling.

## 1. Introduction

Self-organizing mobile sensor network is built with sensor nodes that may spontaneously create an ad hoc network, assemble the network themselves, dynamically adapt to degradation or device failure, manage movement of sensor nodes, and react to changes in task and network requirement. Such networks can be deployed for environmental monitoring, surveillance application, disaster management, and so forth.

Sensor networks are event-based systems that rely on the collective effort of densely deployed sensor nodes which continuously observe physical phenomenon. The main objective of a sensor network is to reliably detect/estimate event features from the collective information provided by sensor nodes [1]. Therefore, the energy and hence processing constraints of small wireless sensor nodes overcome by this collective sensing notion which is realized via their networked deployment. While the collaborative nature of sensor nodes brings significant advantages over traditional sensing including greater accuracy, larger coverage area, and extraction of localized features, the spatiotemporal correlation among the sensor observations is another significant and unique

characteristic of the sensor networks which can be exploited to drastically enhance the overall network performance.

In addition to the collaborative nature of sensor networks, the existence of spatial and temporal correlations [2] bring significant potential advantages for the development of efficient communication protocols [3] well suited for the sensor networks paradigm. For example, intuitively, due to the spatial correlation, data from spatially separated sensors is more useful to the base station than highly correlated data from nodes in proximity. Therefore, it may not be necessary for every sensor node to transmit its data to the base station; instead, a smaller number of sensor measurements might be adequate to report the event features within a certain error level. Similarly, for a certain event tracking application, the measurement reporting frequency, at which the sensor nodes transmit their observations, can be adjusted such that temporally correlated signal is captured at the fusion center within a certain distortion level and with a minimum energy expenditure.

In this paper, we propose a scheme for building an autonomous system, for spatio-temporal reconstruction of environmental phenomena using mobile nodes. A key feature of our approach will be the focus on energy efficiency,

by employing adaptive sampling in both space and time. In the process, we optimize the number of message transmissions needed for collaboration among nodes, since the transmission contributes significantly to the total power consumption. The most important point, however, is to make the system self-organizing and is achieved using simple, local rules. The main advantage of such systems is that it reduces the dependence of critical nodes for their proper functioning. For example, in case of multisensor detection, if a fusion center fails, most likely a new one will be elected, but it will waste resources in doing nonproductive work. Also, readings need to be collated at fusion centers, before a result is produced. Clearly, this produces accurate results, but again, at increased resource/energy costs. We envisage that using a self-organizing system shall maintain the balance between fidelity of results and energy consumed, while retaining scalability and fault tolerance.

The remaining part of the paper is organized as follows. In Section 2, some previous works in the field of self-organizing sensor network are discussed. In Section 3, a detailed discussion of the problem is presented. In Section 4, the proposed EDHM is described in detail. In Section 5, the EDHM-based spatio-temporal signal reconstruction is presented. In Section 6, results of both simulation and hardware implementation are discussed. Finally, Section 7 concludes the paper.

## 2. Related Work

A novel computational model for self-organization was first proposed by Turing as Turing's reaction-diffusion model [4]. This model was mathematically supported by [5] and experimentally by [6]. Later, cellular automata-(CA-) based model for self-organization was developed by Gutowitz [7]. The CA-based model was further modified by Leach and Taffoli for incorporating nonlinearity.

In fact, the DHM [8] for self-organization has been proposed as a potential mechanism for development and differentiation in artificial life systems [9]. Swarm robotics [10] is a very active research area and has proposed many approaches. References [11, 12] showed that a set of autonomous agents can use pheromone to form interesting and complex global behaviors and exhibit swarming behaviors [13].

The concept of biological hormones proposed by [14] was one of the earliest attempt to build systems that are robust, flexible, and have the capability to self-organize. The ADS (autonomous decentralized systems) technology has been applied to industrial problems [15] and has the properties of on-line expansion, on-line maintenance, and fault tolerance. This attracted researchers to build ADS [16, 17].

Hormone-inspired DHM (digital hormone model) [8] extends Turing's reaction-diffusion model by considering not only the interplay between reactions and diffusions, but also the network topological structure around each robot, the local sensory and actuator states, and the movements of individual robots. Ant colony, schooling fish, and nectar source selection by honey bees are examples of biological self-organization systems. A simulation of task allocation in

the multirobot system using DHM for optimizing the cost of robot system is proposed by Li et al. [18]. Conservation and effective prolong lifetime of WSN are established by hormone-based distributed clustering algorithm [19]. Inspired from bioendocrine systems, a tracking strategy is devised in [20]. Hormone messages are used in sensor networks to coordinate sensor nodes for mobile target tracking in a self-adaptive way.

Reference [21] defines a sensor network architecture which allows sensor nodes to autonomously adapt their duty cycles for power efficiency and responsiveness of data transmission, to collectively self-heal (i.e., detect and eliminate) false positives in their sensor readings, and to be lightweight. This method assumes a regular grid deployment structure which is not always possible in remote areas. Authors of [22] address node localization using particle swarm optimization (PSO) and bacterial foraging algorithm (BFA). This method is computationally expensive due to its iterative nature. Authors of [23] report a design of a distributed sampling scheme referred to as the virtual sampling scheme with a fixed sampling period.

The proposed work is an example of a self-organizing wireless sensor network which is partially inspired from the foraging in bees and has been theoretically described using a digital hormone model [8]. The existing extensions of DHM [18–20] are modeling spatial behavior of static robots, whereas the proposed extended DHM is used for modeling a self-configuring system of static and mobile robots based on spatio-temporal environment and tasks. Unlike DHM proposed by Shen et al. [8], EDHM uses lazy cascading to avoid message flooding. The rules are so defined that different nodes will do different actions with the same hormone message as it depends on the current state of the node. This avoids multiplexing of communication, sampling, and locomotion and makes the node simple. The novelty of the proposed method is its autonomous behavior with adaptive spatiotemporal sampling feature.

The proposed model (extended DHM) behaves like natural systems, executes various rules, based on the local information obtained through sensors. The individual nodes communicate with each other through hormone exchanges. The model shows analogies that can be applied to spatiotemporal sensing and form the basis for the theoretical framework. Due to the judicious combination of self-organization based on natural systems and DHM, EDHM maintains a neat balance between the amount of local and distributed processing, taking into account the task deadlines and available computation and communication resources.

## 3. Problem Description

The aim of the proposed work is to design and implement a sensor network that uses mobile nodes to sample a region for reconstructing the spatiotemporal signal.

A Digital hormone model (DHM) [8] encapsulates dynamic networks of mobile nodes or robots that use hormone-like messages to communicate, collaborate, and

achieve their goals. The hormones received by a node influence but do not determine their local behaviors, as it is the function of both the type of hormone and local state of the node. Thus, different nodes react differently to the hormones despite the fact that all are running the same protocol. The DHM consists of three components: a dynamic network of mobile nodes, a Probabilistic Function for individual node behavior, and a set of rules and equations for hormone reaction and propagation.

**3.1. Dynamic Network of Mobile Nodes.** Distributed control of mobile nodes is a difficult task as the global behavior of the network should emerge as a result of local actions. Therefore, we propose the bio-inspired digital hormone model for spatiotemporal sampling of sensing space. In this model, the nodes are considered as biological cells that communicate and collaborate via hormones and execute local actions via receptors. The model can be visualized as a system having 3 basic components which makes it dynamic, probabilistic, and fault tolerant as in the case of a biological hormone. This model is used for implementing event-based relocation of sensors in a distributed manner.

- (i) Each node has a set of “connectors” through which it can connect to other nodes. In a wireless network, “connectors” mean connecting link (1). These connectors or connecting links dynamically change with the movement of nodes.
- (ii) A set of probabilistic “receptors” or functions allows individual nodes to take actions based on the current state of the node and received hormone (2).
- (iii) The nodes interact within a cell of radius “ $R$ ” directly through relayed “messages”.

A dynamic network of mobile nodes is a formal representation of the traditional wireless sensor network that has been deployed for spatiotemporal sensing. The term dynamic emphasizes the impromptu links that form as mobile nodes move or nodes fail or come up at a time instance  $t$ . Each node has a set of connectors through which nodes dynamically connect with each other.

In this model, connectors simply denote a radio link that comes up whenever a probing node transmits a hormone and a listening node receives that hormone. Note that the links are formed only between a transmitting node and a listening node within its transmitting range. Let  $N_t$  denote the set of active nodes at time  $t$  and  $L_t$  the set of links, then  $DN$  can be defined as

$$DN \equiv (N_t, L_t). \quad (1)$$

Through the  $DN$  nodes can exchange hormones. In our model, a hormone generated at a node is propagated on all the links of that node and it is not propagated further.

**3.2. Function for Node Behavior.** Each node runs a protocol that identifies its behavior based on the current state of events. A mobile node selects its actions  $B$ , based on a probability

function  $P$ , that is, trained on three local factors: the state information  $S$ , the sensor data  $D$ , and the hormone level  $H$ :

$$P(B | S, D, H). \quad (2)$$

The actions include commands to transmit a hormone, move to another location, or switch sensors on/off. The function  $P$  simulates the hormone receptors found in various biological organisms. Although  $P$  is local, yet it greatly influences the global behavior.  $P$ , in fact, is the key to achieving self-organization in the network. The rules for self-organization of mobile sensor nodes for implementing reconstruction of a spatio-temporal signal are proposed in the next section using an extended digital hormone model framework.

