Ensuring Area Coverage in Hybrid Wireless Sensor Networks

Nadeem Ahmed[1,2], Salil S. Kanhere[1], and Sanjay Jha[1]

[1] Computer Science and Engineering, University of New South Wales, Australia.
{nahmed, salilk, sjha}@cse.unsw.edu.au
[2] National ICT Australia (NICTA), Australian Technology Park, Sydney, Australia.

Abstract. Success of Wireless Sensor Networks largely depends whether the deployed network can provide desired coverage with acceptable network lifetime. This paper proposes a distributed protocol for ensuring area coverage using a combination of mobile and static sensor nodes. Most of the assumptions made in our approach are realistic (sensing model, movement thresholds based on real radio characteristics etc.) and implementable in real-life applications. We demonstrate that, for different type of initial deployments, our proposed movement algorithms consume only 30-40% of the energy consumed by the basic virtual force algorithm. We formulated our problem as Integer Linear Program to arrive at idealistic optimal solutions that form basis of our performance comparison. We validated our results through extensive discrete event simulations.

Key words: Coverage, deployment, hybrid wireless sensor networks.

1 Introduction

In hostile or harsh environments such as enemy territories in battlefields, fire or chemical spills, it is impossible to deploy the sensor nodes in a predetermined regular topology. Random (possibly aerial) deployment of sensor nodes is a solution in such scenarios. However, such random deployments are highly susceptible to the creation of uncovered regions, referred to as *coverage holes*. There are many factors that contribute to this including the presence of obstacles, sloping grounds like hills, strong winds or dense forestation during aerial deployment, etc. A potential solution for enhancing the existing coverage involves the use of mobility capable sensors [1] [2], which would help fill in the voids. In this paper, we seek to address the problem of determining the current coverage achieved by the non-deterministic deployment of static sensor nodes and subsequently enhancing the coverage using mobile sensors. We propose a distributed *Mobility Assisted Probabilistic Coverage (MAPC)* protocol, which aims at maximizing the area coverage with least expenditure of energy.

Our primary contribution is a pragmatic approach to sensor coverage that we hope would lower the technical barriers to its field deployment. Most of the assumptions made in the proposed protocol are realistic and implementable in real-life applications, e.g., coverage calculations based on realistic sensing model

(Section 4), and use of thresholds based on real radio characteristics (Section 5) etc. Further, our movement algorithms result in substantial energy savings (60% to 70% as compared to basic virtual force based algorithms), a major challenge for the success of WSN.

The primary focus of this paper is the design and evaluation of the MAPC protocol. We discuss related research work in Section 2 and provide context to our work in Section 3. We present various phases of our protocol and simulation results in Sections 4 and 5. We present our observations and open problems in Section 6.

2 Related Work

Use of mobility capable sensors to guarantee coverage during deployment has been proposed in various research efforts [1], [3], [4], and [5] etc. A potential field based approach is proposed in [1] assuming compact initial concentration of the mobile nodes. Wang et al. proposed three different deployment protocols in [3], that spread out the mobile sensors once coverage holes are detected using Voronoi diagrams. A centralized incremental deployment scheme is proposed in [4] that deploys sensors one by one while requiring line of sight among the nodes. Similarly, a centralized Virtual Force Algorithm for the movement of mobile sensor nodes is proposed in [6]. All these works assume an all mobile sensor network. In [5] the coverage problem is solved by a moving robot that is guided by the already deployed nodes for exploring poorly covered areas. Use of mobile sensors for enhancing coverage at deployment stage for hybrid WSN is proposed in [2] and [7]. Wang et al. proposed a bidding protocol [2] where static nodes bid for mobile nodes while mobile nodes also consider the loss of coverage at its existing location due to the movement.

Our work is different from these proposed approaches in several ways. We propose a coverage and energy-aware, distributed variant of the VFA for spreading out the mobile sensors as opposed to the centralized VFA employed in [6]. Our proposal do not require provision of any specialized hardware for determining the range and bearing [1], or line of sight [4]. In addition, we also propose the simulated movement approach for energy efficient movement of mobile sensor nodes. Our approach is different from the proxy based logical movement approach proposed in [7] where static sensor nodes are utilized as proxies for message passing. We also propose movement controlling thresholds based on real radio characteristics for ensuring continued reliable communications between current neighbors.

