Selection and Navigation of Mobile Sensor Nodes Using a Sensor Network

Atul Verma, Hemjit Sawant and Jindong Tan
Department of Electrical and Computer Engineering
Michigan Technological University
Houghton, MI 49931, USA

Abstract

Hybrid sensor networks comprise of mobile and static sensor nodes setup for the purpose of collaboratively performing tasks like sensing a phenomenon or monitoring a region. In this paper, we present a novel approach for navigating a mobile sensor node (MSN) through such a hybrid sensor network. The static sensor nodes in the sensor network guide the MSN to the phenomenon. One or more MSN's are selected based on their proximity to the detected phenomenon. Navigation is accomplished using the concepts of credit based field setup and navigation force from static sensor nodes. Our approach does not require any prior maps of the environment thus, cutting down the cost of the overall system. The simulation results have verified the effectiveness of the proposed approach. In each of the simulation runs, the static sensor nodes were able to successfully guide the MSN towards the phenomenon

1 INTRODUCTION

A networked system of mobile robots and static sensors opens new frontiers in a variety of civilian and military applications and in some scientific disciplines. A mixture of networked mobile robots and static sensors reduces the cost but preserves the flexibility and advantageous capacities of a multi-robot system. The interaction of multi-robot systems and static sensor networks enhances each other's capability. The mobile sensors can reallocate sensing, networking and computing resources to provide required coverage and specified sensing accuracy, to collect data from close by static sensors with higher energy efficiency, to task the static sensors, to repair and maintain the network, while the wireless network and static sensors provide environmental sensing, communication, coordination and navigation to the multi-robot systems. Mobility allows the static sensor network to collect data from remote and/or hazardous locations, and to adapt itself to unpredictable and variable environments. Hybrid sensor networks find myriad military applications including battle-field surveillance, reconnaissance and enemy tracking and civil applications consisting activities such as habitat monitoring [2, 19], environment observation [10], health [23] and other commercial applications [16].

In a hybrid sensor network, the mobile sensors are generally equipped with more resources such as sensors, power, computation, etc. Upon detection of an event, static sensors around the event may request the mobile sensors navigate themselves to the area of interests. Mobile sensors enhance sensing, communication and computation capabilities in the area of interests. Additionally, sensor nodes are low power devices and rely on in-network processing for carrying out tasks such as target tracking and collaborative sensing. Sensors sense the phenomena collectively and deliver sensing data via a wireless channel. As computation is much cheaper in terms of energy expenditure than communications, so instead of sending the raw data to the nodes, sensor nodes use their processing capabilities to locally carry out simple computations and transmit only the required and partially processed data [24]. At the same time sensor network are subject to a certain set of resource constraints such as limited on-board battery power and network communication bandwidth.

Navigation of mobile sensor nodes (MSNs) is a challenging task. In this paper, we have used sensor networks for guiding MSN towards the goal position. Consider the scenario shown in Figure 1. It shows a MSN which needs to be guided towards the goal. We can assume that the MSN has GPS capabilities but it does not have to know the coordinates of the goal position. It can also be assumed that the sensor network can calculate a path from its current position to the goal position designated by the sensor nodes. The path designated by the sensor nodes is able to reflect the presence of static or moving obstacles. For example, we can see that there is a pond in this path. Even if the MSN is equipped with obstacle detecting sensors like sonar [20] or millimeter wave radar, yet it shall not be able to differentiate between the surface of land and the surface of water. Hence, it may go directly into the pond if it follows the straight line path. *Hybrid sensor networks* can be used as follows to solve this navigation problem.

Proceedings of the 3rd IEEE Int'l Conf. on Pervasive Computing and Communications (PerCom 2005)
0-7695-2299-8/05 $20.00 © 2005 IEEE

In Figure 1, static sensors have been deployed around the pond. These sensors have sensing, computation and communication capabilities. When an MSN comes in the communication range of any of these sensors wireless communication takes place between the MSN and the surrounding static sensors. The static sensor nodes can then navigate the MSN around the pond towards the goal. The MSN may choose any trajectory depending upon the direction it approaches the sensor network. Thus, we can see that by collaboration among static and mobile sensor nodes a challenging problem like the one discussed above can be solved. This problem will become more complex if there are many MSNs in the hybrid sensor network and events occur concurrently at different locations of the network. As is discussed ahead in this paper, MSN selection becomes a crucial step in the functioning of the overall sensor navigation system.

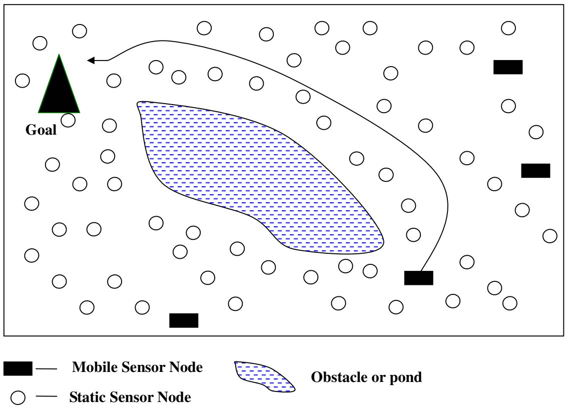


Figure 1. Navigation of a mobile sensor node using a static sensor network

The organization of the remaining part of this paper is as follows. Related work is discussed in Section 2. Section 3 presents the problem formulation and the model of static and mobile sensors. In Section 4, we propose a credit based navigation approach. The metrics for selecting an MSN from among several MSNs deployed in the sensor network are given in Section 5. Performance evaluation of our protocol is presented in Section 6. Conclusion and future work is discussed in Section 7.