## 4. Extended Digital Hormone Model (EDHM)

Sensor networks have been used for a variety of applications one such being spatio-temporal sampling of an environment. It has been previously brought out that such kind of sampling is analogous to foraging activities of bees. Taking that analogy further, we present a tailored, extended digital hormone model (EDHM) that succinctly captures the rules required for spatio-temporal sampling, in a theoretical framework. This model defines the rules for collaboration among mobile nodes for sampling a region in space and time, in an energy efficient manner. We further look at the conditions under which our model will serve its purpose of being energy efficient and also see how can a spatio-temporal signal be reconstructed using this method.

**4.1. Foraging Activities of Honey Bee.** honey bees have been found to focus their foraging activities around more profitable nectar sources. A collaborative effort of several bees maintains a steady flow of food supply in the bee hive. It has been verified that the foraging activities emergent at the colony level occur without any central leadership or a global rule. The overall nectar source election process can be understood with respect to the behavior of individual bees [24].

The following steps are used for the collaboration of honey bees.

- (1) Honey bees are divided into employed and unemployed forager.
- (2) The employed forager will communicate by performing a waggle dance to share the knowledge (quality, distance, and route) about the food source with other bees.
- (3) Unemployed forager observing the dance has an opportunity to become well informed about the various food sources exploited by their colony.
- (4) Based on the observed dance, the unemployed forager decides whether to start foraging that particular source or to wait for other dances. It has been deduced that the decision to forage is based on the information

received during a small time frame and as such does not involve an accumulation of knowledge over successive dances by different bees.

To implement the above steps in mobile robots, for sensing signal in a collaborative fashion, we considered the following assumptions and framed rules based on these assumptions.

#### 4.2. Assumptions

(A1) The state of mobile robots is classified as Probe, Listen, and Move. The state information of nodes is represented by a variable  $S$ . In the probe state the node takes samples of environmental data through its sensors and conditionally transmits hormones. In the Listen state, a node turns off its sensors and listens to its neighbors for possible hormone transmissions. The Move state captures the locomotion of the node from an initial coordinate to a final one.

(A2)  $D$  refers to both the set of sensor readings  $r_i(t_x)$ , from sensor  $i$  of node at sampling instance  $t_x$  and the remaining battery  $batt$ , where

$$batt \leq 1. \quad (3)$$

(A3) All robots have short range wireless communication facility (either RF or infrared) and can talk to robots that are in proximity.

(A4)  $H$  denotes the hormone that is transmitted from a probing node to a listening node. Each  $H$  is a three tuple containing the  $(x, y)$  coordinates of the transmitting node as well as the sensor reading  $r_i(t_x)$ .

(A5) We assume that a node remains in the Probe or Listen states for a time  $T$ , before executing  $P$ . The node can be in the Move state as long as it does not reach its target location.

(A6) It is also assumed that each node knows its location in a global co-ordinate system, and that this information is transmitted along with the sensor reading whenever a hormone  $H$  is propagated. The model does not require any global node identifiers.

(A7) The value of  $r_i$  can fall in three regions as shown in Figure 1.  $r_i$  falls in the region  $L$  if  $r_i \leq LTH_i$ ,  $M$  if  $r_i$  in between  $HTH_i$ , and  $LTH_i$  and  $H$  if  $r_i \geq HTH_i$ , where  $LTH_i$  and  $HTH_i$  are lower threshold and upper threshold values set for  $r_i$ . Let  $MAX_i$  be the maximum possible value of sensor reading  $r_i$ .

#### 4.3. Actions

(B0) (State change rule)

If  $S = \text{Probe}$  AND  $r_i(t_x) < LTH_i$  for entire  $T$ , then  $S = \text{Listen}$

OR

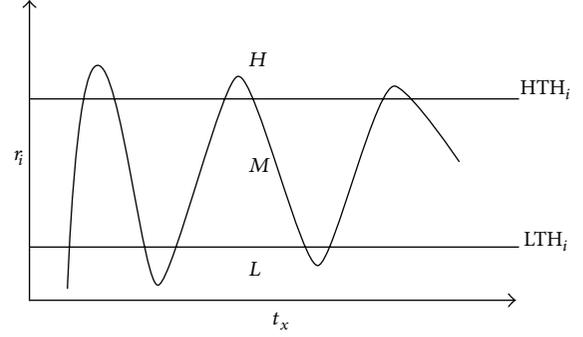


FIGURE 1: Sensor-reading regions.

If  $S = \text{Listen}$  AND  $H = 0$  for entire  $T$ , then  $S = \text{Probe}$  where  $LTH_i$  is a threshold parameter for sensor  $i$ .

(B1) (Random Move rule) if (B0) has been executed continuously for more than a number of times, then a random vector  $\mathbf{R}$  is calculated that determines the direction and distance to move. The target point  $(x_t, y_t)$  is such that distance  $d_{i,t}$  between the initial and final coordinates is proportional to  $batt^\alpha$ , that is,

$$|\mathbf{R}| \propto d_{i,t} \propto batt^\alpha, \quad (4)$$

where,  $\alpha \geq 1$ . We took  $\alpha = 1$  for all calculations. The direction to Move is, though, randomly determined. The node finally sets  $S = \text{Move}$ .

(B2) (Hormone transmission rule) A node with  $S = \text{Probe}$  transmits a hormone  $H(x_i, y_i, r_i(t_x))$  if there is zero-crossing between  $r_i(t_{x-1})$  and  $r_i(t_x)$

or

$$r_i(t_x) > HTH_i. \quad (5)$$

Using this rule, we are doing a differential sampling. If there is a change in reading (positive or negative), a hormone message is generated and transmitted. This reduces redundancy.

(B3) (Gradient Move rule) This rule is particularly important as it significantly contributes to macroscopic self-organization through microscopic actions.

A node in  $S = \text{Listen}$  accumulates all the hormones  $H_k$  it receives for the duration  $T$ .

After that, the node calculates gradient  $g_k$ , towards each of the nodes in its vicinity that transmitted the hormone,

where

$$g_k = \frac{(r_k(t_x) - r_i(t_x))}{d_{i,k}}. \quad (6)$$

Based on the  $g_k$ ,  $g_k^{\max}$  is determined for all nodes within the communication range of  $k$ th node at time  $t_x$ , and a locomotion vector  $\mathbf{L}$  is calculated. Like  $\mathbf{R}$ ,

$$|\mathbf{L}| \propto batt^\alpha. \quad (7)$$

The direction is, however, towards the node with a maximum gradient. The node then sets  $S = \text{Move}$ . This rule helps in adaptive sampling.

(B4) (Destination arrival rule)

If  $S = \text{Move}$ , then  $\{S = \text{Probe if } (x_i, y_i) = (x_t, y_t) \text{ else } S = \text{Move}\}$ .

(B5) (Adaptive sampling rule) This rule is the key to saving energy while taking sensor readings. It defines how often to take samples during the sampling period  $T$  when  $S = \text{Probe}$ . A node predicts when to take the next sample based on the past values of sensor readings. The next sampling instance  $t_{x+1}$  is given by

$$t_{x+1} = t_x + (\text{MAX}_i - r_i(t_x)) \frac{(t_x - t_{x-1})}{(r_i(t_x) - r_i(t_{x-1}))}. \quad (8)$$

Here,  $0 < t_i \leq T$ . The value of sampling interval is directly proportional to the ratio of time interval and difference in sensor reading between two consecutive samples and also to variation with  $\text{MAX}_i$ . Hence, the sampling interval is decided adaptively which further optimizes energy consumption.

The sensor will be moving in a direction where the

**4.3.1. Action Execution Probabilities.** We define the following probabilities for classifying  $r_i(t_x)$  into three regions formed between  $LTH_i$  and  $HTH_i$ , assuming a uniform distribution of values.  $\text{MAX}_i$  denotes the maximum value returned by sensor  $s_i$ :

$$\begin{aligned} P_L &= \frac{LTH_i}{\text{MAX}_i}, \\ P_M &= \frac{HTH_i - LTH_i}{\text{MAX}_i}, \\ P_H &= \frac{\text{MAX}_i - HTH_i}{\text{MAX}_i}. \end{aligned} \quad (9)$$

$P_L$ ,  $P_M$ , and  $P_H$  denote the probability that  $r_i(t_x)$  falls in the region  $L$ ,  $M$ , and  $H$ , respectively, defined in Figure 1, also,

$$P_L + P_M + P_H = 1. \quad (10)$$

We next define a probability function  $P(H_{Tx})$  for a node to transmit a hormone, based on the rule (B2):

$$P(H_{Tx}) = P(S = \text{Probe}) [2P_L P_M + P_H]. \quad (11)$$

The probability of a node being in Probe, Listen, or Move states is defined in the following equations:

$$P(S = \text{Probe}) = [1 - P(S = \text{Move})] [1 - P(H_{Tx})]^m, \quad (12)$$

where  $m$  is equal to the number of neighbors, that is, nodes within its communication range:

$$\begin{aligned} P(S = \text{Listen}) &= [1 - P(S = \text{Move})] [1 - P(S = \text{Probe})], \\ P(S = \text{Move}) &= P(B1) + P(B3). \end{aligned} \quad (13)$$

Also,

$$P(S = \text{Probe}) + P(S = \text{Listen}) + P(S = \text{Move}) = 1. \quad (14)$$

This means that a node can be exactly in one state at any given time.

