A Novel Distributed Mobility Protocol for Dynamic Coverage in Sensor Networks

Thakshila Wimalajeewa
Dept. of Electrical Engineering and Computer Science
Syracuse University, Syracuse, NY 13244
Email: twwewelw@syr.edu

Sudharman K. Jayaweera
Dept. of Electrical and Computer Engineering
University of New Mexico, Albuquerque, NM 87131
Email: jayaweera@ece.unm.edu

***Abstract*—In this paper, we propose a novel mobility protocol for mobile node navigation in a hybrid sensor network consisting of both static and mobile nodes to improve the dynamic coverage. Use of mobile nodes in sensor networks for coverage improvement is suggested in recent research. However, most of the existing literature on hybrid sensor networks considered the use of node mobility at the deployment stage in which nodes do not move after the initial deployment. In this paper, our focus is on efficiently managing the node mobility to provide dynamic coverage in hybrid sensor networks compensating the lack of coverage provided by static nodes. The key feature of the proposed mobility protocol is that, mobile nodes are directed to move to maximize the *coverage-time* of the uncovered area by static nodes. The proposed mobility protocol can be implemented distributively by collaborating among mobile and static nodes locally. The effectiveness of the proposed mobility protocol is shown in terms of the presence probability matrix and *coverage-time*.**

I. INTRODUCTION

When sensor nodes are first deployed in a region, a random placement is often desirable especially when a priori knowledge of the terrain is unavailable. However, such random deployment strategies may not result effective coverage always, since some nodes might be overly clustered while some of them might be sparsely located. Exploiting node mobility to reconfigure the node locations to improve the coverage of such networks was addressed by some authors, for example in [1]–[7]. In these approaches, nodes move only during the deployment stage and the maximum coverage area achieved by the network after reconfiguration is limited by the number of total nodes and nodes' sensing ranges. For example, if the total number of nodes is relatively small, even by reconfiguration of mobile nodes to provide a uniform coverage, a large portion of the network may be remained not-covered. On the other hand, node failures after the initial reconfiguration might cause coverage holes in the network. Liu et. al. in [8] showed that the coverage can be improved by allowing nodes to be mobile continuously in a mobile sensor network over time compared to that with a static network. Use of node mobility after initial deployment in detection and tracking applications was addressed by some recent work [9], [10] in different perspectives. However, neither of the above work addressed the problem of how to efficiently use the node mobility to compensate for the lack of coverage resulted by static nodes dynamically in a hybrid sensor network.

In this paper, we propose a new distributed mobility protocol for mobile node navigation in a hybrid sensor network by collaborating with static nodes to provide an efficient dynamic coverage for the area not covered by the static nodes. We assume that the sensor network is partitioned into square cells such that a node can cover a cell completely when it is located at the cell center. We divide these cells into two categories and name them as, *static* and *void* cells. *Static* cells correspond to the cells in which there is at least one static node, and the *void* cells are the ones in which there is no any static node. Mobile nodes are directed to move among these *void* cells based on a certain criteria. Each *void* cell is given a certain *base price* which reflects a value corresponding to the criteria for mobile node navigation. This *base price* is updated by static nodes based on the time that the *void* cell remains not-covered by at least one mobile node. At each movement step, mobile nodes communicate with their closest static nodes locally to search for *void* cells which have not been covered for a long time. Static nodes provide necessary information for mobile nodes in their neighborhoods. At a given time, we assume that a mobile node can visit a certain number of candidate *void* cells from its current position. These candidate *void* cells are determined by the mobile node's maximum speed. Taking base prices (collected from neighboring static nodes) of the candidate *void* cells into account, each mobile node selects the best *void* cell to move by the next time step, as the one that is not covered for a long time. We show, from simulation results, the effectiveness of the proposed scheme in terms of the presence probability matrix and the average time that an arbitrary point in the network is not covered. The presence probability matrix contains the probabilities of the presence of at least one node at each cell at any given time instant.

The paper is organized as follows: Section II presents the network model. In Section III, the proposed mobility protocol is described. Performance results are shown in Section IV and the concluding remarks are given in Section V.