3 Protocol Overview

Our system assumes a hybrid network consisting of a large number of static sensors deployed in a non-deterministic manner. We also assume that a few mobile sensors are available for plugging the coverage holes. The proposed MAPC protocol works in two distinct phases.

Phase I aims at estimating the existing coverage provided by the randomly deployed static nodes. The widely used binary detection model, assumes that the sensing coverage of a sensor node is uniform in all directions, often represented by a unit disc. However, in reality, the sensing capabilities of a sensor are often affected by environmental factors. In particular, for range-based sensor modalities such as acoustics, radio, etc, the signal strength of the triggering signal decays as a function of distance. This implies that the detection capabilities of these sensors would exhibit similar characteristics as opposed to a uniform sensing range. In an effort to employ more realistic models in the computation of area coverage, in our work, we use the *Probabilistic Coverage Algorithm (PCA)* [8], that takes into account the probabilistic sensing behavior of sensor nodes.

Phase II manages relocation of the mobile sensors. For this phase, we propose a set of coverage and energy aware variants of *Virtual Force Algorithm.* These algorithms work in rounds and manage the mobile node relocation that serves a dual purpose. Firstly, this relocation *increases* the coverage during deployment by allowing the mobile nodes to fill in the coverage holes with minimal expenditure of energy. Secondly, the additional mobile sensors are uniformly spread in the target area for further discovery of coverage holes. The movements of the mobile nodes are controlled by novel thresholds based on real radio characteristics. This ensures that the nodes can communicate with each other, with high probability of successful transmission, during the round-by-round movements. The mobile nodes relocation in phase II thus results in increase in area coverage during the deployment stage. The individual phases of the proposed protocol are elaborated in detail in the subsequent sections.

4 Phase I: Coverage Estimation

We make the following assumptions:

- Static nodes are deployed in a random, non-deterministic fashion in an unknown obstacle-free environment.
- Location information is available using any existing GPS-less sensor network localization scheme.
- Sensors cannot detect the physical boundary of the region in outdoor environments. This assumption is consistent with the real world capabilities of existing sensor hardware.
- Nodes lying on the outer boundary of the deployed network topology (referred to as B-nodes) are aware of their special topological position using any of the existing outer boundary nodes identification schemes such as [9], [10], etc.

4.1 The Probabilistic Coverage Algorithm (PCA)

For coverage calculation, we use a realistic probabilistic coverage model proposed in our earlier work [8]. Using the probabilistic coverage model, the change in detection probabilities with distances can be represented by concentric circles drawn at constant distance increments around the sensor location. Each circle thus represents the probability of correctly receiving a signal, with strength above the receiving threshold, at a distance equal to the radius of that particular circle (see Figure 1). Assuming that the transmit power, P_t and receive threshold, γ, is known for a sensor through a priori field experiments and sensor calibration, a *probability table*, PT can be precomputed that provides the discrete detection probability at various distances from the sensor. If ρ_{reqd} represents the desired detection probability required by the application using the sensor network, the corresponding distance value, d_{reqd}, can be found from the PT.

For a deployed sensor network, a point in the target region can be covered by more than a single sensor. The overall detection probability Pr of a point in the region is thus given by Equation 1.

$$Pr = 1 - \prod_{i=1}^{N} (1 - Pr_i) \tag{1}$$

where N represents the number of sensor node covering a particular point and Pr_i denotes the detection probability of a point for a sensor i. The detection probability at any location is thus increased by contributions from all the sensors covering that point and it is possible to achieve the desired detection probability at distances greater than d_{reqd} from the sensor. The basic idea is to take the next higher distance from the probability table PT as d_{eval} (with lower detection probability than ρ_{reqd}) and evaluate whether contributions from neighbors makes the perimeter at d_{eval} sufficiently covered or not.

A node Sj that is a neighbor of Si has several concentric circles representing regions of different detection probabilities (see Fig 1). Node S_i calculates the cumulative detection probability at intersection of its circle at d_{eval} with various circles of neighbor S_j e.g., Pjs in Figure 1. $C(r, p)$ represent the circle around S_j with radius r providing probability of detection p.