Assuming  $P(S = \text{Move}) > 0$ , and considering  $P(S = \text{Probe})$  and  $P(S = \text{Listen})$  to be equally likely implies that

$$P(S = \text{Probe}) \approx P(S = \text{Listen}) < 0.5. \quad (15)$$

Based on the above equations, we define the probability of each action, (B0)–(B5), being executed:

$$\begin{aligned} P(B0) &= [1 - P(S = \text{Move})] [P_L + (1 - P(H_{Tx}))^m], \\ P(B1) &= P(B0)^\delta, \end{aligned} \quad (16)$$

$\delta$  is an implementation-specific constant for specifying how long should a node wait at a point before moving to a new, random, location in case of low sensor activity and no received hormones:

$$\begin{aligned} P(B2) &= P(H_{Tx}), \\ P(B3) &= mP(H_{Tx}) [1 - P(S = \text{Move})], \\ P(B4) &= P(S = \text{Move}), \\ P(B5) &= \beta P_H P(S = \text{Probe}), \end{aligned} \quad (17)$$

where  $\beta$  is an adaptive value depending on the frequency of events.

In (B5), the samples are taken more frequently as  $r_i(t_x)$  approaches  $HTH_i$ , therefore, we assume that the average number of times (B5) is executed and is directly proportional to  $P_H$ .

The rule (B0) ensures that a node alternatively probes or listens. (B1) prevents a node from being stationed at a point in case the sensors denote low activity for too long. (B2) propagates the information from a node to another and is the basis for internode communication. (B3) is generally a consequence of another node executing (B2), and it helps in accumulating more nodes towards regions with high sensor activity.

Since our objective is to probe with reference to the threshold, that is, for sensor values above a threshold, (B2) and (B3) together form an effective way for reconstructing the spatiotemporal signal at a global level through simple, local interactions. (B4) simply ensures that a node does not respond to  $D$  or  $H$  as long as it is moving. (B5) is another important rule that makes energy-efficient sampling possible by varying the sampling rate based on the stream of samples collected so far. Specifically, rules (B2)–(B3) guide sampling in space, whereas (B5) guides sampling in time. It should be noted that the above rules are designed so as not to do multiplexing of tasks such as sampling, communication, or locomotion. The objective behind this is to reduce

TABLE I: Probability of execution.

Action/state	S = Probe	S = Listen	S = Move
B0	$P_L$	$[1 - P(H_{Tx})]^m$	0
B1	$P_L^\delta$	$[1 - P(H_{Tx})]^{m\delta}$	0
B2	$P(H_{Tx})$	0	0
B3	0	$mP(H_{Tx})$	0
B4	0	0	1
B5	$\beta P_H$	0	0

the instantaneous power consumption and also to simplify the model. It should be noted that what action is selected at a time is mainly dependent on the current state of node and environmental factors such as received hormones and sensor readings. A summary of action execution probabilities for each action/state pair is listed in Table I. When a node is in the Listen state,  $P(H_{Tx})$  represents the probability of a neighbor to transmit a hormone.

**4.4. Rules for Hormone Reaction and Propagation.** The propagation of hormone occurs in a two-dimensional space and is governed by standard laws for radio signal propagation. Although in biological organisms, hormones diffuse through a cellular interaction; in case of mobile sensor nodes, the radio transmission reception helps in the hormone diffusion. The gradient calculated towards each hormone transmitting node aptly represents the concentration  $C$  of hormone and therefore  $C$  is a function of both relative difference in sensor readings and distance between the nodes, that is,  $C(\Delta r_i, d_{i,k})$ . The DHM proposed by Shen et al. [8] establishes  $C$  as a function of time, but, in our case, we simply assume that a hormone  $H$  is of merit to a node only up to the next decision-making step.

Our approach, though a bit discreet, still serves the purpose of guiding nodes towards regions of high sensor activity; as hormones are transmitted every time, conditions for rule B2 are met. Thus, if a node missed a transmission, it might receive it the next time it enters the Listen state. The hormone  $H$  in our model is analogous to the activator hormone  $A$ , though the inhibitor hormone  $I$  is implicit and can be assumed, for the sake of completeness, to be virtually generated on timeouts. We shall like to mention here that classical hormone-based models generally retransmit a received hormone to generate a cascading motion with immediate effect. Our approach, however, allows only fresh hormone transmissions (i.e., no forwarding of hormone messages) based entirely on local sensor readings, since we are dealing with a time-varying signal and it may be possible that by the time a node reaches towards a target point the signal has died out. This effect will be more pronounced as more and more nodes, situated multiple hops away, move towards the target region. Hence, our model results in lazy cascading wherein the propagation of hormones is deferred until the receiving node starts probing at the new target point. This method also helps to reduce the overall energy consumption of the network.

**4.5. The DN-EDHM Control Loop.** Each of the nodes can be assumed to occupy a location in a 2-D grid space, where each grid box is equal to the resolution of the localization system deployed. At powerup, the nodes initialize their states  $S$ , to either Probe or Listen, and execute the following loop.

- (1) Select actions by executing  $P(B | S, D, H)$ .
- (2) Execute the selected action. This includes sampling, hormone transmission/reception, and locomotion.
- (3) Go to Step (1).

DN-EDHM control loop has fewer steps than DHM because we have coupled hormone generation, propagation, and reaction with action steps. Another major difference with DHM is that it is not guaranteed that all the nodes will be executing the same step in the control loop, since B1 and B3 may take different times to execute in different nodes based on  $\mathbf{R}$  and  $\mathbf{L}$ .

## 5. Reconstructing the Spatiotemporal Signal

Sensor networks are characterized by the dense deployment of sensor nodes that continuously observe physical phenomena. Due to high density in the network topology, sensor observations are highly correlated in the space domain. Furthermore, the nature of the physical phenomena constitutes the temporal correlation between each consecutive observation of a sensor node. These spatial and temporal correlations along with the collaborative nature of the sensor networks make it suitable for spatio-temporal sampling of physical fields. The dense deployment of sensors increases redundancy which makes the system energy craving. Accurate temporal reconstruction of signals depends on frequency of temporal sampling. EDHM reduces redundancy through rule (B<sub>2</sub>) and (B<sub>3</sub>) and optimizes temporal sampling frequency through rule (B<sub>5</sub>). Rule (B<sub>3</sub>) decides the direction of motion of nodes towards maximum signal level.

The model proposed above demonstrates how self-organization and distributed actuation can be achieved using simple rules that process only local information. Even though the nodes carry out the sensing task without any intervention, yet the readings taken by the network as a whole need to be collated for further analysis. This shall require the nodes to periodically (or in response to a query) transmit the sensor readings to a base station.

Watching the data sent by geographically spread nodes is like watching a “sensor movie” [25], since the readings of a sensor correspond to pixel values of varying intensity in an *environment image*, and watching a sequence of such images (or frames) gives the impression of a video of sampled values.

(1) *PREMON Model.* Reporting the values to a base station is, nevertheless, a costly task and therefore should be minimized. Reference [25] proposes a *prediction-based monitoring (PREMON)* model, where the base station computes a “prediction model” for a sensor, and the sensor transmits a value if it significantly differs from the value computed

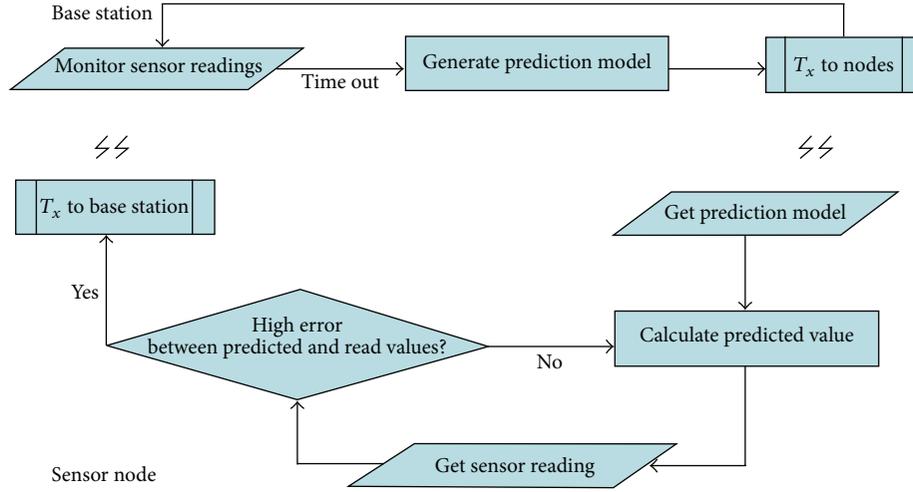


FIGURE 2: PREMON loop.

using the prediction model (see Figure 2). The prediction model is valid only for a limited time interval, after which it should be recomputed. In the proposed model, the hormone messages are sent only if there is a significant difference in the consecutive readings. The sampling frequency is also adaptively varied according to the difference in the sensor readings. Hence, the prediction model can be easily applied in this model.

In view of our case, the prediction model can be classified as follows.

- (i) *Spatial*. When the reading at sensor  $s_i$  is a function of the reading at sensor  $s_k$  during the time slot  $t_x$ .
- (ii) *Temporal*. When the reading at sensor  $s_i$  at a time  $t_x$  is a function of its reading at a previous time slot  $t_{x-j}$ .
- (iii) *Spatiotemporal*. When the reading at sensor  $s_i$  at a time  $t_x$  is a function of reading at  $s_k$  in a previous time slot  $t_{x-j}$ .