II. SENSOR NETWORK MODEL

We consider a hybrid sensor network made of N number of total sensor nodes deployed in a region $\mathcal{R}$ with network dimension of $b \times b$. We assume that there are N_s number of static nodes and N_m number of mobile nodes. Denote $\lambda = \frac{N}{b^2}$ to be the spatial density of the nodes and $\lambda_m = \frac{N_m}{N}$ and $\lambda_s = \frac{N_s}{N}$ to be the fractions of mobile and static nodes respectively. Let $\mathcal{V}$ be the set containing all node indices in the network and let $\mathcal{V}_m$ and $\mathcal{V}_s$ be the sets containing mobile and static node indices, respectively.

Suppose that the sensing region is divided into a square grid with a grid length of $l = \sqrt{2}r$ where r is the effective sensing radius of a sensor. We assume both static and mobile nodes have the same sensing radii and the analysis can be slightly modified to deal with different sensing radii for mobile and static nodes. When a sensor node is located at the center of a cell in the grid the corresponding cell is completely covered by the sensor node. Consider the hybrid network with only static nodes as shown in Fig. 1. We denote the set of cells that is not covered by the static nodes as the set of *void* cells as shown in Fig. 1 with void squares. When a static node is located in a particular cell (crossed cells in Fig. 1) we consider that the corresponding cell is covered by the relevant static node and call that cell a *static* cell. However, note that since a static node does not necessarily locate at the middle of a cell, corresponding cell may not be completely covered by the static node. We address this problem later and for the moment assume that the cell is covered by the corresponding static node. Now the problem is how to use the mobile nodes efficiently to cover the *void* cells as shown in Fig. 1 over time

978-1-4244-5638-3/10/$26.00

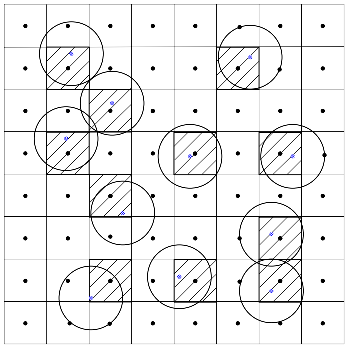


Fig. 1. Sensor network with only static nodes

such that revisiting time of any cell by at least one mobile node is maximized. In the following, we propose a new distributed interactive protocol to achieve the required task by collaborating among mobile and static nodes.

III. INTERACTIVE, DISTRIBUTED MOBILITY PROTOCOL

A. *Introduction*

We assign a base price for each *void* cell according to the following rule. Initially, at time $t = 0$, we assign a base price $\mathcal{P} = 0$ for each *void* cell in which there is at least one mobile node. For all the other *void* cells we assign $\mathcal{P} = K$ where K is a large value. Let T_m be the time interval in which the mobility management is performed, which can be determined by the mobile node's maximum speed and the length of a grid. At each time step T_m, the base price of each *void* cell is updated considering the time it remains uncovered (or unvisited by a mobile node). More specifically, at each step T_m, if a particular cell is visited by a mobile node, its base price $\mathcal{P}$ is set to zero and the base prices of all other *void* cells are increased by 1 unit.

Without loss of generality we assume that at time $t = 0$ each mobile node has moved to the cell center which it belongs to, and at each step T_m, mobile nodes move among cell centers. In the following we explain how a mobile node selects the best cell to be visited at each time step distributively by collaborating with static nodes.

1) Determining T_m: We assume that any mobile node can reach $L_c = 8$ number of closest distinct cell centers and itself at any given time step. Then the maximum distant that a node has to move during time T_m is $2r$. Thus it is desirable to choose the time step T_m as $T_m = \lceil \frac{2r}{v_{max}} + \epsilon \rceil s$ where ϵ is a bias factor which accounts for the scenarios when it is needed to heal the lack of coverage at *static* cells which will be explained in subsection III-D in detail.

Let each cell (cell center) in the square grid be given an ID labeled by indices $1, 2, \cdots, L_T$ where $L_T \approx \frac{b^2}{l^2}$ is the total number of cells. Let the number of cells covered by static nodes (*static* cells) be L_s and $L_v = L_T - L_s$ be the number of cells that are not covered by static nodes (*void* cells). Also denote $\mathcal{U}$, $\mathcal{U}_s$ and $\mathcal{U}_v$ to be the sets containing all cell indices of the network, *static* cell indices and *void* cell indices, respectively.