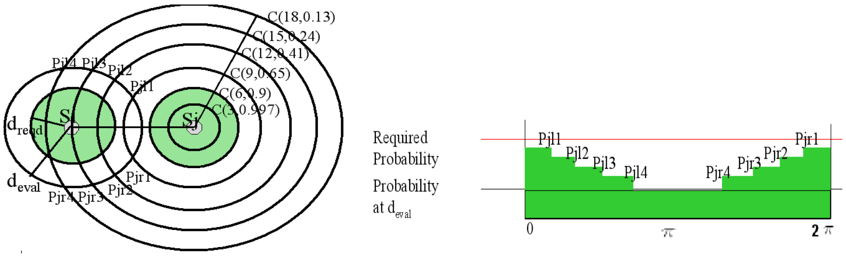

Fig. 1. Perimeter Coverage using PCA

The cumulative detection probability is then placed on a line segment $[0, 2\pi]$ representing the perimeter of S_i at d_{eval}. This is repeated for each neighbor until the whole perimeter is found covered by probability ρ greater than the required value. After executing the PCA, if a node finds that its evaluated perimeter is uncovered, the node can calculate a deployment location where a redundant helper node, S_h, can be placed such that the perimeter coverage constraint for the node is satisfied [8].

5 Phase II-Spreading out of Mobile Sensors

Random aerial deployment is extremely challenging for mobile nodes as it may result in physical damage to their locomotive parts. In realistic deployments, the mobile nodes are normally accumulated at one or more points near the target area. We therefore, consider two different initial deployment methodologies namely *Normal* and *Island* distribution. In normal distribution, mobile sensors form a single cluster at the boundary, while in island distribution they form disconnected clusters at different locations on the boundary. For spreading of the clustered mobile sensor nodes, we propose to use the concept of Virtual Forces from robotics [1]. In this context, we propose two variants of *Virtual Force Algorithm (VFA)* (i) Coverage and Energy Aware VFA (CEA-VFA) and (ii) CEA-VFA with Simulated Movements (SM). These two proposed algorithms are explained in detail in later sections.

We assume that location information is available using any existing GPS-less WSN localization scheme for the mobile nodes such as [11] and that mobile nodes have significantly more initial energy than the static nodes.

5.1 Coverage and Energy Aware Virtual Force Algorithm (CEA-VFA)

We first provide a short overview of the basic VFA proposed in [1], [6]. Basic VFA attempts to iteratively spread the mobile sensors in the target area by using a combination of attractive and repulsive forces. Two mobile sensors will exert virtual forces on each other if the Euclidean distance, $d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$, between them is not between a given range of thresholds, Th_{push} and Th_{pull} (discussed in detail later in this section). This virtual force, F_{ij} is a pull or attractive force if the distance between the two mobile nodes is greater than the pull threshold, Th_{pull}, while if the distance is less than the push threshold, Th_{push}, a push or repulsive force is exerted. Equation 2 shows the model used for decision making.

$$\boldsymbol{F}_{ij} = \begin{cases} F_{push}, & \text{if } d_{ij} < Th_{push} \\ 0, & \text{if } Th_{push} \le d_{ij} \le Th_{pull} \\ F_{pull}, & \text{if } d_{ij} > Th_{pull} \end{cases} \tag{2}$$

where $\boldsymbol{F}_{ij}$ is the force exerted on mobile node Si by neighbor Sj. Taking the midpoint of range between Th_{push} and Th_{pull} as the desired distance between the mobile nodes, Equations 3 and 4 represent the push/pull virtual forces.

$$F_{push} = (\frac{Th_{pull} + Th_{push}}{2}) - d_{ij} \tag{3}$$

$$F_{pull} = d_{ij} - (\frac{Th_{pull} + Th_{push}}{2}) \tag{4}$$

Representing magnitude of the force in terms of distance, the share of the virtual force for a node Si due to node Sj is represented by $\boldsymbol{dm}_{ij} = \frac{\boldsymbol{F}_{ij}}{2}$, the magnitude of which is denoted by $|dm_{ij}| = |(F_{ij}/2)|$. We can express the total force exerted on a mobile sensor S_i by its n mobile neighbors, denoted by $\boldsymbol{F}_i$, as,

$$\boldsymbol{F}_i = \sum_{j=1, j \neq i}^{n} \frac{\boldsymbol{F}_{ij}}{2} = \sum_{j=1, j \neq i}^{n} \boldsymbol{d}m_{ij} \tag{5}$$

Note that $\boldsymbol{F}_i$ is the vector sum of all the forces acting on mobile sensor node S_i, the magnitude and orientation of which can be easily calculated, e.g., Robomote [12] uses an on-board compass combined with localization information for navigation purposes. The VFA works in rounds and the mobile nodes are iteratively moved to attain a more uniform distribution in the region.