The *PREMON* model can be suitably applied in our case also, the only difference being that [25] uses a setup of static nodes, as compared to mobile nodes in our approach. Figure 3 illustrates that by superimposing snapshots taken at different time intervals, we can construct the environment image at a base station. The coloured grids specify the information collected from that grid. Clearly, the mobile nodes provide a dynamic coverage of the spatial region (the number of coloured grid is more), whereas static nodes cannot provide dynamic coverage. Thus, by combining *PREMON* with mobile nodes and *EDHM*, we can obtain a far more accurate spatio-temporal reconstruction at reduced energy costs.

**5.1. Energy Savings through EDHM.** In a conventional, fixed rate, sampling model, sensor readings are categorised as those

above or below a certain threshold  $TH_i$ . Let us assume that the sampling rate in this case is proportional to  $P_T$ , the probability of a value being above  $TH_i$ :

$$P_T = \frac{MAX_i - TH_i}{MAX_i}, \quad (18)$$

where,  $LTH_i \approx HTH_i < TH_i \approx MTH_i < MAX_i$ .  $MTH_i$  is the *maximum threshold value*.

A simple hormone or message transmission rule in this case can be to transmit whenever  $r_i(t_x) > TH_i$ , where the probability of transmitting a hormone is defined by

$$P(H'_{Tx}) = P(1)P_T, \quad (19)$$

thus, the hormone transmission rule for the conventional model becomes

$$P(B2') = P(H'_{Tx}) \quad (20)$$

and that for the sampling is

$$P(B5') = \beta P_T P(1). \quad (21)$$

Here,  $P(1) = 1$  denotes that a node is always probing.

The energy savings at a node can then be calculated by comparing the relative execution probabilities of actions B2 and B5 in EDHM with  $B2'$  and  $B5'$  in a conventional model. The value of  $RP_i$ , defined below is an indication of how often these actions are executed *vis-a-vis* the conventional model, thus, giving an indication, how often the transmission and sampling costs are incurred. A value less than 1 shall denote

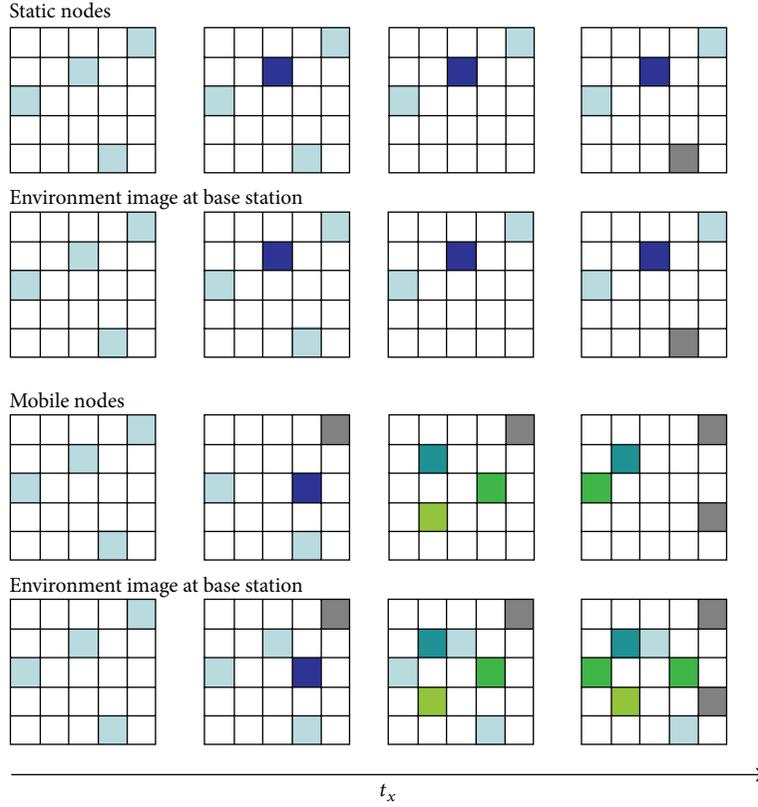


FIGURE 3: PREMON using static and mobile nodes.

energy savings in our approach, whereas a value greater than 1 means that our model is not energy efficient. Consider

$$\begin{aligned}
 RP_i &= \frac{P(B2)}{P(B2')} + \frac{P(B5)}{P(B5')} \\
 &= \frac{P(H_{Tx})}{P(H'_{Tx})} + \frac{P_H P(S = \text{Probe})}{P_T P(1)} \\
 &= \frac{P(S = \text{Probe}) [2P_L P_M + P_H]}{P(1) P_T} + \frac{P_H P(S = \text{Probe})}{P_T} \\
 &= \frac{P(S = \text{Probe}) [2P_L P_M + P_H]}{P_T} + \frac{P_H P(S = \text{Probe})}{P_T} \\
 &= 2P(S = \text{Probe}) \frac{[P_L P_M + P_H]}{P_T} \\
 &\approx 2P(S = \text{Probe}) \frac{[P_L (P_M + 1)]}{P_T}.
 \end{aligned} \tag{22}$$

We know that,  $P(S = \text{Probe}) < 0.5$ , that is,  $2P(S = \text{Probe}) < 1$ . This means that for  $RP_i$  to be less than 1,  $[P_L (P_M + 1)]/P_T$  should be less than 1. By suitably choosing the threshold parameters  $LTH_i$  and  $HTH_i$  based on  $TH_i$ , we

can guarantee  $RP_i$  to be less than 1, that is, saving energy. In this case if we assume  $P_L \approx P_M$  we get

$$RP_i \leq P_L + \frac{P_L}{P_T}. \tag{23}$$

The condition for  $RP_i$  to be less than 1, in terms of  $P_L$  and  $P_T$ , then becomes

$$P_L < \frac{P_T}{P_T + 1}. \tag{24}$$

The above discussion only provides a theoretical insight of the savings possible with a hormone-based model while transmitting messages or sampling at a point in space. More savings are possible if we also consider the fact that a mobile node in this model will move less often than where nodes randomly move about to sample the region. Our experiments show a significant saving in such a case, albeit at the cost of less accurate signal reconstruction, the results of which are discussed in Section 4.

## 6. Results

**6.1. Simulation.** In order to carry out experiments on the power consumption and accuracy in signal reconstruction using the hormone-based model, a simulator called *Shimla*, was designed. Shimla compares the power consumption of nodes, as they move around, by simultaneously executing

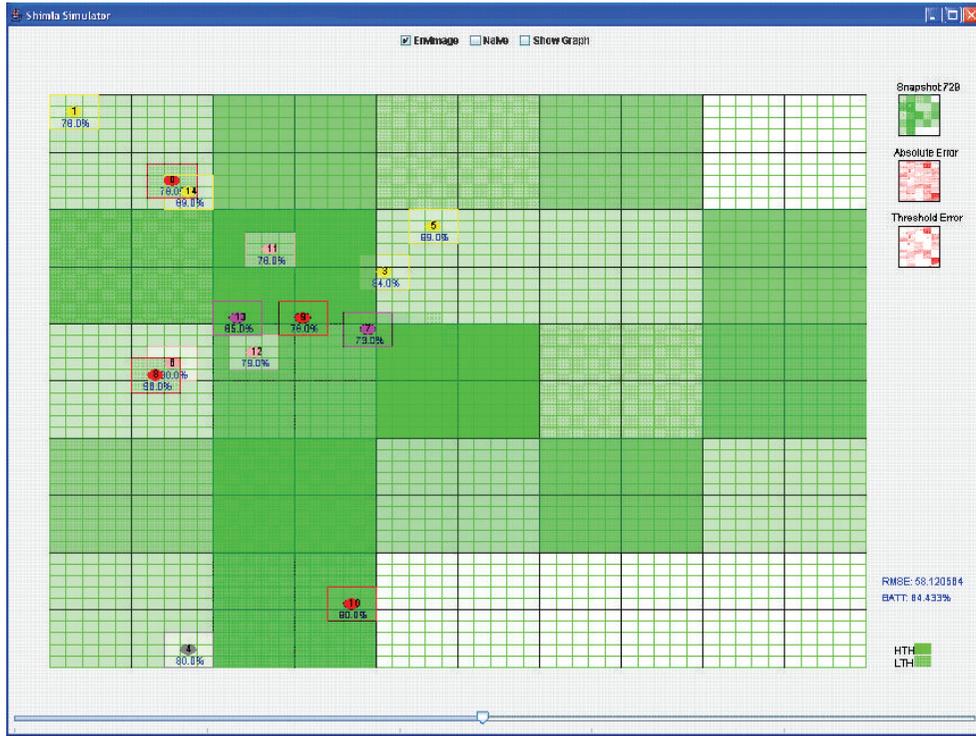


FIGURE 4: Shimla simulator.

a simple probabilistic model and our digital hormone-based model. Shimla is a discrete time simulator. The entire simulation period is divided into a number of epochs, where each epoch represents a fraction of the time  $T$  for which a node is in Probe or Listen state. It also calculates the error in reconstruction of a static and time-varying spatio-temporal signal. Simulations can be made by varying many parameters such as number of nodes, number of epochs, their initial position, and signal. The results are written to a file, that can be accumulated for a number of runs and viewed in a companion utility called RView. The results of simulation under Mirchi and Shimla are discussed in the next section.