We assign a certain number of cells to each static node in the network. Each static node in the network is responsible for updating the base price for each cell that belongs to it. Corresponding cells for each static node are assigned based on Voronoi partitions. According to Voronoi partitions, any point inside a Voronoi polygon of a static node is closer to that node rather than to any other static node in the network. Thus for a given static node s_k, the cell centers belonging to its Voronoi polygon are closer to the static node s_k than any other static node in the network. We assume that each static node has the knowledge of the positions of the *void* cell centers belonging to itself. Note that at the initial stage, static nodes can communicate with their Voronoi neighbors locally to construct Voronoi polygons. By knowing its own location, based on the grid length (in terms of the sensing range) each static node can determine the *void* cells in its Voronoi polygon. Denote $\mathcal{U}_{s_k}$ to be the set of *void* cell indices belongs to the Voronoi polygon of the static node s_k for $s_k \in \mathcal{V}_s$ and $L_{s_k} = |\mathcal{U}_{s_k}|$ be the number of *void* cells (cell centers) belongs to static node s_k. Note that we have then $\mathcal{U}_v = \bigcup_{k \in \mathcal{V}_s} \mathcal{U}_{s_k}$. Further denote $\mathbf{g}_{s_k}(nT_m)$ to be the L_{s_k}-length vector containing the base prices for all *void* cells attached to the static node s_k at time nT_m for $k \in \mathcal{V}_s$. Each static node s_k is responsible for updating $\mathbf{g}_{s_k}(nT_m)$ at each time step $t = nT_m$ for $n = 1, 2, \cdots$.

B. *Updating $\mathbf{g}_{s_k}(nT_m)$*

1) At time $t = 0$: At time $t = 0$, each mobile node broadcasts its current location (or equivalently current cell ID) to its neighborhood. The static nodes located close to the mobile node receive this information and if the corresponding mobile node's cell ID belongs to $\mathcal{U}_{s_k}$ then the static node s_k sets the base price for the corresponding cell to zero. Base prices for all the other cells in $\mathcal{U}_{s_k}$ are set to a large integer number K. Note that at time $t = 0$, all *void* cells which have no mobile node at time $t = 0$, have the same base price K.

2) At time $t = nT_m$, $n \geq 1$: At time $t = nT_m$, each mobile node broadcasts its location information (current cell ID) to its nearest static nodes. Based on this information, each static node updates base price vector $\mathbf{g}_{s_k}(nT_m)$ as follows: Let $N_{m,k}(nT_m)$ be the number of mobile nodes that the static node s_k receives location information at time nT_m and $\mathcal{U}_{m,k}(nT_m)$ be the set corresponding to those locations (cell indices). Then for a given static node s_k for all cell indices $c_j \in \mathcal{U}_{s_k}$, it checks whether c_j also belongs to $\mathcal{U}_{m,k}(nT_m)$. If yes static node s_k sets the base price of the cell c_j to be zero otherwise static node increases the c_j-th cell's base price by 1 unit.