2 Related Work

Navigation is the art of getting from one place to another in an efficient manner. It is one of the fundamental problems in Robotics. The general problem of mobile robot navigation can be described by three questions: “Where am I?”, “Where am I going?” and “How should I get there?” [3] In other words, navigation deals with guiding the robot from its present location to the desired location or *goal* by avoiding obstacles. In order to achieve this task, the robot makes runtime decisions based on data acquired by its sensors. The concept of artificial potential fields for the purpose of obstacle avoidance was presented in [14]. Some innate drawbacks of this approach such as, trap situations due to local minima or oscillations in narrow passages were discussed in [15]. The concept of Vector Field Histograms (VFH) based on locally constructed polar histograms for robot navigation was presented in [4], [25].

In [17], a neural map based approach for mobile robot navigation is developed. In [5], a landmark based approach for robot navigation is presented. In this approach, the environment contains easily detectable, unique landmarks that guide the robot motion. These approaches have inherent advantages and disadvantages. These approaches either assume that a map of the surroundings is provided beforehand or the sensors attached to the robot are very expensive. In our approach, static sensor nodes and mobile sensor nodes collaborate among themselves to guide the mobile sensor node (MSN) to its destination. This cooperation among the MSN and static sensor nodes can significantly reduce the cost of the robot. Batalin et al described a similar approach based on sensor networks in [1]. They utilized the transition probabilities called utilities, which are calculated at each sensor node for navigating a mobile robot. These utilities help in computing the optimal direction for guiding the mobile robot towards its goal. To reliably predict the node neighborhood the robot is in, they developed a scheme called Adaptive Delta Percent. In [18], a distributed algorithm is described for guiding mobile sensors across a sensor network along the safest path, away from danger that has been detected by the sensors. A artificial potential field is setup to guide the robot along the safest path. In our approach, a navigation path from MSN to the goal location is setup by using a credit field to increase the reliability of the path. The credit field is also used to develop virtual navigational forces from surrounding static sensor nodes to guide the robot to its destination. The closest approach for the selection of a mobile senor node is the bidding protocol for deploying mobile sensors [26]. In this approach, each mobile sensor has some price and static sensors bid for the mobile sensors. But this approach ignores two important parameters, power and distance of mobile node to the event for choosing a particular mobile node.

Information sharing and communication are inherent features of any navigation protocol in collaborative sensor networks. Some of the significant related research in this field are as follows. Directed diffusion is a data centric protocol for sensor network application [13]. In directed diffusion, nodes request data by transmitting interests for named data. LEACH is based on cluster formation in sensor networks where sensor nodes are randomly chosen as cluster heads in order to minimize energy dissipation [8]. We have used a credit-based approach in which nodes are assigned credit values according to their distance from the phenomenon. GEAR [28] uses energy aware neighbor selection to route packets towards the destination and restricted flooding to disseminate packets near the destination. GRAB [27] is based on building cost field and forwarding data along the interleaved mesh towards the sink. In our approach, the MSN calculates its navigation direction towards the phenomenon using the navigation force from the neighboring static sensor nodes.

3 Problem Formulation

A hybrid sensor network comprises of static sensor nodes and mobile sensor nodes which co-operatively perform tasks like collaborative sensing, environment monitoring, etc. Figure 3 depicts a hybrid sensor network. The *circles* represent *static sensor nodes*. These sensor nodes can be equipped with different types of sensors, such as, fire detecting sensors, temperature sensors, motion detectors, seismic sensors, etc. The *solid rectangles* depict MSNs which are sensor nodes capable of locomotion. These MSNs have more energy and longer communication range than static sensor nodes. The MSN can be equipped with enhanced sensing using visual sensors, obstacle detection sensors, like sonar or GPS receivers. The location of the static sensor nodes can be localized using the mobile sensor nodes [12, 21]. However, Our approach does not require any pre-defined map of the environment. Hence, the overall cost of the system is low. Consider that a hybrid sensor network, containing static and mobile sensor nodes has been setup for the task of detecting a fire. Using our scheme, the static sensor nodes should be able to guide a MSN towards the region of the event for the purpose of gaining enhanced coverage of the phenomenon. For example, in the event of a fire, some sensor nodes will detect the phenomenon. We call this region where the fire has occurred as the *region of phenomenon*. In Figure 3, the circle denotes the region of phenomenon. The task of the static sensor nodes is to communicate an alert signal through the sensor network such that the message reaches one or more MSNs. One or more of the MSNs will be selected and navigated towards it according to the task requirements. This process of selection and navigation of the MSN is explained in sections 4 and 5, respectively.

Modeling of Sensor Network

Consider that a sensor network consists of n static sensor nodes and m mobile sensor nodes, like the one shown in Figure 3. Generally, the number of mobile sensor nodes is comparatively less than the number of static sensor nodes.

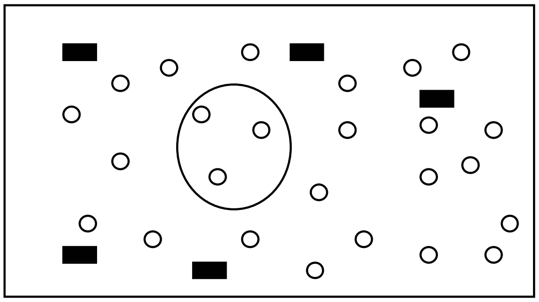


Figure 2. A hybrid Sensor Network where the circle represents a cluster of static sensor nodes that have detected the phenomenon.