We carried out simulations using Shimla simulator as (Figure 4) to find out how does our approach fare in terms of power consumption and how well can it reconstruct a spatio-temporal signal. Experiments were carried out to measure the performance in terms of average battery remaining and root-mean-squared error. Simultaneously, a naive protocol was also executed which served as a benchmark for our model. Due to the inherent randomness in self-organizing systems, for better accuracy, each experiment was repeated 50 times as (Figure 5) for the same signal and the same set of parameters, though at each run the initial location and state of the node were randomly chosen. Based on the compiled data from these experiments, the effectiveness of our model was calculated as a ratio between relative battery savings and relative root-mean-squared error (*RMSE*).

This process was repeated again by increasing the nodes to 2, 3, . . . up to a total of 25 nodes, without changing the signal.

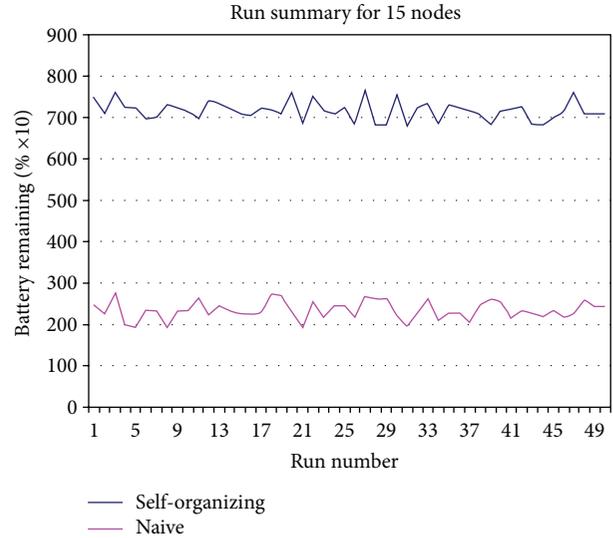


FIGURE 5: Simulation summary for 15 nodes.

Thus, a total of 1250 executions each, of Shimla, were carried out for static and time-varying signal. The environment consisted of a grid of  $50 \times 50$ , with a communication range set to 25 units, and sensor range set to cover an area of 9 grid squares. It should be noted that Shimla primarily calculates the energy expended in locomotion and radio communication, since these two costs are much higher than the cost of probing (see Table 2).

TABLE 2: Power model for mica mot.

Mode	Current	Mode	Current
cpu		radio	
Active	17 mA	$R_x$	7 mA
Idle	3.2 mA	$T_x$ (-20 dBm)	3.7 mA
Power down	103 $\mu$ A	$T_x$ (-15 dBm)	5.4 mA
Power save	110 $\mu$ A	$T_x$ (-8 dBm)	6.5 mA
Standby	216 $\mu$ A	$T_x$ (-5 dBm)	7.1 mA
Extended standby	223 $\mu$ A	$T_x$ (0 dBm)	8.5 mA
Internal oscillator	930 $\mu$ A	$T_x$ (+4 dBm)	11.6 mA
LEDs	2.2 mA	$T_x$ (+6 dBm)	13.8 mA
Sensor board	0.7 mA	$T_x$ (+8 dBm)	17.4 mA
Ower save	103 $\mu$ A	$T_x$ (-15 dBm)	5.4 mA

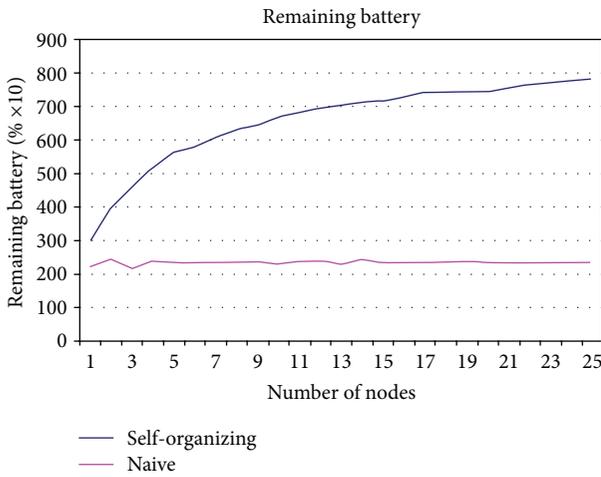


FIGURE 6: Battery consumption while sampling a spatiotemporal signal.

6.2. Execution of EDHM for Spatiotemporal Reconstruction. Executing the hormone-based model in parallel to a naive algorithm (where nodes probabilistically move to sample a region) helped us compare the performance of the model in terms of power consumed and error in a signal reconstruction. The results (shown below) demonstrate that the nodes self-organize themselves to decrease the per node power consumption and reduce the error in a signal reconstruction.

6.2.1. Energy Consumption. Figure 6 compares the average battery remaining after 1500 epochs (over 50 runs) the against number of nodes with a naive algorithm, for both static and time-varying signal. In both cases, as the number of nodes is increased the average battery remaining increases implying a decrease in the power consumed per node, for EDHM. If we look at the power remaining in the naive case (conventional fixed rate sampling), we find that it remains close to 20% irrespective of the number of nodes, even though the naive case does not expend any energy in radio communication. The increase in battery remaining, as the number of nodes increases, establishes the fact that nodes do collaborate

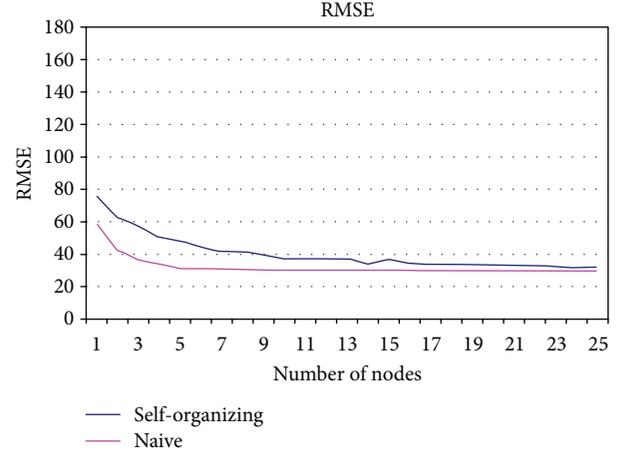


FIGURE 7: RMSE in spatiotemporal signal.

among themselves by sharing the task. This collaboration, therefore, achieves our goal of energy efficiency.

6.2.2. Error in Reconstruction. Besides saving energy, the collaboration among nodes also decreases the error in signal reconstruction, as visible in Figure 7. Again, compared to EDHM model, in the random sampling approach the RMSE remains roughly the same even as the number of nodes increases. The self-organizing nature of the proposed model helps us to achieve both our goals of improving energy efficiency and reducing RMSE error. It should also be noted that even though the random approach produces less error in reconstructing the signal, the difference from our model decreases, as the number of nodes increases. This is mainly because the random strategy results in oversampling of the region, whereas the nodes in our case coordinate their activities by dividing the sampling task.

6.3. Effectiveness of the Model. It is clear from the above results that the proposed model offers a tradeoff between energy efficiency and error in a signal reconstruction. The model can be compared (naive protocol whose characteristics are explained in Section 5.1) on the basis of how much energy it saves and how much error does it produce. The aim of the proposed work is to maximize energy savings while reducing the error. To gauge the model on these two metrics, we define the utility  $U$  of the model as the ratio of relative battery savings (between our model and the naive algorithm) to relative root-mean-squared error, for a given set of parameters:

$$U = \frac{R_{\text{batt}}}{R_{\text{rmse}}}. \quad (25)$$

Our aim, now, is to find those set of parameters that maximize the utility. In the proposed model, the behavior of a node is mostly affected by the sensor readings and the hormone receptions, both of which are dependent on external factors. Figure 8 shows how the utility of model increases as the number of nodes are increased.

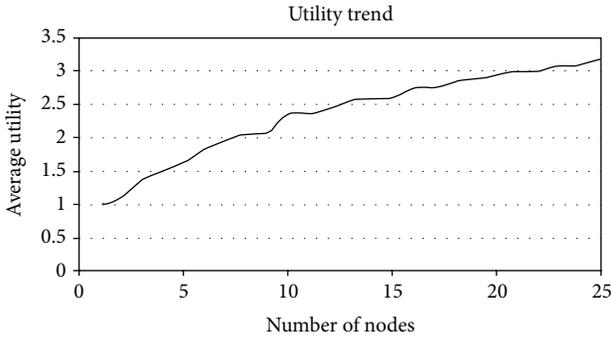


FIGURE 8: Utility of hormone-based model.

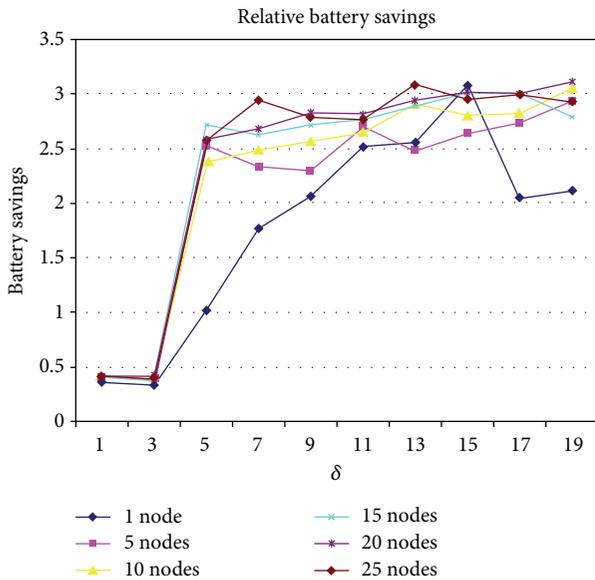


FIGURE 9: Relative battery savings.