After updating the base price vector $\mathbf{g}_{s_k}(nT_m)$ at time nT_m at each static node s_k, the problem is to determine the next cell ID to be visited by each mobile node by time $t = (n+1)T_m$ such that the cell-revisiting time is maximized. Denote $\mathcal{C}_{m,j}(nT_m)$ to be the set of candidate locations (cells) of the j-th mobile node at time nT_m. Also let $\mathcal{U}_{s_k}^{m_j}(nT_m)$ be the set of cell indices belongs to both $\mathcal{C}_{m,j}(nT_m)$ and $\mathcal{U}_{s_k}$. Note that the maximum size of the set $\mathcal{U}_{s_k}^{m_j}(nT_m)$ is $|\mathcal{U}_{s_k}^{m_j}(nT_m)|_{max} = L_c + 1 = 9$, since we assume that each mobile node can move to one of the 8 distinct candidate locations and itself during a given time step. For a given mobile node m_j from which the static node s_k receives the location information, the static node s_k checks whether any cell in m_j-th candidate set $\mathcal{C}_{m,j}(nT_m)$ belongs to $\mathcal{U}_{s_k}$ at time $t = nT_m$. If not, static node s_k does not need to communicate with mobile node m_j at time nT_m. If yes, or in other words, if the set $\mathcal{U}_{s_k}^{m_j}(nT_m)$ is not empty, the static node s_k queries the mobile node m_j to check whether m_j is isolated with respect another mobile node. We call the mobile node m_j is isolated with respect to another mobile node, if there is no at least one mobile node within a distance d_t from its current location where d_t (equals to $4r$) is a threshold distance determined such that no duplicate covering occurs as discussed in subsection III-C. We assume that the mobile node m_j can communicate locally with other mobile nodes within a distance of d_t to check whether it is isolated. Note that in the rest of the paper a mobile node is isolated means that the mobile node is isolated with respect to another mobile node. If m_j is isolated, static node s_k finds the cell from the set $\mathcal{U}_{s_k}^{m_j}(nT_m)$ which has the maximum base price and sends a message corresponding to the cell ID and the maximum corresponding base price. Note that all the candidate cells for mobile node m_j may not belong to one static node. In particular, they may belong to multiple near-by static nodes. Once the mobile node m_j gets maximum base prices from multiple static nodes in which its candidate cells belong to, it selects the best location for time $(n+1)T_m$ by comparing the base prices it gets from different static nodes and selects the one with maximum base price.

If the mobile node m_j is not isolated (that is there are other mobile nodes very close to it) there might be situations which lead to duplicate covering; that is two or more mobile nodes may try to go to the same cell at time $(n+1)T_m$. To combat this problem (as discussed in subsection III-C), when a mobile node is not isolated, each static node s_k sends all the candidate cell IDs in the set $\mathcal{U}_{s_k}^{m_j}(nT_m)$ and their base prices to the mobile node to assist in resolving the duplicate covering problem.

C. Duplicate covering at a given time

When two mobile nodes are close to each other there might be situations where both select the same *void* cell as the candidate location. For example, consider the scenario as depicted in Fig. 2 where two mobile nodes try to heal the same cell. It can be shown that this might happen when two mobile nodes are located within a maximum distance of $d_t = 2\sqrt{2}l = 4r$. Assume that two mobile nodes m_1 and m_2 are located in cells represented by A and B at time $t = nT_m$ as shown in Fig. 2. According to the information received from closest static nodes, both mobile nodes can access to the base prices of all of their candidate cells, for example marked at the north-east corner of each candidate cell for both mobile nodes in Fig. 2. According to the base prices, both will try to select the cell C as the next location for time $(n+1)T_m$ which has the highest base price from each mobile nodes' candidate sets. Since this will lead to inefficient coverage, we propose for two mobile nodes to exchange their local information to avoid duplicate covering. Since this phenomenon occurs when two mobile nodes are located close to each other, we assume that these two mobile nodes can exchange their information to check whether a duplicate covering is going to happen. If so, they exchange the next maximum base prices from their candidate sets, and check which mobile node has the second maximum base price. Accordingly, the node with the highest maximum second base price selects the corresponding cell as the candidate cell. According to Fig. 2, since the mobile node m_1 has the second maximum base price (compared to mobile node m_2), it moves to the corresponding cell (denoted by cell D) while the mobile node m_2 moves to the cell C. If the second maximum base price is same for both nodes, they can select either one of the nodes to move to the cell with the second maximum base price arbitrarily. When there are more than 1 mobile sensors within the distance d_t from node m_j, the same procedure can be extended by exchanging the relevant information among those nodes. In such cases it might be necessary to exchange, $2nd$, $3rd$,... highest base prices among neighboring mobile nodes.