Mobile Sensor Node Model

The Mobile Sensor Nodes, denoted by $M = \{M_1, M_2, M_3, ..., M_m\}$, are deployed in a network. The configuration of MSN, M_i, is denoted by $q_i(t) = [x_i, y_i, \theta_i]^T, i = 1, 2, 3, ..., m$, where x_i and y_i are the coordinates of MSN, M_i and θ_i is the orientation of the MSN with respect to its local coordinate system as shown in Figure 3. The dynamics of mobile robot M_i can be described by $\dot{q}_i = f_i(q_i, u_i)$, where u_i is the control input of MSN, M_i. This controller u_i is executed according to algorithm given in section 4 and provides the MSN with the direction towards which it should navigate.

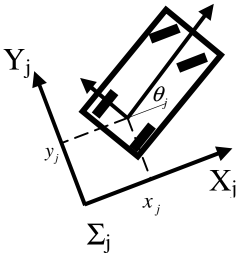


Figure 3. Model of a mobile sensor node

Static Sensor Node Model

The static sensor nodes which are deployed in large number are denoted by $S = S_1, S_2, S_3, ..., S_n$ and their configuration is denoted by p_i, as shown in equation 1.

$$p_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix}, i = 1, 2, 3, ..., n. \quad (1)$$

4 Credit Field Based Navigation

Once the static sensors have detected a phenomenon, the static sensors in the *region of phenomenon* form a dynamic cluster which includes the sensors which can detect the phenomenon. The cluster elects a leader [7, 22] and performs collaborative signal processing. According to the evaluation of the initial signal processing results, the cluster determines whether and how many mobile sensors are required to enhance the sensing quality in the region of phenomenon. If the mobile sensor is not in this region, then to perform enhanced sensing of the phenomenon, the static sensor nodes should navigate a mobile sensor in the network to the region of phenomenon. The first step is to select a mobile sensor among available MSNs and build up a navigation path between the mobile sensor and the *region of phenomenon*. The second step is to navigate the mobile sensor through the sensor network.

As a sensor network is a distributed system and there is no prior map of the environment available, the leader node does not know the location of the MSNs in the network. So a *weight request packet (WREQ)* is broadcast into the network by the leader node. This request propagates through the network in an ad-hoc manner till it reaches the various MSNs deployed in the network. The MSNs which are busy with some events or are low in power reject the *WREQ*. Other MSNs reply back to the initiating leader node by reversing the route through which the *WREQ* propagated. We assume that the request and reply packets follow the shortest path between the leader and the corresponding MSNs. This reply packet includes the weight of each MSN which is calculated by the metrics given in section 5. Lower is the weight of the MSN, greater is its probability of navigation to the region of phenomenon. When the leader receives replies from the available MSNs, it selects the MSN with the least weight. More than one MSN can be selected depending upon the requirements of the cluster.

4.1 Building up the Navigation Field

After the leader has selected a particular MSN, a navigation field from the phenomenon to that MSN is built. Initially, the leader is set at the highest credit value. This node then broadcasts an *advertisement* packet (ADV) with the node's credit value, C_1. The static sensor nodes which receive this ADV packet set their credit value at C_2 such that $C_2 < C_1$. This set of nodes which received the ADV packet also includes the node through which WREQ packet propagated and was in the route reply from MSN to the leader. Now, only this node, with its credit value as C_2, broadcasts the packet. All those nodes which receive this advertising packet (ADV) set their credit value at C_3 such that $C_3 < C_2 < C_1$. This packet may be received by some nodes which have credit value is of C_1. These nodes ignore this ADV packet. Similarly, this set of nodes which received the ADV packet include the node which was in the route reply from MSN to the leader. Now, only this node, with its credit value as C_2, broadcasts the packet. Again, some nodes at credit C_2 may receive this packet. These nodes discard this packet. This process continues till the ADV packet reaches the MSN. In this way a *credit hierarchy* or a *credit field* is created which determines the extent of the mesh.

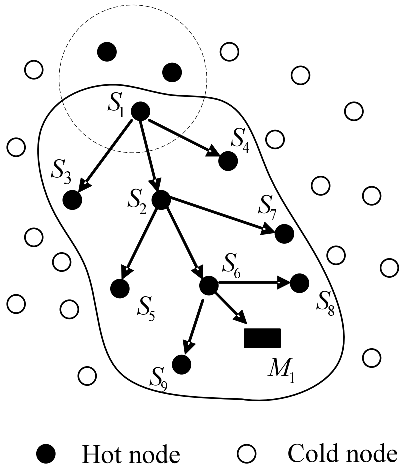


Figure 4. Navigation field denoting the mesh comprising of hot nodes to navigate MSN towards the phenomenon.

As shown in Figure 4, S_1, S_2, S_6, M_1 form the path taken by WREQ packet to reach MSN, M_1, from the leader node S_1 and for Route Reply to go from MSN to the leader node. S_1 sets itself at credit value C_1 and broadcast ADV packet containing this value. S_4, S_3, S_2 receive this packet and and set their credit values to C_2, such that $C_2 < C_1$. Now only node S_2 broadcasts the ADV packet containing its credit value. S_5, S_6, S_7, S_1, S_3, S_4 receive this packet. As S_4, S_3, S_1 already have some credit value so they ignore this packet where as S_5, S_6, S_7 set themselves at C_3 such that $C_3 < C_2 < C_1$. Now, only S_6 broadcasts the ADV packet containing its credit value and hence MSN M_1 is set at the lowest credit value.

All those nodes which are at some credit value are called as *hot nodes* while other nodes are called as *cold nodes*. As a result of these hot nodes a *mesh* of sensor nodes is built in the network which is used to guide the MSN to its destination. This *mesh* is called as the *navigational field*. The rational behind building this navigational field is that our simulations show it is very difficult to guide the MSN along a particular path without using a prior map of the environment. The additional nodes in the network provide *redundancy* i.e even if the MSN is lost, these additional nodes can guide the MSN to its destination.