For a given node, the only parameter that can be controlled is  $\delta$ , that affects the execution of rule (B1). The parameter  $\delta$  is important because it determines how often a node should do random sampling in case of no sensor or hormonal activity. A low value of  $\delta$  will result in frequent wandering of a node which improves spatial sampling but also costs heavily on the battery (Figure 9). Another parameter that can be controlled, at the network level though, is the number of nodes deployed. We have seen from the results that a larger number of nodes reduce average battery consumption and root-mean-squared error. We used Shimla to calculate the utility  $U$  for a given number of nodes at a particular value of  $\delta$  (Figure 10) and also averaged it over 50 runs. The value pairs where the utility is more than 1 indicate that the model performs better than the naive model, if battery savings are considered as equally important as reducing the error in signal reconstruction. For a grid size of  $50 \times 50$ , the maximum utility is obtained at a deployment of 20 to 25 nodes with  $\delta$  between 13 and 17. It should be noted that the value of  $\delta$  is expressed in terms of number of epochs.

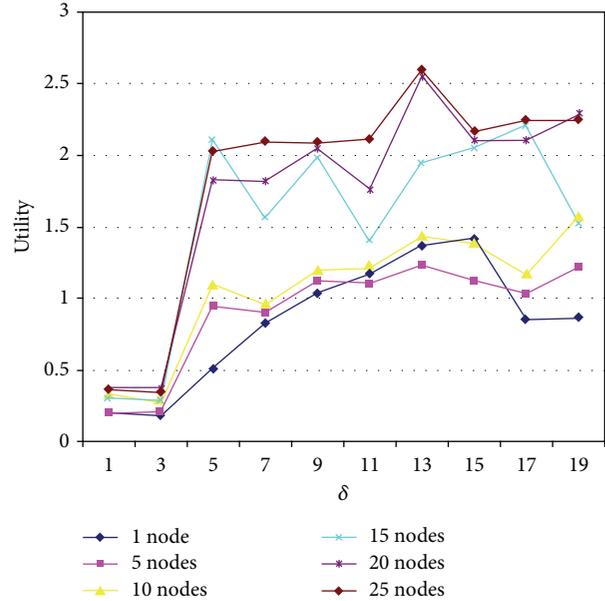


FIGURE 10: Correlation between delta and utility.

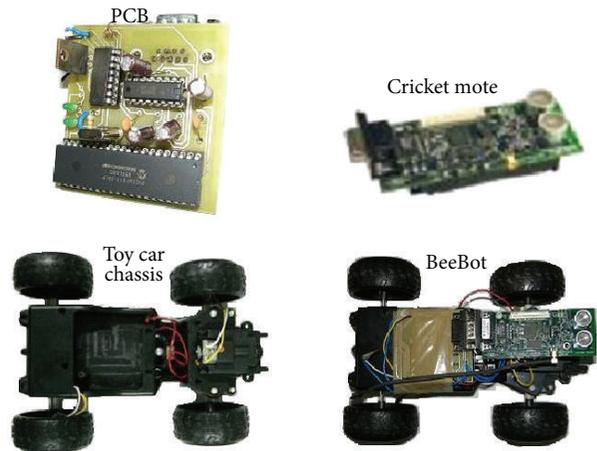


FIGURE 11: From a toy car to a mobility platform.

At a given number of nodes, for lower values of  $\delta$ , rule (B1) executes too frequently resulting in decreases in relative battery saving and hence decreasing the utility. On the other hand, at higher values the execution of (B1) is deferred too late, causing it to idle and consequently miss events, hence, increasing the error in signal reconstruction and therefore decreasing the utility. It can also be seen in Figure 9.

Thus, it can be concluded that an effective deployment requires higher number of nodes with combined sensor coverage of around 10% of the total area and an idling time ( $\delta$ ) of the order of lower tens of the probing epoch.

**6.4. Hardware Implementation.** In order to verify the feasibility of an extended hormone-based model for spatiotemporal sensing, we implemented the project in 3 phases. In phase 1, we simulated a basic hormone-based model using

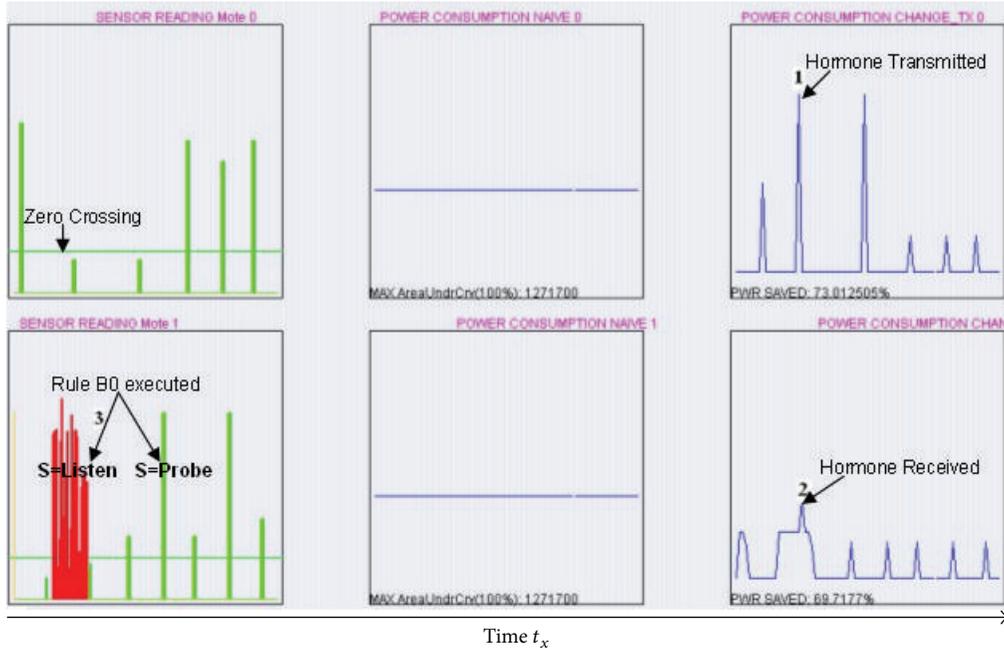


FIGURE 12: Execution of rule (B0).

static nodes for studying the self-organization among nodes, through hormone exchanges. Phase 2 involved implementing the basic model on Mica2-cricket motes. In the last phase, we implemented the complete model with mobile nodes and a localization system. We used Crossbrow cricket motes as sensor nodes. We placed these nodes over a toy car chassis and used PIC microcontroller for driving the car. The whole system works as a mobile sensor node (Figure 11). Experimental set up is as shown in Figure 16.

**6.4.1. Mirchi.** Before the cricket motes were procured, we made a plugin for TinyViz called Mirchi as (Figure 13). It is a Java-based plugin made by extending the simulation environment provided in TinyViz. We designed Mirchi for estimating the energy saved through the use of adaptive sampling, along with bistate operation of a node, where in one state only the sensor was on (with occasional transmissions) and the other only for receiving messages. The plugin demonstrates the execution of rules (B0) and (B5) as (Figure 12) for a pair of two nodes in each others neighborhood. The power model used in Mirchi was that of Mica2 (see Table 2). Based on the feedback from Mirchi, we went ahead with the implementation of the model on cricket motes.

**6.4.2. MirchiBLUE.** MirchiBLUE was built in Java for displaying, in real time, the on-going network activity. Besides, the network's traffic, it displays the tessellation of the region sampled by the nodes. MirchiBLUE supports the light, temperature, pressure, and humidity sensing modalities along with the remaining power level in a node. MirchiBLUE receives data from the network through a cricket mote attached to it through the serial port. MirchiBLUE can also be extended to give commands to the motes in real time.

**6.4.3. Embedded Code.** Three separate softwares were used to implement the system. Firstly, the firmware supplied with the motes was uploaded on three motes which served as *beacons*. This firmware was modified slightly for use as a *Listener*, that ran the hormone model. Finally, a single node was programmed to act as a packet forwarder for the MirchiBLUE application.

**6.4.4. Navigation.** Rules (B1) and (B3) require the node to move to a new target location. Since (B1) makes a random movement, we implemented it by having the node choose a random direction (among forward, back, forward left, forward right, backward left and backward right) and executing it for a time proportional to the remaining battery power. (B3), however, requires an accurate displacement of the node from the initial to the target co-ordinate.

This requires the bot to know its orientation and that of the target relative to it for it to choose the appropriate locomotion command (Figures 14 and 15). The navigation logic is incorporated as rule (B4), and the algorithm for that can be stated as shown in Algorithm 1.

Algorithm 1 iteratively navigates the bot to the target by deciding what direction to move to, from a given point, by calculating the bots orientation.