D. Compensating for the lack of coverage in a static cell

As mentioned earlier in this section, since a static node might not be located at the center of a *static* cell in the grid, there might be certain uncovered portions of the corresponding cell. Note that this uncovered portion is maximum when a static node is located very close to one of the cell corners in which it belongs to. Consider the scenario that the static node is located very close to the north-east corner of the cell it belongs to (denoted by c_1), as shown in Fig. 5 with a circle with solid line. To compensate for the lack of coverage in the corresponding cell, we propose the following procedure. It can be shown that with the relationship between the side length of a cell in the grid and the sensing range, when a mobile node comes to a cell located either to the left or to the bottom of the *static* cell, and if they are moved a distance of $r - \frac{r}{\sqrt{2}}$ (at the worst case) beyond the cell center towards the *static* cell, the corresponding *static* cell can be completely covered. This is illustrated in Fig. 5 where when a mobile node comes to either cell centers A or C, and if it is allowed to move a distance of $r - \frac{r}{\sqrt{2}}$ (i.e. to either B or D, respectively), the uncovered portion of the *static* cell can be completely covered. To address this problem, at time nT_m, when a mobile node selects its candidate cell for time $(n+1)T_m$, it also checks whether there is a static node to the right, left, up or down to the selected cell. Then based on the static node location, it approximates the required distance it should move (maximum of $r - \frac{r}{\sqrt{2}}$) beyond the selected cell center to compensate for the lack of coverage of the *static* cell.

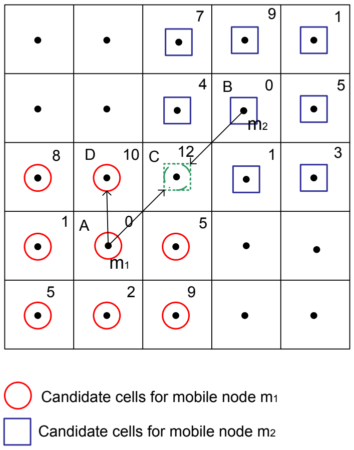


Fig. 2. Duplicate covering at a given time

Note that according to the proposed mobility algorithm we allow mobile nodes to move between cell centers at consecutive time steps T_m. However, when we need to address this *static* cell compensating problem, mobile nodes have to move little far away from a cell center. When this happens (i.e. a mobile node may move to location B (or D) instead of A (or C) in Fig. 5), the mobile node may need to move a maximum distance of $\approx 2.2168r$ to reach its next candidate cell at next time step. As shown in Fig. 5, when the mobile node is at the point D in the cell c_3, it can reach all candidate cells by next time step, except E and F by moving a maximum distance of $2r$. To reach the candidate cells E and F it has to move a maximum distance of $\approx 2.2168r$. Thus when determining the time step T_m as pointed out in subsection III-A1, we need to take this scenario into account. Thus T_m is selected as, $T_m = \lceil \frac{2r}{v_{max}} + \epsilon \rceil s$ where $\epsilon = \frac{0.2168r}{v_{max}}$.

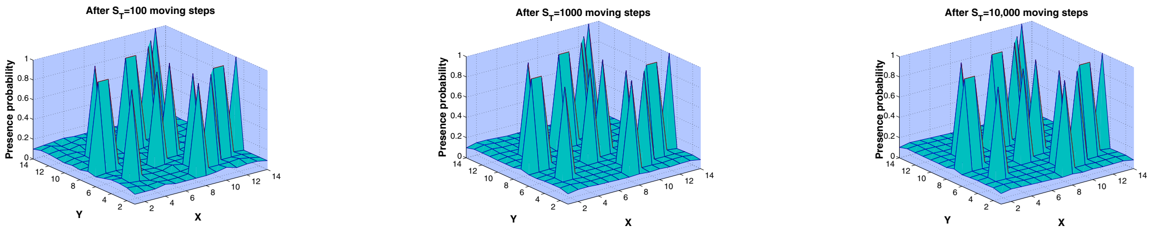
The proposed protocol for node mobility management of hybrid sensor network is summarized in Algorithm 1.

IV. PERFORMANCE EVALUATION

To evaluate the effectiveness and efficiency of the proposed mobility protocol, we perform experiments to investigate how well the desired area is covered over time to minimize the time that a *void* cell is unvisited by a mobile node. We depict the results in different perspectives as described in the following.