Having explained the concept of virtual forces, we now detail the changes that have been adopted in CEA-VFA in the following sections.

Coverage Awareness. In CEA-VFA, a mobile node first checks whether it is in the vicinity of a coverage hole (has a neighbor static node with uncovered perimeter) and if so, it reacts to plug in the discovered hole before participating in the virtual force calculations. This coverage check is performed by each mobile node in each round of the CEA-VFA. Each round starts with mobile and bounday nodes (B-nodes) exchanging the location information using *Hello* messages. A Hello message contains the sender node ID, current location coordinates, and the remaining energy. A mobile node on reception of a Hello message marks the sender as its current neighbor and starts a TM_{wait} timer.

Static nodes with uncovered perimeter also start a TS_{wait} timer on receiving a Hello message from a mobile node. This timer is reset each time a Hello message is received. On expiry of the TS_{wait} timer, a static node with uncovered perimeter selects the nearest mobile node from its mobile node neighbor list and sends a *Help* message to the selected mobile node. The Help message contains sender node id, location, requested deployment point, and the expected gain in coverage (difference between the required detection probability ρ_{reqd} and the existing probability at uncovered perimeter ρ_{exist}).

A mobile node may receive multiple Help messages from different static nodes with uncovered perimeters. The mobile node selects the requested deployment point that involves the highest gain in coverage, broadcasts a *Move* message, and starts moving toward the requested deployment point. As the Move message is a broadcast, it is received by both mobile and static sensor nodes. Neighboring mobile nodes that receive the Move message, remove the sender node from the list of neighboring mobile nodes. The mobile nodes after expiry of the TM_{wait} timer perform the virtual force calculations to spread out in the topology to discover more coverage holes.

Energy Awareness. To ensure that the mobile nodes expend energy uniformly while in motion during the deployment phase, we introduce an adaptive policy based on the residual energy of the nodes. The virtual forces are made proportional to the residual energy of the node, with a higher energy node absorbing a greater portion of the virtual force than its low energy neighbor. Let E_I represent the initial energy of a mobile node, E_{ci} and E_{cj} represent the current remaining energies of nodes S_i and S_j respectively. We have $e_{ij} = (E_{ci} - E_{cj})/E_I$, where e_{ij} is the proportional energy coefficient. Representing magnitude of the force in terms of distance, $\boldsymbol{dm}_{ij} = \frac{\boldsymbol{F}_{ij}}{2}(1 + e_{ij})$. The total force exerted on a mobile sensor S_i by its n mobile neighbors, denoted by $\boldsymbol{F}_i$, can be expressed as,

$$\boldsymbol{F}_i = \sum_{j=1, j\neq i}^{n} \frac{\boldsymbol{F}_{ij}}{2}(1 + e_{ij}) = \sum_{j=1, j\neq i}^{n} \boldsymbol{dm}_{ij} \tag{6}$$

Choice of Thresholds, $\boldsymbol{Th_{push}}$ and $\boldsymbol{Th_{pull}}$. We want to ensure that the mobile nodes are able to communicate with neighbors during the round by round operation of the algorithm. Rather than using static values of movement triggering thresholds, Th_{push} and Th_{pull}, these thresholds should depend on the link quality. If the communication link quality is good, sensor nodes can be placed further apart (higher value of Th_{pull} can be used). Although other complex models can also be used, we use a simple radio propagation model based on log-normal shadowing [13] to characterize the communication link between two sensors.

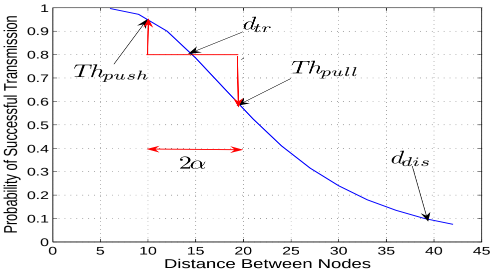

Fig. 2. Change in communication probability with distance

Each sensor has a *receive threshold* value γ that describes the minimum signal strength that can be correctly decoded at the sensor. The probability Pr that a received signal level, P_{rec} at a sensor will be above this receive threshold, γ, is given by Equation 8, with Q-function to compute probability involving the Gaussian process. The Q-function is defined as

$$Q(z) = \frac{1}{\sqrt{2\pi}} \int_{z}^{\infty} exp(-\frac{x^2}{2})dx \tag{7}$$

where $Q(z) = 1 - Q(-z)$.

$$Pr[P_{rec}(d) > \gamma] = Q[\frac{\gamma - P_{rec}(d)}{\sigma}] \tag{8}$$

For a given transmit power and receive threshold value, we can calculate the probability of receiving a signal above the receive threshold value, γ, at a given distance using Equations 7 and 8 as shown in Figure 2.