4.2 Navigation of Mobile Sensor Node

A mobile sensor nodes is initially set to zero or lowest credit value. As each MSN moves it broadcasts a packet called *Navigation Request(NAV)*. All the hot nodes which are within the communication range of MSN reply back. Cold nodes ignore this NAV packet.

The message from hot nodes includes their credit field value, i.d and location information. MSN selects those hot nodes which have the maximum credit value and sets itself with the highest credit value received. It then calculates navigation controller u_i based on these selected sensor nodes. This controller u_i is executed for some time and provides the MSN with the direction towards which it should navigate.

For example, M_1 is the mobile sensor node in Figure 5 and S_1, S_2, S_3, S_4 and S_5 are the hot static sensor nodes. Credit value of node S_1 is same as that of node S_3. Similarly, S_2 and S_4 have the same credit vaue. The credit value of S_1 and S_3 is greater than that of S_2 and S4 which in turn is greater than that of S_5. MSN M_1 broadcasts the NAV and all the hot nodes S_1, S_2, S_3, S_4, S_5 reply back. M_1 selects S_1 and S_2 as they have the highest credit values and executes controller u_i for navigation. The algorithm is detailed in algorithm 1.

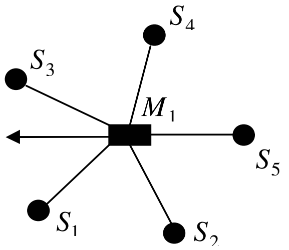


Figure 5. Calculating MSN's direction of motion

The navigational controller u_i is calculated on the basis of virtual attractive force generated by each static sensor which have the maximum credit field value during each phase of broadcast of navigation packet. The total energy associated with a mobile sensor node, (MSN_i), is defined as follows:

$$V_i = \frac{1}{2}\sum_{j=1}^{m_i} k_i(||p_{ij}||)^2 + \frac{1}{2}k_{iv}||v_i||^2 \qquad (2)$$

where $\mathbf{p}_{ij}$ is a vector from MSN, M_i to static sensor nodes S_n in the coordinate frame M_i and $||\mathbf{p}_{ij}|| = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$. m_i is total number of static sensor nodes which have the maximum value of credit during each phase of broadcast of navigation packet. k_i and k_{iv} are parameters of virtual potential energy and kinetic energy of the MSN, M_i. The navigational control input of the MSN, M_i is given by:

$$\begin{aligned} u_i &= \frac{\partial V_i}{\partial \mathbf{p}_i} - \frac{\partial V_i}{\partial \mathbf{v}_i} \\ &= -\sum_{n=1}^{m_i} k_i\, p_{ij} - k_{iv} v_i \qquad (3) \\ &= -F_i - k_{iv}\mathbf{v}_i, \end{aligned}$$

where $F_i = \{F_{ix}, F_{iy}\}$ is the virtual navigation force generated by static sensor nodes which are at the maximum credit value during each phase of broadcast of navigation packet. For the simulations, we assumed the value of k_i, as 0.25. In the above equation, The potential force is the vector summation of virtual forces from all *hot nodes* in the current communication range of the mobile sensor, as designated below:

$$F_i = \sum_{n=1}^{m_i} k_i\, \mathbf{p}_{ij}. \qquad (4)$$

The navigation process is terminated while the mobile sensor nodes reaches the *region of phenomenon*, which has the highest current credits. It is worth noting that the algorithm is suitable for dynamic events also. With a dynamic event, the *region of phenomenon* and the cluster changes over time. The credit field will be modified according to the change of the cluster. A higher credit will be assigned to the new cluster head. Another important issue is the local adjustment of the mobile sensor location in the *region of phenomenon*. As soon as the mobile sensors get into the region, local mobile sensor placement algorithms, can be used to adjust the position of the mobile sensors. Upon local adjustment, the sensing, computation and communication capability will be enhanced in the *region of phenomenon*. The modification is incremental to the established credit field. If the *region of phenomenon* has been significantly changed, a new credit field need to be negotiated to find a better path.

Algorithm 1: Mobile Sensor Navigation

```
1. Broadcast $NAV$
2. Collect values form hot nodes which include their credit field value, i.d. and location information
3. Select sensor nodes which have the maximum credit field value and set its (i.e. the MSN) credit as this value
4. Execute Navigational controller $u_i$ for certain period of time $t_i$
5. Return to step 1 if credit value of MSN $\neq$ credit value of initiating node; else STOP (i.e MSN has reached the phenomenon).
```

5 Selection of a Mobile Sensor Node

In order to select a particular MSN among various MSNs deployed in the network, three metrics, viz. coverage, power of each MSN and distance of each MSN form the phenomenon. Using these metrics the weight of each MSN is calculated. Lower is the weight more is the probability of the MSN reaching the phenomenon. This metric computation is performed only by the MSN and not by the static sensor nodes, hence, saving their energy. Firstly, each sensor node, whether static or mobile, is given a task of detecting a phenomenon in a certain region. Each sensor provides coverage in some specified area. When a mobile sensor is asked to move to a particular region where an event has occurred, its task of providing coverage at its place has to be taken by neighboring sensors; otherwise, it may leave coverage holes in the network. Our first metric is based on the *degree of coverage* provided by each MSN. Secondly, as each MSN is battery operated, so our selection criteria have to be energy conserving so that life of each MSN is maximized. Finally, the third metric considered is the distance of the phenomenon from the MSN. We now discuss the three selection metrics in detail.