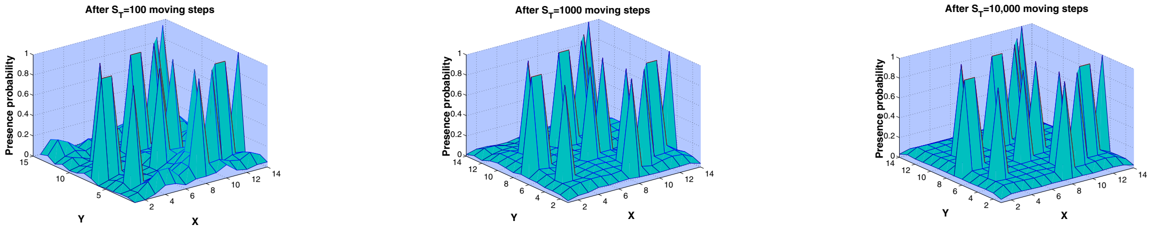
A. Presence probability at each cell

Denote p_{c_k} to be the probability that at least one node is present at the cell c_k at any given time. Let $\mathbf{\Lambda}$ be the presence probability



(a). $S_T = 100$ (b). $S_T = 1000$ (c). $S_T = 10,000$

Fig. 3. Presence probability matrix with proposed mobility protocol, $N = 40$, $\lambda_m = 0.5$, $v_{max} = 10m/s$



(a). $S_T = 100$ (b). $S_T = 1000$ (c). $S_T = 10,000$

Fig. 4. Presence probability matrix with bounced random walk model, $N = 40$, $\lambda_m = 0.5$, $v_{max} = 10m/s$

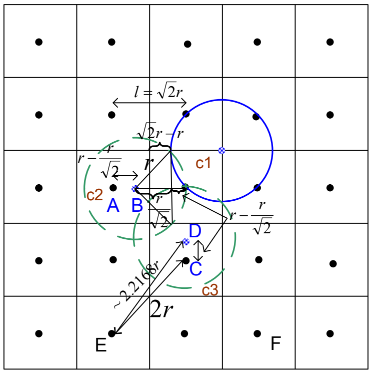


Fig. 5. Compensating for the lack of coverage in *static* cells

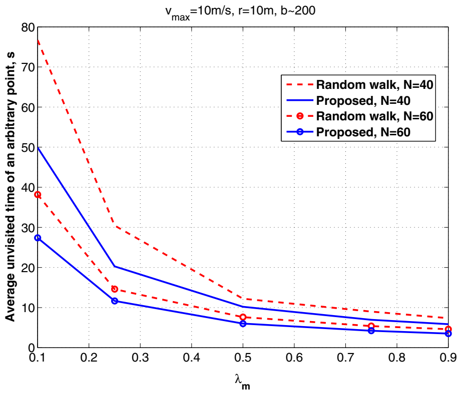


Fig. 6. Average time taken for an arbitrary point to be revisited for different N: $v_{max} = 10m/s$, $r = 10m$, $b \approx 200m$

matrix containing the probabilities of the presence of at least one node at each cell at a given time instant. For simulations, we consider a sensor network deployed in a $\approx 200 \times 200m^2$ square region with 14×14 grid. We let $r = 10m$ such that the grid length becomes $l = \sqrt{2}r \approx 14.14m$. Denote S_T to be the number of moving steps. We compare the performance of the proposed mobility protocol with bounced random walk mobility model with a step size of l. We mean by bounced random walk, that when the mobile nodes hit the boundary under random walk, they bounce back with probability 1. Figures 3 and 4 show the presence probability matrices with proposed mobility scheme and with bounced random walk scheme, respectively. The presence probability matrices are shown after completing $S_T = 100$, $S_T = 1000$ and $S_T = 10,000$ moving steps, respectively, for $N = 40$ and $\lambda_m = 0.5$. Note that in Figs 3 and 4, the high peaks with presence probability 1 reflect the presence probability of *static* cells. Looking at the presence probabilities of *void* cells under two mobility schemes, from Fig. 3 it can be seen that the presence probabilities of *void* cells are becoming uniform after completing relatively a small number of steps compared to that with random walk model (Fig. 4). When the number of movements steps is large, it can be seen from Fig. 4 that the presence probabilities of *void* cells under random walk mobility models are also becoming uniform, as expected. However, as can be seen from Figs. 3 and 4, in terms of the number of movement steps needed to achieve this uniformity the proposed protocol for hybrid sensor network outperforms the random mobility schemes.