The implementation of the hormone model has been largely successful, except for the time it takes for a bot to settle down at the target point. In order to implement the model, for deployment in a real world scenario, it would be advisable to use a digital compass and a separate hardware for localization. We developed a coverage panel in Java for displaying, in real time, the actual location of sensor nodes as shown in Figure 17. Three cricket beacons were used for localization of nodes which are represented as black dots

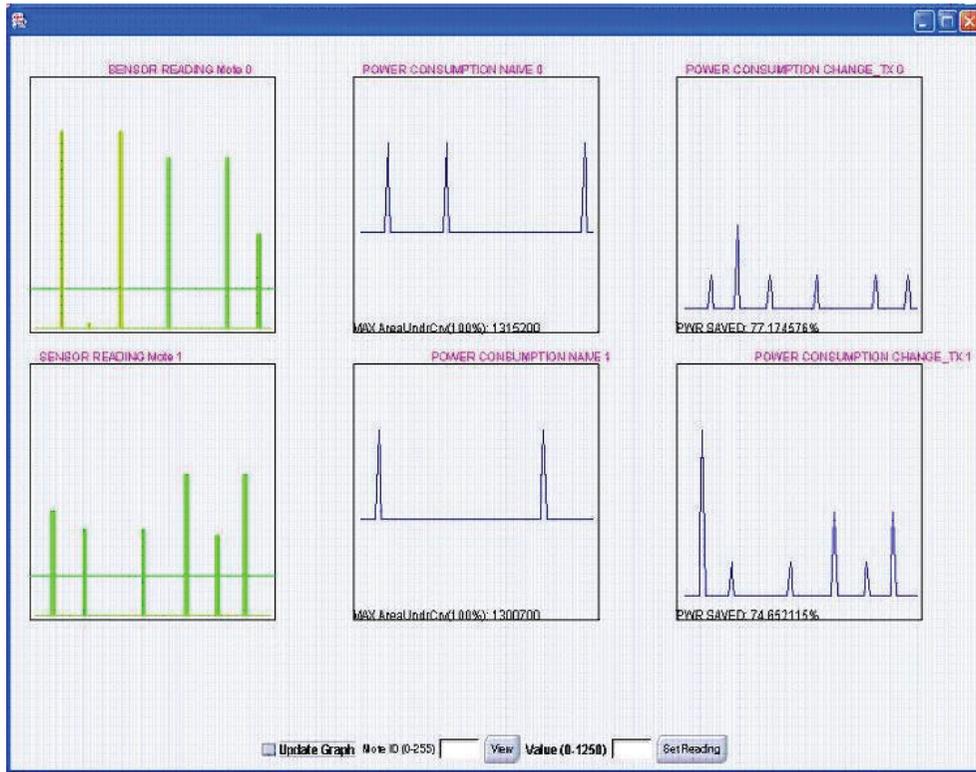


FIGURE 13: Mirchi plugin for TinyViz.

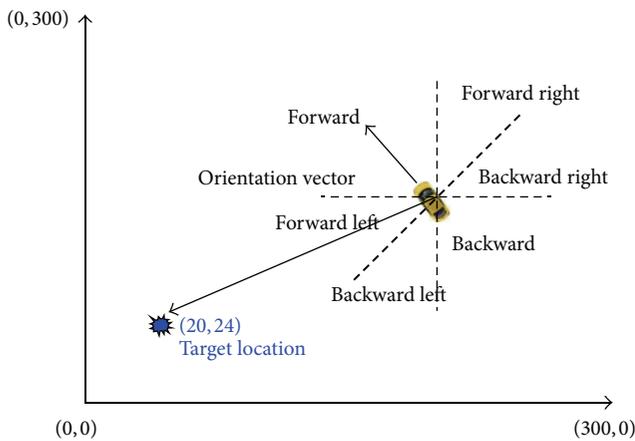


FIGURE 14: Relative orientation of target.

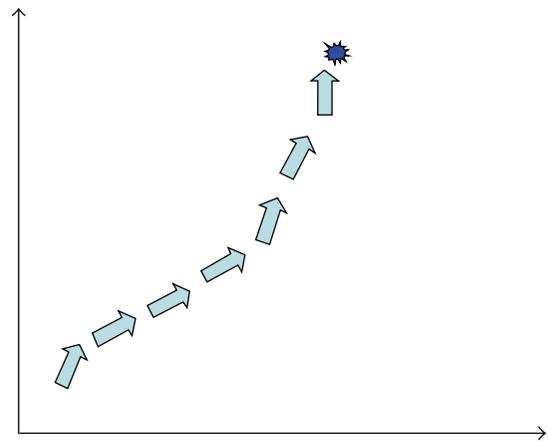


FIGURE 15: Navigation to the target.

and the distribution of nodes after deployment algorithm is shown in Figure 17. We used nesC to embody the structuring concepts of TinyOS operating system. Three separate softwares were used to implement the system. Firstly the firmware supplied was uploaded on 3 motes which served as beacons. This firmware was slightly modified to use as a node which ran our model. Finally, a single node was programmed as a packet forwarder and connected to the serial port to display details in the coverage panel. This packet forwarder acts as an interface between sensor nodes and serial port of computer. We programmed the nodes to handle

self deployment, localization, event detection, and relocation themselves [26].

Figures 18 and 19 give the intermediate location and final positions of mobile nodes after implementation of DHM-based self-organizing algorithm. Figure 20 shows the final location of mobile node after doing the experiment.

**6.4.5. Practical Issues.** While implementing the model, we faced several issues that need to be understood for improving the performance of the model in an actual deployment. These are summarized below.

```

(1) Get Target Location.
(2) Get Current location.
(3) If current location is within a distance  $d_n$  of target then STOP.
(4) Get orientation of the bot.
(5) Calculate the orientation ort of target, relative to the bot.
(6) Select locomotion action act, based on the following:
(6a) If ort < 30 AND ort > -30 then act = FWD
(6b) If ort < -150 AND ort > 150 then act = BACK
(6c) If ort < -30 AND ort > -90 then act = FWD_RIGHT
(6d) If ort > 30 AND ort < 90 then act = FWD_LEFT
(6e) If ort > 90 AND ort < 150 then act = BACK_LEFT
(6f) If ort < -90 AND ort > -150 then act = BACK_RIGHT
(7) Execute action act
(8) Go to step 1.

```

ALGORITHM 1: Navigation algorithm.



FIGURE 16: Experimental set up.

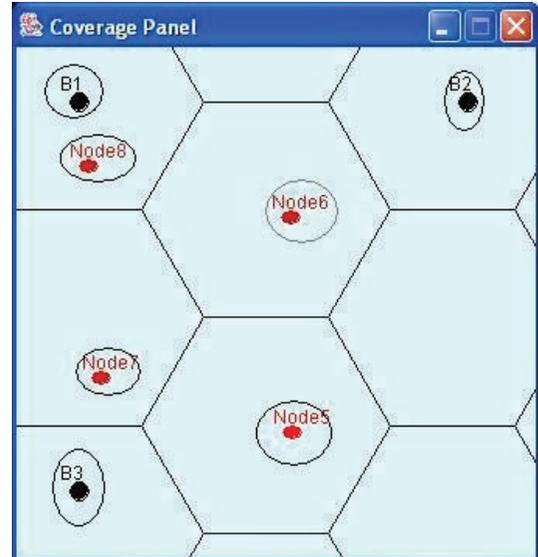


FIGURE 17: Coverage panel.

- (i) *Radio Conflict*. As has been previously brought out, localization uses RF (along with ultrasound) which is also required for transmitting and receiving hormones. This not only leads to a time duration where no useful work is being done (model execution is paused) which can cause missed events and hormones resulting in loss of performance, but also increases channel contention. Having a second radio for communication is feasible if the cost of hardware is not of concern. Another solution can be to synchronize the transmission schedules using protocols such as S-MAC [27].
- (ii) *Localization Time*. Since the localization code is multiplexed with the model on the same processor, the model is paused from executing. Only the possible approach is to either have a separate node running the localization code (on a different frequency) or use another localization system such as GPS (which does not work indoors).
- (iii) *Braking*. Since the BeeBot does not have an instantaneous brake and due to the time required for knowing the coordinates, often the bot overshoots the target region leading to oscillatory behavior. This leads to an increase in the converging time for the navigation

algorithm. The problem can be solved by making the localization hardware separate and also by providing an effective braking mechanism.

- (iv) *Orientation*. Knowing the orientation is another challenging task. The methods discussed in the last chapter are only partly successful as they are either susceptible to errors or require more number of nodes. The best way to get the orientation shall be to use a digital compass as it will provide instantaneous readings with comparatively less hardware cost and power consumption.
- (v) *Beacon Coverage*. Since we have only used three beacons, accurate localization can only be done where the ultrasound pulses from all three beacons reach (Figure 21). More accuracy and wider coverage are possible by using a greater number of beacons.

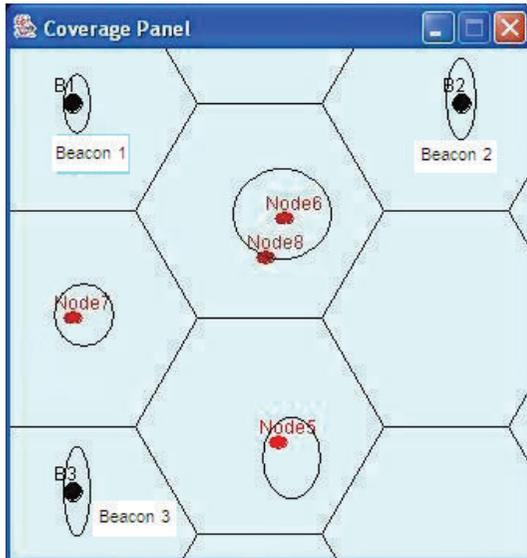


FIGURE 18: Intermediate location of mobile nodes.