Following the terminology in [14], there are three distinct reception regions in a wireless link: connected, transitional, and disconnected. The transitional region has highly unreliable links and its region bounds can be found either by analytical or empirical methods [14]. Let d_{tr} and d_{dis} represent the points where the transitional and disconnected regions begin respectively. We define Th_{push} and Th_{pull} as $(1-\alpha)d_{tr}$ and $(1+\alpha)d_{tr}$ respectively, where α denotes the error tolerance coefficient. Note that the values of Th_{push} and Th_{pull} are bounded by d_{dis}. α reflects the tolerance to the errors in localization and odometry during navigation of the mobile nodes. As long as the final position after movement is within this range, the deviation from the ideal trajectory during movement can be tolerated by our movement algorithm. Figure 2 shows d_{tr}, Th_{push}, Th_{pull} and d_{dis} in terms of probability of correct packet reception.

Boundary Considerations. To ensure that mobile sensors remain within the virtual boundary, B-nodes exert virtual forces ($\boldsymbol{F}_{ib}$) on mobile sensor nodes. A repulsive virtual force $\boldsymbol{F}_{ib}$, based on the mobile node's distance from a B-node, is included in the resultant vector sum of all virtual forces. The new total force is then expressed as,

$$\boldsymbol{F}_i = \sum_{j=1, j\neq i}^{n} \frac{\boldsymbol{F}_{ij}}{2}(1+e_{ij}) + \sum_{b=1}^{k} \boldsymbol{F}_{ib} \tag{9}$$

where $\boldsymbol{F}_{ib}$ is the repulsive force exerted on the mobile node Si by its k neighbor B-nodes and is modelled by Equation 10.

$$\boldsymbol{F}_{ib} = \begin{cases} \boldsymbol{F}_b, \text{ if } d_{ib} < \frac{(Th_{push}+Th_{pull})}{2} \\ 0, \quad \text{otherwise} \end{cases} \tag{10}$$

The repulsive force $\boldsymbol{F}_b$ is given by Equation 11 where d_{ib} is the distance between the node and the B-node.

$$\boldsymbol{F}_b = \frac{(Th_{push}+Th_{pull})}{2} - d_{ib} \tag{11}$$

As B-nodes are all static nodes, mobiles nodes absorb all of the virtual force resulting from the B-nodes. As a final check, mobile nodes should not cross the virtual boundary formed by the known B-nodes.

5.2 Virtual Force Algorithm with Simulated Movement

Mobile nodes move after each round of CEA-VFA due to the virtual force exerted by their neighbors before settling down to their final position in the topology. If we could calculate the final position of a mobile node and move directly to that final position, we can save energy by moving much lesser distances, i.e., by using the cut-through paths instead of the zig-zag paths that result from the round-by-round operation of CEA-VFA.

We propose CEA-VFA with Simulated Movement (SM) approach that attempts to use the cut-through paths for movement. Nodes go through the CEA-VFA iterations and calculate new position after each round. The difference is that nodes do not physically move after each iteration, rather, they stay at their original position and simply assume the new calculated virtual position. The Hello messages at the start of the next round contain the new virtual positions enabling the recipients to use this updated position information for the upcoming round of CEA-VFA. Nodes only move once they have calculated their final positions.

This simplistic approach has a few disadvantages. B-nodes, coverage holes, and new mobile neighbors are discovered by the per round movements in CEA-VFA. If a fully simulated run of CEA-VFA is used, it is highly likely that the mobile nodes may not detect the presence of the other mobile nodes and B-nodes in the region. Similarly, it is not possible to locate all of the coverage holes. One way to offset this disadvantage is to use intermittent simulated movement instead of fully simulated movement. In the CEA-VFA Intermittent Simulated Movement (ISM) approach, nodes simulate the movement for x number of rounds. Actual physical movement only takes place after every x number of rounds e.g., in ISM2 ($x = 2$) nodes physically move after every second round. Similarly, in ISM6 a node physically moves after every sixth round of virtual force calculations.