Coverage of the Mobile Sensor Node in the Network

As each MSN is given a task of coverage or surveillance of a particular region, so moving a particular MSN should not leave a coverage hole its location. We use Voronoi cell approach to determine the coverage area provided by a particular mobile sensor node (MSN) [6, 11].

The Voronoi Cell has a property that it divides a plane with N points into N convex polygons such that each polygon contains exactly one point and every point in a given polygon is closer to its central point than to any other. These polygons formed are referred to as Voronoi cells.But, since a sensor network is a totally distributed system and no single sensor node knows the whole topological structure of the network, so it is difficult for a particular node to calculate Voronoi area with respect to the whole network.

Algorithm 2: Calculating Voronoi Area

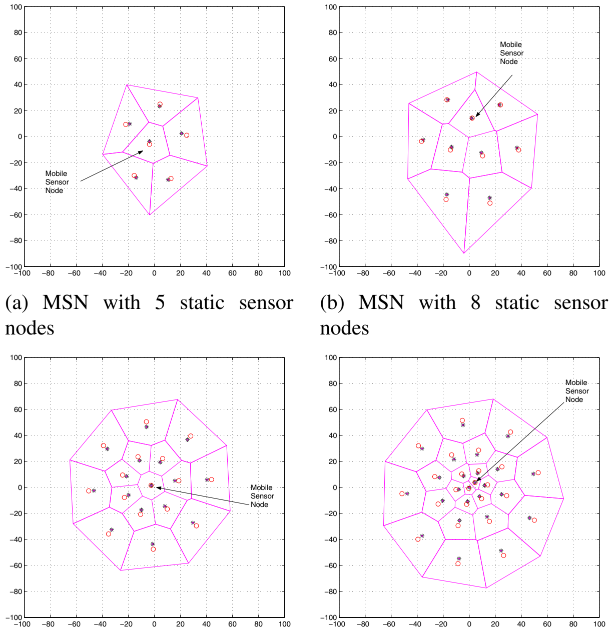
```
1. Let M be Mobile Sensor Node (MSN)

2. M broadcasts Hello message containing hop length
   equal to $n$ (i.e. the packet will travel $n$ hops)

3. $\forall$ ($Sensor\ nodes(S_i)$) that receive Hello message.
   Reply with their location information $(x, y)$ to M

4. At M, calculate Voronoi_Area
```

We propose a simple distributed algorithm for calculating the Voronoi area of a Mobile Sensor Node locally. This computation is performed at the MSNs hence it saves the energy of the static sensor nodes. According to this algorithm, each MSN broadcasts a *Hello* message requesting the neighboring nodes to send their location information. When it receives the location information from the neighboring nodes it calculates the Voronoi area with respect to these neighboring nodes. Greater is the Voronoi area, greater is the coverage provided by that MSN and more will be its weight. The algorithm is detailed in Algorithm 2.



(a) MSN with 5 static sensor nodes

(b) MSN with 8 static sensor nodes

(c) MSN with 15 static sensor nodes

(d) MSN with 24 static sensor nodes

Figure 6. Voronoi area for a MSN with different number of neighboring static sensor nodes

For our simulation we took $n = 2$, where n is the total number of hops a packet travels. Keeping $n = 2$ not only limits the number of *Hello* messages to be broadcasted by each MSN into the network saving lot of power but also calculates the voronoi area of each MSN optimally. Four instances are shown in Figures 5(a), (b), (c) and (d) where an MSN has 5, 8, 15 and 24 neighbors respectively. It is clearly demonstrated that greater the number of neighboring nodes, smaller is the Voronoi area of the MSN and less will be its weight.

Power of the Mobile Sensor Node

It is an important factor in choosing a particular sensor to do a particular task. The available power in each MSN is limited by battery lifetime which is difficult to replace. As a result, each MSN has to be used judiciously in order to elongate its life. Most of the energy is consumed during locomotion of the node. Greater the power of the mobile node greater is the distance it can traverse in the Network.

Distance

The third and final metric we have chosen to calculate the weight of the MSN is the distance of MSN from the phenomenon. As no single sensor node knows the whole topological structure of the network, so it is difficult to calculate the shortest distance between MSN and the phenomenon. Hence, the number of *hops* i.e. the total number of intermediate nodes through which the *weight request WREQ* packet travels from the phenomenon to the MSN is used to calculate the distance of each MSN from the phenomenon. More the number of hops more is the weight of the MSN.

Weight

Using the above three metrics the weight of each MSN is calculated as

$$Weight = \frac{Voronoi\_Area \times Distance}{Power} \qquad (5)$$

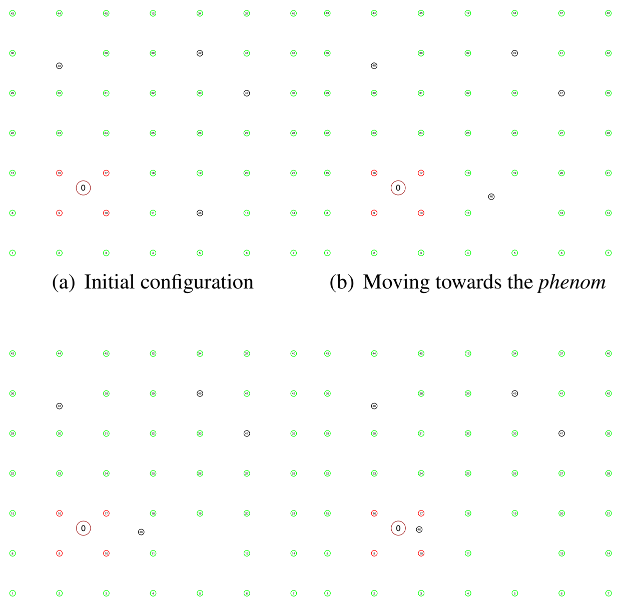
Lower is the weight more is the probability of the MSN reaching the phenomenon and vice versa. This computation is performed locally at each MSN. In case of two or more MSN's having equal weight the one which is closer to the phenomena in is selected for navigation.