B. *Average unvisited time of an arbitrary point*

In the next experiment, we evaluate the performance of the proposed mobility scheme in terms of the average time that any arbitrary point is uncovered by the hybrid sensor network. We compare the results of the proposed scheme with a random mobility model. Figure 6 shows the average unvisited time of an arbitrary point in the network with the proposed mobility protocol and random walk mobility model (with step size of l) for $N = 40$ and $N = 60$. In Fig. 6, we let $v_{max} = 10m/s$, $r = 10m$. It can be seen that when the fraction of mobile nodes is small, by the proposed mobility protocol for

Algorithm 1 Mobility protocol

```
NOTATIONS:
$\mathbf{g}_{s_k}(nT_m)$: base price vector at static node $s_k$ at time $t = nT_m$
$\mathcal{U}_{s_k}$: set of all void cell indices belongs to static node $s_k$
$N_{m,k}(nT_m)$: number of mobile nodes from which the static node $s_k$ receives locations information at time $nT_m$
$\mathcal{C}_{m,j}(nT_m)$: set of cell indices corresponding to candidate cells of mobile node $m_j$ at time $nT_m$
$\mathcal{U}_{s_k}^{m_j}(nT_m)$: set of cell indices belongs to both $\mathcal{C}_{m,j}(nT_m)$ and $\mathcal{U}_{s_k}$
$\mathbf{g}_{s_k}^{m_j}(nT_m)$: base price vector corresponding to cell indices in $\mathcal{U}_{s_k}^{m_j}$
$P_{j,k}^*$: element with maximum value (maximum base price) in $\mathbf{g}_{s_k}^{m_j}(nT_m)$
$c_{j,k}^*$: cell index corresponding to $P_{j,k}^*$
INITIALIZATION AT TIME $t = 0$:
  Determine $\mathcal{U}_{s_k}$ for all $k \in \mathcal{V}_s$ based on Voronoi partitions
  Initialize $\mathbf{g}_{s_k}(0)$ as in subsection III-B1
AT STATIC NODE $s_k$ AT TIME $t = nT_m$:
After receiving location (cell) information from neighboring mobile nodes:
  Update the base price vector $\mathbf{g}_{s_k}(nT_m)$ as in subsection III-B2
  for $j = 1 : N_{m,k}(nT_m)$ do
      Check $\rightarrow$ $\mathcal{U}_{s_k}^{m_j}(nT_m)$ is non-empty
      if yes then
          check $\rightarrow$ $m_j$ is isolated
          if yes then
              Find $P_{j,k}^*$ and $c_{j,k}^*$ and transmit to mobile node $m_j$
          else {$m_j$ is not isolated}
              Send cell IDs and their base prices in the set $\mathcal{U}_{s_k}^{m_j}(nT_m)$ to mobile node $m_j$
          end if
      else {no}
          Send nothing to mobile node $m_j$
      end if
  end for
AT MOBILE NODE $m_j$ AT TIME $t = nT_m$:
  Broadcast location information to neighboring static nodes
After receiving base prices for relevant candidate locations from neighboring static nodes:
  check $\rightarrow$ $m_j$ is isolated
  if yes then
      select candidate cell with maximum base price
  else {no}
      call $duplicate\_covering(m_j)$
  end if
After selecting candidate cell corresponding to time $(n+1)T_m$:
  Check $\rightarrow$ need for static cell compensation
  if yes then
      Adjust the location to be moved in the selected candidate cell according to subsection III-D
  else {no}
      Move to the center of the selected candidate cell by time $(n+1)T_m$
  end if

$duplicate\_covering(m_j)$
  Exchange local information with neighboring mobile nodes to check for duplicate covering
  if yes:(duplicate covering) then
      Exchange next highest base prices to determine the best candidate cell as in subsection III-C
  else {no:(no duplicate covering )}
      select candidate cell with maximum base price
  end if
```