The simulated movement approach does not incur any additional communication overhead than the CEA-VFA rather it reduces energy consumption due to movement by moving the mobile sensors in fewer number of rounds. This also results in quicker deployment due to reduction in the time spent in the per round zigzag movements of the CEA-VFA.

5.3 Performance Evaluation

We conducted NS2 simulations for basic VFA, CEA-VFA and different variants of ISM and compared the results with an centralized optimized assignment.

A Centralized Optimization. We want to optimally assign the available mobile sensors, firstly, to locations requested by static nodes and secondly, to locations in the topology so as to form a uniform distribution. This is a classical *Assignment* optimization problem, also referred to as *bipartite weighted matching* problem in graph theory. A centralized optimization can be performed, if we assume that the location information of all coverage holes and the existing positions of all mobile nodes are known.

The assignment problem can be formulated as following: Let S_m represent the set of p mobile nodes, $S_m = \{S_{m1}, S_{m2}, \ldots, S_{mp}\}$ and set S_h represent the location of q coverage holes in the topology, $S_h = \{S_{h1}, S_{h2}, \ldots, S_{hq}\}$. After assigning q out of p mobile nodes, we have r remaining mobile sensors, ($r = p - q$), that needs to be uniformly distributed in the coverage area. The set S_u represent the deployment locations of these r mobile nodes, $S_u = \{S_{u1}, S_{u2}, \ldots, S_{ur}\}$.

Let $S_d = \{S_{d1}, S_{d2}, \ldots, S_{dt}\}$ represent the combined deployment locations with $S_d = S_h \cup S_u$ and $t = q + r$. If a mobile node i is assigned to a location, j, there is a cost (energy consumption due to movement) of c_{ij}. We can minimize the total value of assignment (minimize energy consumption). The objective function can be defined as:

"What is an assignment schedule in order to minimize energy consumption?"

There are additional constraints that each mobile node can be assigned to exactly one location and each location must have one assigned mobile node. The assignment problem can be formulated as an Integer Linear Program (ILP) as follows;

$$Minimize \sum_{j=1}^{t} \sum_{i=1}^{p} c_{ij} x_{ij}$$

Constraints:

- $x_{ij} \leq 1; i = 1, \ldots, p; j = 1, \ldots, t$
- $\sum x_{ij} \leq 1; j = 1, \ldots, t$
- $\sum x_{ij} \leq 1; i = 1, \ldots, p$

Simulations Results. We implemented the B-node selection algorithm [10], PCA, basic VFA, CEA-VFA and simulated variants in discrete event simulator NS2. Values of Th_{push}, Th_{pull}, and d_{dis} were set at 25m, 33m, and 40m respectively. Initial energy of each node was 4528J (3.7v, 345mAh) and energy consumed in movement was taken as 8.274 J/m (Robomote [12]). Maximum number of rounds was set to 12. The results are averaged over three different topologies for each type of deployment.

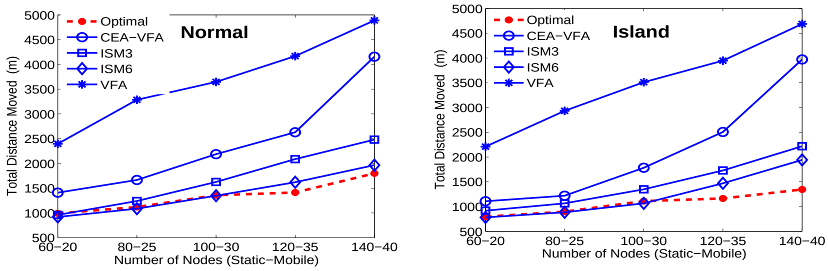

Fig. 3. Total distance moved

Figure 3 compares the total distance moved by all mobile nodes. The results show that CEA-VFA and the simulated variants perform better than the basic VFA, for both normal and island initial deployment, for different numbers of mobile nodes. On average, CEA-VFA causes the mobile nodes to move about 63% and 57% of the total distance moved in case of basic VFA for normal and island deployments respectively. This saving is primarily due to the coverage