6 Simulation Results

For simulating the sensor networks, we used the NRL's Sensor Network Extension to ns-2 [9]. The phenomenon was simulated using a *phenom* node which transmitted *phenom* packets. The static sensor nodes were modeled so as to detect these *phenom* packets. The mobile sensor nodes were similar to the static sensor nodes, except that they could move and had a larger communication range. The simulated environment was 500 meters by 500 meters. The proposed approach were executed for three different scenarios; a uniformly distributed sensor network, a randomly distributed sensor network and a sensor network with a "coverage hole" in it which could represent an "obstacle" or an impassable natural terrain, like the "pond" as depicted in Figure 1. All the MSNs were at the same power level at the start of the simulations. The simulation was assumed to be successful when the MSN was able to receive a "*phenom*" packet from the *phenom* node in a single hop i.e. the MSN was now in a position so to *sense* the *phenom*.

Uniformly distributed sensor network

Figure 7 shows the snapshots of executing our algorithm for a sensor network with uniformly distributed sensor nodes. This network consisted of four MSNs, fortyfive static sensor nodes and one *phenom* node.The MSN numbered 46 was selected and navigated towards the *phenom* node. As the network is uniformly distributed, Voronoi area calculated by each MSN is same and the power level of each MSN was same at the start of the simulation. So the MSN closest to the phenomenon, i.e. MSN number 46 is selected for the navigation.



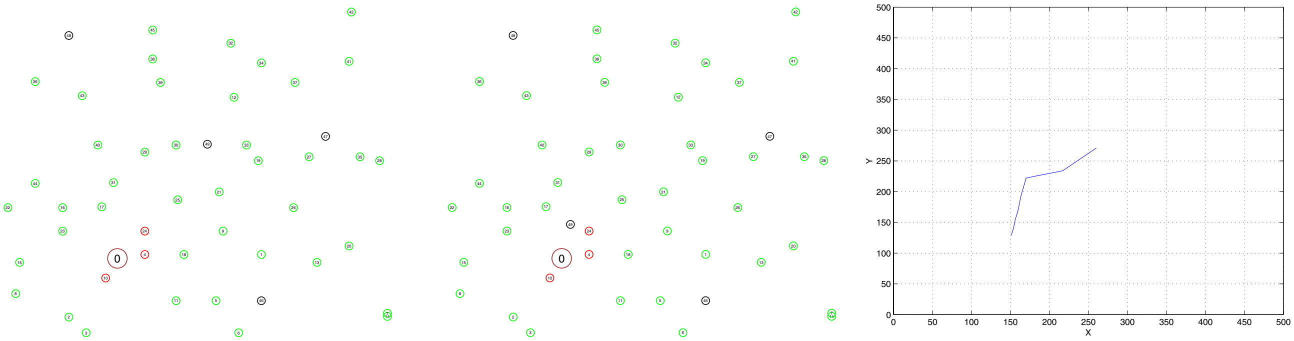
(a) Initial configuration (b) Moving towards the *phenom*

(c) Moving towards the *phenom* (d) Final configuration

Figure 7. Snapshots of executing the MSN navigation algorithm for a uniformly distributed sensor network

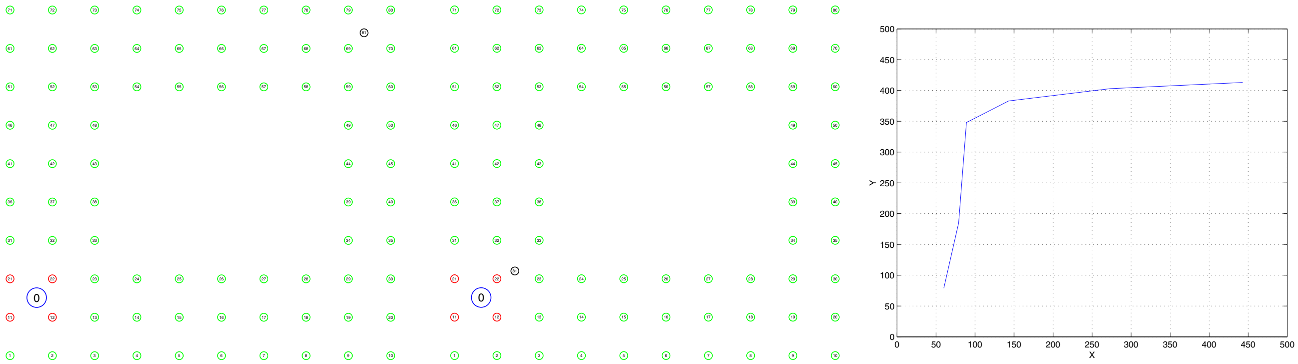
Randomly distributed sensor network

Figure 8 show the snapshots of executing our algorithm for a sensor network with randomly distributed sensor nodes. The number of static and mobile sensor nodes was the same as that for the uniformly distributed sensor network. There was one *phenom* node. MSN numbered 49 was selected for navigation. Its selection was based on its distance to the *phenom* node and its coverage area which is calculated on basis of number of hops the WREQ packet travels and the Voronoi_area of the respective MSN. MSN numbered 46 is closer to the *phenom* node than MSN 49. However, if MSN 46 is selected it will leave a large coverage hole at its position. So instead of MSN 46 MSN 49 is selected. As we see that the path traced by MSN is not necessary the shortest. The reason is that as MSN follows the path set by static sensor nodes and the shortest path may not always have enough sensor nodes. So the chances of MSN getting lost are more.