hybrid sensor network, a significant performance improvement can be obtain over random walk mobility model. Note that due to the extra cost needed to deploy mobile nodes compared to static nodes, this is the most interesting scenario. As mentioned earlier in the paper, random mobility models are not well suited for hybrid sensor networks specially for small λ_m's since they may provide duplicate coverage, which results in an inefficient usage of mobile nodes. Since deploying mobile nodes is not as cost effective as deploying static nodes, it is more desirable to efficiently use the node mobility in order to improve the network coverage. However, from Fig. 6, it can be seen that when λ_m is increasing, the unvisited time with the proposed scheme is not much different from the random walk scheme since then there is a large number of mobile nodes compared to static nodes and thus the duplicate coverage caused by random walk mobility model is less. Also when the total number of nodes is increasing, it can be seen that even with a smaller fraction of mobile nodes, relatively lower unvisited time can be obtained by the proposed scheme. The performance gain of the proposed scheme over the random walk mobility model is more significant when N is smaller, that is when the network is to be covered by a small number of total nodes. For results in Fig. 6, we ran simulations for 10000s and averaged over $50,000$ arbitrary points.

Although figures are not included due to space limitations, it can be seen that especially with a lower fraction of mobile nodes, the speed of mobile nodes affects the system performance significantly compared to that with a large fraction of mobile nodes. However, irrespective of the node speed, it can be seen that with relatively small fraction of mobile nodes, the proposed mobility scheme outperforms the random mobility schemes.

V. CONCLUSIONS

In this paper we proposed an interactive, distributed protocol for mobile node navigation in a hybrid sensor network to efficiently cover the area not-covered by static nodes by maximizing the revisiting time of an arbitrary point in the network. The proposed scheme can be implemented distributively by collaborating with static nodes, having only communicating in the local neighborhood. It was shown that the proposed scheme provides an approximate uniform coverage after completing relatively small number of moving steps compared to random mobility schemes which is desirable when the network is designed for detecting targets in which the existence is unknown. The proposed scheme outperforms the random mobility schemes especially when the fraction of mobile nodes is small.

REFERENCES

[1] Y. Zou and K. Chakrabarty, "Sensor deployment and target localization based on virtual forces," *Proc. INFOCOM*, pp. 1293–1303, 2003.

[2] G. Wang, G. Cao, and T. L. Porta, "Movement assisted sensor deployment," *IEEE Trans. Mobile Computing*, vol. 5, pp. 640–652, 2006.

[3] A. Howard, M. J. Mataric, and G. S. Sukhatme, "Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem," in *6th Int. Symp. on Distributed Autonomous Robotics Systems (DARS02)*, Fukouka, Japan, June 2002.

[4] S. Chellappan, X. Bai, B. Ma, and D. Xuan, "Sensor network deployment using flip-based sensors," in *Proc. of IEEE Mobile Sensor and Ad-hoc and Sensor Systems (MASS)*, Nov. 2005, pp. 291–298.

[5] J. Wu and S. Wang, "Smart: A scan based movement-assisted deployment method in wireless sensor networks," in *IEEE Conf. on Computer Communications (INFOCOM)*, Miami, FL, March 2005.

[6] G. Wang, G. Cao, and T. LaPorta, "A bidding protocol for deploying mobile sensors," in *Proc. 11th IEEE Int. Conf. on Network Protocols*, Nov. 2003, pp. 315– 324.

[7] W. Wang, V. Sirinivasan, and K.-C. Chua, "Trade-offs between mobility and density for coverage in wireless sensor networks," in *Proc. 13th annual ACM int. conf. on Mobile computing and networking*, Montral, Qubec, Canada, 2007, pp. 39–50.

[8] B. Liu, P. Brass, O. Dousse, P. Nain, and D. Towsley, "Mobility improves coverage of sensor networks," in *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, 2005, pp. 300–308.

[9] R. Tan, G. Xing, J. Wang, and H. C. So, "Collaborative target detection in wireless sensor networks with reactive mobility," in *16th Int. Workshop on Quality of Service (IWQoS)*, University of Twente, Enschede, Netherlands, June 2008.

[10] O. Kosut, A. Turovsky, J. Sun, M. Ezovski, L. Tong, and G. Whipps, "Integrated mobile and static sensing for target tracking," in *Military Communication Conference, MILCOM*, Oct. 2007, pp. 1–7.