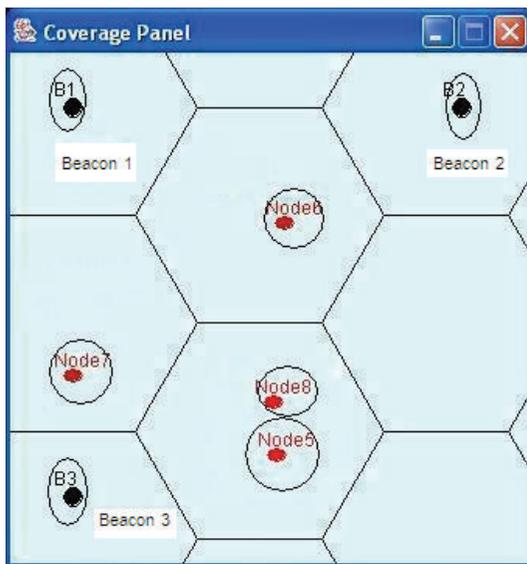


FIGURE 19: Final location of nodes.

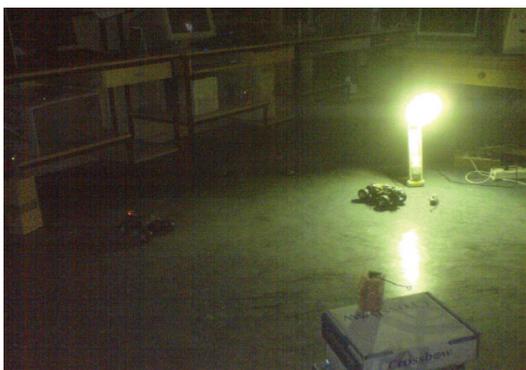


FIGURE 20: Final location of mobile node.

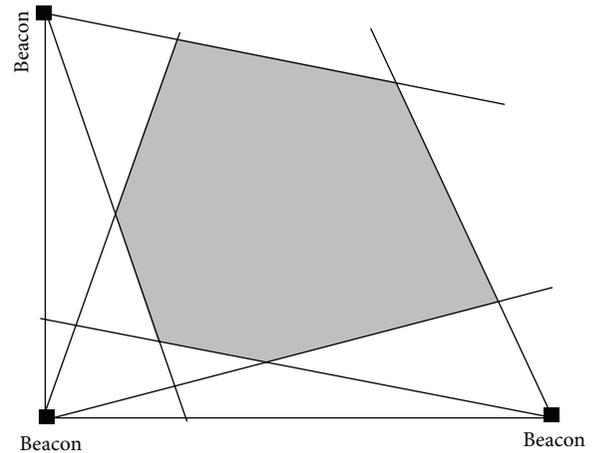


FIGURE 21: Effective localization area.

## 7. Conclusion

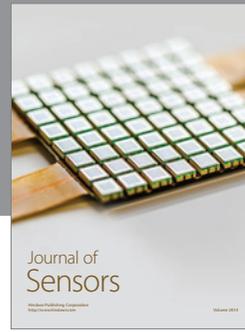
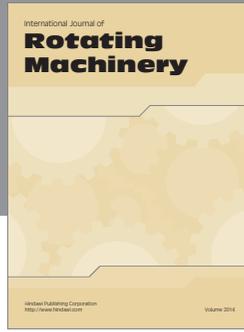
We have seen that EDHM helps to attain balance between the extent to which battery saving is required and the tolerance in reconstruction error. The simulation results exhibit how larger the number of nodes share the task, improving on both the fronts of efficiency and sampling. We shall, however, like to comment that our model is slightly biased towards saving energy than accurately sampling the region. Nevertheless, the utility  $U$  is above one for most of the cases, except when the number of nodes is too less to effectively collaborate. Our experiments also reinforce the view that for a self-organizing system to succeed, the number of individual units should be considerably large.

It can be also seen from the above discussion that simulations are a good instrument for verifying the validity of theoretical proposal, but the actual implementation might require tackling issues that are totally unrepresented (in the theoretical model). Such issues, nevertheless, *must* be resolved for the implementation to perform as expected.

## References

- [1] M. C. Vuran, Ö. B. Akan, and I. F. Akyildiz, "Spatio-temporal correlation: theory and applications for wireless sensor networks," *Computer Networks*, vol. 45, no. 3, pp. 245–259, 2004.
- [2] Q. Huang, C. Lu, and G. Catalin, "Spatio-temporal multicast in sensor networks," in *Proceedings of the ACM Conference on Embedded Networked Sensor Systems (SenSys '03)*, pp. 38–45, 2003.
- [3] M. A. Batalin, "Spatio-temporal multicast in sensor networks," in *Proceedings of the 2nd ACM International Conference on Embedded Networked Sensor Systems (SenSys '04)*, pp. 25–38, Baltimore, Md, USA, November 2004.
- [4] A. M. Turing, "The chemical basis of morphogenesis," *The Philosophical Transactions of the Royal Society B*, vol. 237, no. 641, pp. 37–72, 1952.
- [5] J. D. Murray, *Mathematical Biology*, Springer, Berlin, Germany, 1989.

- [6] Q. Ouyang and H. L. Swinney, "Transition from a uniform state to hexagonal and striped Turing patterns," *Nature*, vol. 352, no. 6336, pp. 610–612, 1991.
- [7] H. Gutowitz, *Cellular Automata: Theory and Experiment*, The MIT Press, Cambridge, Mass, USA, 1991.
- [8] W. M. Shen, P. Will, A. Galstyan, and C. M. Chuong, "Hormone-inspired self-organization and distributed control of robotic swarms," *Autonomous Robots*, vol. 17, no. 1, pp. 93–105, 2004.
- [9] W. M. Shen, B. Salemi, and P. Will, "Hormone-inspired adaptive communication and distributed control for CONRO self-reconfigurable robots," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 700–712, 2002.
- [10] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence from Natural to Artificial Systems*, Oxford University Press, New York, NY, USA, 1999.
- [11] H. van Dyke Parunak and S. Brueckner, "Entropy and self-organization in multi-agent systems," in *Proceedings of the 5th International Conference on Autonomous Agents*, pp. 124–130, Montreal, Canada, June 2001.
- [12] D. Payton, R. Estkowski, and M. Howard, "Entropy and selforganization in multiagent systems," in *Proceedings of the International Conference on Autonomous Agents*, pp. 23–28, Montreal, Canada, 2002.
- [13] H. van Dyke Parunak, "Making swarming happen," in *Proceedings of the Swarming Network Enabled CAISR Conference*, pp. 170–179, Tysons Corner, Va, USA, 2003.
- [14] E. A. Kravitz, "Hormonal control of behavior: amines and the biasing of behavioral output in lobsters," *Science*, vol. 241, no. 4874, pp. 1775–1781, 1988.
- [15] K. Mori, "Autonomous decentralized system technologies and their application to train transport operation system," in *Proceedings of the High Integrity Software Conference*, pp. 75–81, Albuquerque, NM, USA, 1999.
- [16] H. Ihara and K. Mori, "Autonomous decentralized computer control systems," *Computer*, vol. 17, no. 8, pp. 57–66, 1984.
- [17] K. Mori, S. Miyamoto, and H. Ihara, "Autonomous decentralized computer system and software structure," *Computer Systems Science and Engineering*, vol. 1, no. 1, pp. 17–22, 1985.
- [18] X. Li, X. Wang, Y. Lei, and H. You, "A self-organized algorithm based on digital hormone," in *Proceedings of the 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE '10)*, pp. 398–402, August 2010.
- [19] D. Dong, H. You, Y. Zhang, and X. Wang, "A hormone-based clustering algorithm in wireless sensor networks," in *Proceedings of the 2nd International Conference on Computer Engineering and Technology*, pp. 555–559, 2010.
- [20] Y. Zhang, H. You, and X. Wang, "A hormone based tracking strategy for wireless sensor network," in *Proceedings of the IEEE International Conference on Intelligent Computing and Intelligent Systems*, pp. 104–108, 2009.
- [21] P. Boonma, P. Champrasert, and J. Suzuki, "Bisnet a biologicallyinspired architecture for wireless sensor networks," in *Proceedings of the 21st International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '07)*, pp. 17–22, 2007.
- [22] R. V. Kulkarni, G. K. Venayagamoorthy, and M. X. Cheng, "Bio-inspired node localization in wireless sensor networks," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC '09)*, pp. 205–210, 2009.
- [23] L. Xun, T. Shiqi, G. V. Merrett, and N. M. White, "Energy-efficient data acquisition in wireless sensor networks through spatial correlation," in *Proceedings of the International Conference on Mechatronics and Automation (ICMA '11)*, pp. 210–215, 2011.
- [24] S. Camazine, *Self-Organization in Biological Systems*, Princeton Studies in Complexity, Princeton University Press, 2001.
- [25] S. Goel and T. Imielinski, "Prediction-based monitoring in sensor networks," Tech. Rep. DCS-TR-438, 2001.
- [26] Manohar, *Configuring mobile sensors for maximizing even coverage [M.S. thesis]*, IIT Delhi, India, 2008.
- [27] D. E. Y. Wei and J. Heidmann, "An energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 21st International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '02)*, pp. 17–22, 2002.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