(a) Initial configuration (b) Final configuration (c) Trajectory of the MSN

Figure 8. Snapshots and MSN trajectory of executing the MSN navigation algorithm for a randomly distributed sensor network



(a) Initial configuration (b) Final configuration (c) Trajectory of the NSN

Figure 9. Snapshots and MSN trajectory of executing the MSN navigation algorithm for a sensor network with a "coverage hole"

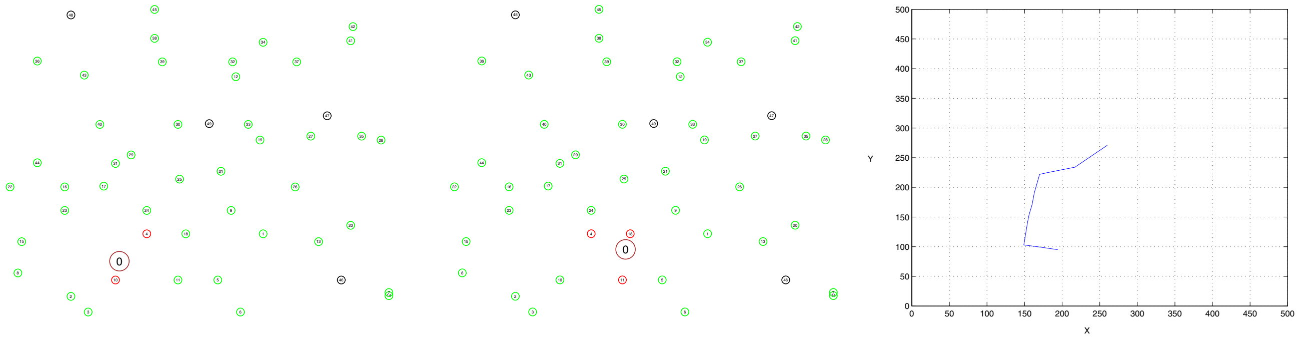
Sensor network with a coverage hole

Figure 9 show the snapshots of executing our algorithm for a sensor network with uniformly distributed sensor nodes and a coverage hole in it. This coverage hole could represent a "pond", as the one depicted in Figure 1. The simulated sensor network had 80 static sensor nodes, one MSN and one *phenom* node. At the start of the simulation, the MSN is placed at one corner of the sensor network. The *phenom* node generates a WREQ packet and the MSNs reply back with their weight. Then the path from the *phenom* node to the MSN is built and hence creating the mesh of hot nodes as explained in section 4. Figure 9(c) provides the path traced by the MSN to reach the *phenom* node. As can be seen from the snapshots and the traced path of the simulation, the path of the MSN skirted the "pond" and hence avoiding it.

Navigation with dynamic events

In the earlier part, we have assumed that the phenomenon is immobile. In this section we discuss how the credit field is built for moving phenomenon to keep track of the phenomena. The static sensors which have detected the phenomenon form a dynamic cluster. This cluster elects a leader where all the nodes in the cluster are at the same credit value. Now, suppose at this instant the phenomenon moves. As result of which, a new dynamic cluster is formed due to the inclusion of new nodes which have detected the phenomenon. This newly formed dynamic cluster elects a leader which broadcasts an ADV packet. The nodes which are already at this credit value reject this packet while all the other nodes are set at the higher credit value than the initiating node which has broadcasted the ADV packet.

As shown in Figure 10 static sensor nodes S_1, S_2, S_3, S_4 and S_5 are the nodes which form the initial cluster and all are at same credit value (say C_{11}). At this instant, the phenomenon moves and nodes S_6, S_7 and S_8 detect the detect the phenomenon and join the node S_5 to form a new dynamic cluster. S_5 is elected its leader which broadcast the ADV packet containing its credit value C_{11}. All the nodes



(a) Initial configuration (b) Final configuration (c) MSN Trajectory

Figure 11. Snapshots and MSN trajectory of executing the MSN navigation algorithm for a sensor network with a moving phenomenon

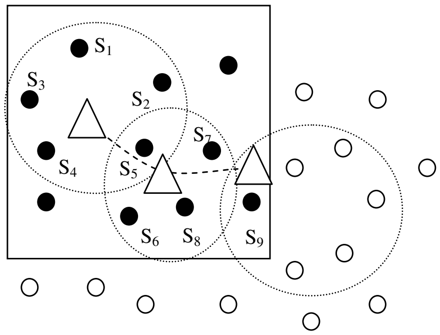


Figure 10. Dynamic cluster formation in sensor network due to moving phenomenon

shown in the rectangle will receive this packet. The nodes which are already at same credit value as C_{11} will ignore this packet while all other nodes become hot nodes and set their credit value as C_{12} where $C_{12} > C_{11}$.As the phenomenon moves the new nodes detect the phenomenon along with S_9 which is now chosen as the next leader and will broadcast the ADV packet with its credit value. In this way credit field is built from low to high for guiding the MSN.

Figure 11 shows the snapshots of implementing our approach for a moving phenomenon. It shows the initial positions of the MSN and the *phenom* nodes. We can see that the *phenom* node has moved to a new location. The trajectory of the MSN through the sensor network is also shown. To guide the MSN and to build the navigation path subsequent leader nodes were chosen. The node that detected the *phenom* first after it moved to its new position was chosen the leader. This node broadcasted an ADV packet on sensing the phenomenon. The other nodes around this node which received this packet formed a dynamic cluster with this node as the leader.