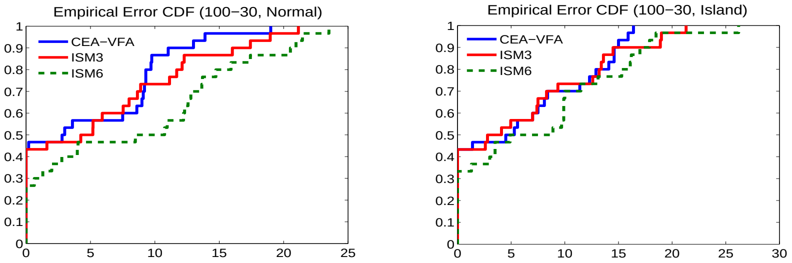

Fig. 4. Empirical error CDF for Normal and Island distributions

awareness that enables CEA-VFA to discover and plug the coverage holes in each round of the VFA. Comparing CEA-VFA with the simulated variants, CEA-VFA causes the mobile sensors to move the highest total distance for all types of topologies and initial deployments. On the other hand, mobile nodes using ISM6 variant consistently moves the least distance. This is because in ISM6, the mobile sensors only moves in two of the twelve rounds. Also note that in some cases, mobile nodes using ISM6 moves a lesser total distance than that given by optimized assignment (e.g., 60-20, 80-25, and 100-30 nodes topologies, for both normal and island initial deployments). This is because in these cases ISM6 not only fails to discover all of the coverage holes in the topology but it also produces a poor topology distribution by moving lesser distances.

Figure 4 illustrates the empirical error Cumulative Distribution Function (CDF) for both normal and island distributions for 100 static-30 mobile nodes topologies. Errors are calculated as the difference between the desired deployment points and the final topology position achieved by different virtual force based algorithms. A zero error means that either a coverage hole has been plugged or a perfect grid point deployment has been achieved. For island distribution, the error CDF of ISM3 matches closely with that of CEA-VFA while ISM3 moves about 55% of the total distance moved by CEA-VFA. For ISM6, lesser number of nodes (about 27-33%) report zero deployment error than ISM3 and CEA-VFA (about 43-47%). Also the spread in error CDF is more for ISM6 than either ISM3 and CEA-VFA.

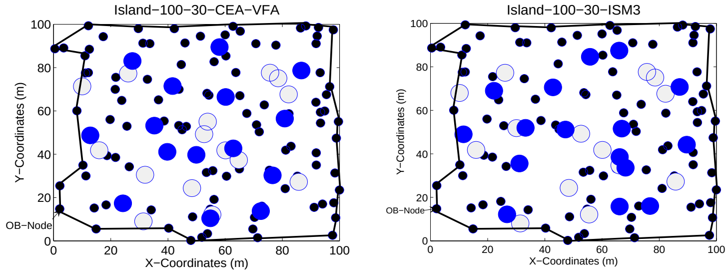

Fig. 5. Topology distribution achieved with CEA-VFA and ISM3

Topology distributions achieved by CEA-VFA and ISM3 algorithms are shown in Figure 5 (we only illustrate the 100 static-30 mobile nodes scenario for island distribution). Small filled circles represent static nodes, boundary nodes are shown by connected virtual boundary. The mobile nodes are initially introduced from the bottom-left and top-right corners of the topology. Light grey filled big circles represent the mobile nodes that have been utilized for coverage improvement while dark filled big circles are the final position of redundant mobile nodes.

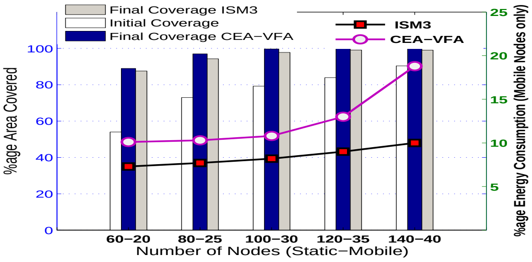

Fig. 6. Percentage of area covered and energy consumed

Figure 6 shows the initial and final percentage of area with sufficient coverage (shown by bars) for CEA-VFA and ISM3 with different topologies. Static nodes are randomly deployed in a 100m by 100m region. Static nodes estimate their coverage using PCA with required coverage probability (ρ_{reqd}) of 0.9 and request assistance from mobile nodes if they find uncovered perimeters. Mobile nodes spread out from an initial island distribution. For 100-30, 120-35, and 140-40 static-mobile node cases, more than 99% of the area is covered after relocation of mobile nodes for both CEA-VFA and ISM3. For 80-25 configuration, coverage is enhanced from 72% to 94% after execution of 12 rounds of the ISM3 algorithm while the corresponding coverage in CEA-VFA is 96.7%. This gain in coverage is at the expense of energy consumption due to relocation of the mobile nodes. Figure 6 also shows that for 140-40 topology, mobile nodes using ISM3 only consume about 10% of their total initial energy in the deployment phase as compared to close to 19% for CEA-VFA (shown by lines).

Simulation results show that ISM variants save a considerable amount of energy by moving the mobile nodes lesser distances than the VFA and CEA-VFA. However, this saving is at the cost of slight non-uniformity in the node distribution. Performance of ISM3 is comparable to ISM6 in terms of energy consumption and yet it achieves topology distribution closer to that of CEA-VFA with similar error CDF. To summarize, the simulation results show that ISM3 is a good compromise with significant savings in energy consumption.

6 Conclusion

In this paper, we have proposed a distributed protocol MAPC, for providing adequate coverage of the target area using a combination of mobile and static sensor nodes. Most of the assumptions made in our protocol are realistic and implementable in real-life applications. Our discrete event simulation results demonstrated that, for different type of initial deployments, our protocol consumes only

30-40% of the energy consumed by the basic virtual force algorithm. MAPC is thus successful in enhancing area coverage to the desired degree while ensuring that minimal energy is expended by the mobile nodes.

In future, we plan to extend the protocol to incorporate obstacles during coverage calculation and the mobile node relocation phases. We also plan to carry out experiments to validate the working of the proposed protocol.

References

1. Andrew Howard, Maja J Mataric, and Gaurav S Sukhatme. Mobile sensor network deployment using potential fields:A distributed, scalable solution to the area coverage problem. In *6th International Symposium on Distributed Autonomous Robotics Systems (DARS02)*, June 2002.
2. Guiling Wang, Guohong Cao, and Tom La Porta. A bidding protocol for deploying mobile sensors. In *11th IEEE International Conference on Network Protocol ICNP '03*, pages 315–324, Nov 2003.
3. Guiling Wang, Guohong Cao, and Tom La Porta. Movement-assisted sensor deployment. In *IEEE INFOCOM 2004*, June 2004.
4. Andrew Howard, Maja J Mataric, and Gaurav S Sukhatme. An incremental self-deployment algorithm for mobile sensor networks. *Autonomous Robots,Special Issue on Intelligent Embedded Systems*, 13(2):113–126, Sep 2002.
5. Maxim A. Batalin and Gaurav S. Sukhtame. Coverage, exploration and deployment by a mobile robot and communication network. *Telecommunication Systems Journal, Special Issue on Wireless Sensor Networks*, 26(2):181–196, 2004.
6. Yi Zou and Krishnendu Chakrabarty. Sensor deployment and target localization in distributed sensor networks. *ACM Transactions on Embedded Computing Systems*, 2(3):1–29, 2003.
7. Guiling Wang, Guohong Cao, and Tom La Porta. Proxy based sensor deployment in mobile sensor networks. In *MASS-04*, pages 493–502, Oct 2004.
8. Nadeem Ahmed, Salil S. Kanhere, and Sanjay Jha. Probabilistic coverage in wireless sensor networks. In *LCN'05*, pages 672–681, Nov 2005.
9. Ivan Stojmenovic and Pedro Eduardo Villaneuva Pena. A scalable quorum based location update scheme for routing in ad hoc wireless networks. Technical Report TR-99-09, SITE, University of Ottawa, 1999.
10. Nadeem Ahmed, Salil S. Kanhere, and Sanjay Jha. Efficient boundary estimation for practical deployment of mobile sensors in hybrid sensor networks. In *MASS'06*, pages 662–667, Oct 2006.
11. David Moore, John Leonard, Daniela Rus, and Seth Teller. Robust distributed network localization with noisy range measurements. In *Proceedings of the ACM SenSys '04*, 2004.
12. K. Dantu, M. Rahimi, H. Shah, S. Babel, A. Dhariwal, and G. Sukhatme. Robomote: Enabling mobility in sensor networks. Technical Report CRES-04-006, University of Southern California, 2004.
13. T. S. Rappaport. *Wireless communications : Principles and Practice.* Prentice Hall, New Jersey, 1996.
14. Marco Zuniga and Bhaskar Krishnamachari. Analyzing the transitional region in low power wireless links. In *SECON '04*, Oct 2004.