7 Conclusion

This paper presented a credit field based approach for mobile sensor navigation in a hybrid network of mobile and static sensors. The credit field provides navigation path between the mobile sensor and *region of phenomenon*. Designated by the static sensors, the credit field is stored locally in the static sensors. A distributed navigation algorithm is also presented which follow the credit field and route the mobile sensor to the *region of phenomenon*. The algorithm is robust to failures of static sensor. The algorithms have been verified in an *ns-2* environment. Future work includes implementation of the presented approach for sensor networks with moving phenomenon.

References

[1] M. A. Batalin, G. S. Sukhatme, and M. Hattig. Mobile robot navigation using a sensor network. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 636–642, New Orleans, USA, April 2003.

[2] E. Biagioni and K. Bridges. The application of remote sensor technology to assist the recovery of rare and endangered species. *International Journal of High Performance Computing Applications*, 16(3):315–324, Aug 2002.

[3] J. Borenstein and H. R. Everett. *Navigating Mobile Robots: Sensors and Techniques*. John Wiley & Sons, 1992.

[4] J. Borenstein and Y. Koren. The vector field histogram – fast obstacle-avoidance for mobile robots. *IEEE Journal of Robotics and Automation*, 7(3):278–288, 1991.

[5] A. J. Briggs, C. Detweiler, D. Scharstein, and A. Vandenberg-Rodes. Expected shortest paths for landmark-based robot navigation. *International Journal of Robotics Research*, 23:717–728, July 2004.

[6] J. Cortes, S. Martinez, T. Karatas, and F. Bullo. Coverage control for mobile sensing networks. In *IEEE Transactions*

on Robotics and Automation, pages 243 – 255, New Orleans, USA, April 2004.

[7] S. Dulman, P. Havinga, and J. Hurink. Wave leader election for wireless sensor networks. In *Proceedings of the 3rd International Symposium on Mobile Multimedia Systems & Applications*, pages 43–50, Delft, Netherlands, Dec 2002.

[8] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the 33rd International Conference on System Sciences*, January 2000.

[9] http://nrlsensorsim.pf.itd.nrl.navy.mil/.

[10] http://www.alertsystems.org.

[11] http://www.voronoi.com.

[12] L. Hu and D. Evans. Localization for mobile sensor networks. In *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*, pages 45–57. ACM Press, 2004.

[13] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion: a scalable and robust communication paradigm for sensor networks. In *Proceedings of the sixth annual international conference on Mobile computing and networking*, pages 56–67. ACM Press, 2000.

[14] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research*, 5(1):90–98, 1986.

[15] Y. Koren and J. Borenstein. Potential field methods and their inherent limitations for mobile robot navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1398–1404, Sacramento, California, April 1991.

[16] V. A. Kottapalli, A. S. Kiremidjian, J. P. Lynch, E. Carryer, T. W. Kenny, K. H. Law, and Y. Lei. Two-tiered wireless sensor network architecture for structural health monitoring. In *Proceedings of the 10th Annual International Symposium on Smart Structures and Materials*, pages 8–19, San Diego, USA, March 2003.

[17] M. G. Lagoudakis and A. S. Maida. Neural maps for mobile robot navigation. In *Proceedings of the IEEE International Joint Conference on Neural Networks*, Washington, D.C., USA, July 1999.

[18] Q. Li, M. DeRosa, and D. Rus. Distributed algorithms for guiding navigation across a sensor network. In *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking*, pages 313 – 325, San Diego, USA, Sep 2003.

[19] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson. Wireless sensor networks for habitat monitoring. In *The ACM International Workshop on Wireless Sensor Networks and Applications*, Atlanta, USA, Sep 2002.

[20] J. Manyika and H. Durrant-Whyte. A tracking sonar sensor for vehicle guidance. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 3, pages 424 – 429, Atlanta, USA, May 1993.

[21] P. Pathirana, N. Bulusu, S. Jha, and A. Savkin. Node localization using mobile robots in delay-tolerant sensor networks. *IEEE Transactions on Mobile Computing*, 4(4):90–98, 2005.

[22] K. Premaratne, J. Zhang, and M. Dogruel. Location information-aided task-oriented self-organization of ad-hoc sensor systems. *IEEE Sensors Journal*, 4(1):85–95, Feb 2004.

[23] L. Schwiebert, S. K. S. Gupta, and J. Weinmann. Research challenges in wireless networks of biomedical sensors. In *Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 151–165, Rome, Italy, July 2001.

[24] R. Shah, S. Roy, S. Jain, and W. Brunette. Data MULEs: modeling a three-tier architecture for sparse sensor networks. In *Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications,*, pages 30–41, May 2003.

[25] I. Ulrich and J. Borenstein. Vfh+:reliable obstacle avoidance for fast mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1572–1577, Leuven, Belgium, May 1998.

[26] G. Wang, G. Cao, and T. LaPorta. A bidding protocol for deploying mobile sensors. In *Proceedings of the 11th IEEE International Conference on Network Protocols*, pages 315–324, Nov 2003.

[27] F. Ye, G. Zhong, S. Lu, and L. Zhang. Gradient broadcast: A robust data delivery protocol for large scale sensor networks. *ACM Wireless Networks (WINET)*, 11(2), 2005.

[28] Y. Yu, R. Govindan, and D. Estrin. Geographical and energy aware routing: A recursive data dissemination protocol for wireless sensor networks. *UCLA Computer Science Department Technical Report UCLA/CSD-TR-01-0023*, 2001.