Brief Announcement: Distributed Algorithms for Dynamic Coverage in Sensor Networks

Lan Lin
Department of Computer Science
University of Denver
Denver, Colorado 80208
llin@cs.du.edu

Hyunyoung Lee
Department of Computer Science
University of Denver
Denver, Colorado 80208
hlee@cs.du.edu

Categories and Subject Descriptors

G.2.3 [**Mathematics of Computing**]: Discrete Mathematics—*Applications*

General Terms

Algorithms

Keywords

distributed algorithms, sensor networks, sensor coverage

The coverage problem is of great interest for many sensor network applications, for example, detection of intruders in the sensor field. Topological changes in sensor networks may affect qualities of sensor coverage. We present three suites of algorithms for dynamically maintaining the coverage and the measures of its qualities. Using only local knowledge, our algorithms capture the dynamic changes of network topology and efficiently maintain the coverage by updating the radii of sensors combined with limited sensor mobility. Our algorithms have the advantages of low communication complexity with no need of a tight bound on message propagation delay.

1. DYNAMIC MAINTENANCE OF BEST AND WORST COVERAGE RADII

We assume that sensors have the same sensing and communication range, where the communication range is twice of the sensing range [3]. The sensor network of this kind is a uniform sensor network. Sensors are represented by a set of points P in $\mathbb{R}^2$. Let $D(P, r)$ be the union of all disks centered at points of P with radius r. Let $\overline{D(P, r)}$ be the complement of $D(P, r)$. Given a pair of points S and T: The *best coverage radius* [1] is the minimum coverage radius r such that there exists a trajectory between S and T that is totally covered by $D(P, r)$, shown as the dotted line in Fig. 1 Left. The *worst coverage radius* [1] is the maximum coverage radius r such that there exists a trajectory between S and T that is covered by $\overline{D(P, r)}$, shown as the dotted line in Fig. 1 Right.

Topological changes to the network such as the loss of sensor nodes may cause changes to the best and/or worst coverage radii. Examples of the breaches to best and worst coverage radii are shown in Figure 1 with the failed nodes in dotted circles. For the best coverage radius, the radius is to be updated when no trajectory between S and T is completely covered by $D(P, r)$, i.e., $D(P, r)$ is divided into two disconnected regions. For the worst coverage radius, the radius is to be updated when the coverage is divided into two regions, either disconnected, or tangent to each other.

To maintain the topological information of the network, we keep track of the changes to the boundary of $D(P, r)$ in four cases: loss of a node on the boundary, moving a node on the boundary to another location, loss of a non-boundary node that causes a hole not covered by any nodes, and, moving a non-boundary node within range of some boundary nodes. Each sensor on the boundary determines if itself is a *critical* node where its disk is not continuously covered by its neighboring nodes. It notifies its neighbors of its critical node status.

When a critical node is lost a breach occurs and causes the division of its neighboring nodes into two disconnected sets. To restore the radius the neighboring nodes first determine the initial new radius by calculating the minimum radius to connect the two disconnected sets using computational geometry algorithms for finding the closest pair of points between two sets of points. Two leaders are elected, each from one of the two disconnected sets, to broadcast the new radius. There are many leader election algorithms, we choose to use node IDs for simplicity. Upon receiving the radius, a boundary node of $D(P, r)$ increases its radius to the new radius and broadcasts a message for communicating with boundary nodes of the other region. If a node of the other region receives the message, it responds with an updated radius of half of the distance between the two nodes, i.e., when their disks are tangent to each other. They then broadcast the updated radius to their neighbors. Note that the updated radius can only be less than the received radius. The process stops when the radius is received by the two leaders. The radius is the new radius and is propagated throughout the network.

We assume uniformly distributed sensors, thus the number of boundary nodes is $O(\sqrt{n})$. The message carrying the radius information is sent and received only twice by each boundary node. Since each message carries the sender ID, which costs $O(\log n)$, the total message complexity for finding the best coverage radius is $O(\sqrt{n} \log n)$.

2. DYNAMIC COVERAGE WITH MIGRATION OF REDUNDANT NODES

Nodes in a sensor network can be divided into two sets: redundant nodes and non-redundant nodes. A *redundant node* is a sensor whose sensing area is completely covered

PODC'07, August 12–15, 2007, Portland, Oregon, USA.
ACM 978-1-59593-616-5/07/0008.

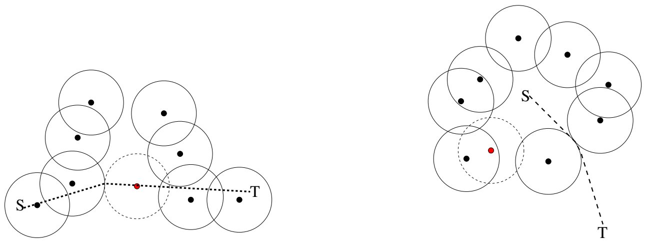

Figure 1: Left: example of best coverage radius. Right: example of worst coverage radius.

by some other sensors. A *non-redundant node* is a sensor whose sensing area is partly covered by the sensor alone. We keep track of the closest redundant node to each non-redundant node such that when a non-redundant node fails the redundant node is moved to its location for replacement.

We construct a tree rooted at each redundant node. Each non-redundant node determines the closest redundant node as follows. If a non-redundant node has redundant nodes in its neighbors, it connects to the closest redundant node and stores its location. Otherwise it connects to the closest neighboring non-redundant node whose closest redundant node is minimum distance to itself. We call the structure a communication tree. We prove that this structure is indeed a tree with the property that any node of the tree is closer to the root node in distance than to any other redundant node. We develop schemes for inserting, deleting, and updating of non-redundant nodes and redundant node of the tree.

3. LAZY COVERAGE WITH DIFFERENT RADII AND LIMITED MOBILITY

The main idea is that when node failures are detected sensors do nothing until a threshold of percentage of uncovered area is reached. To recover the area, sensors increase their radii and move a limited distance with small total energy consumption. Migrating the multi-hop neighbors of a dead node to recover the area was proposed [2]. Our approach combines mobility with different radii. We consider the area that each sensor participates in coverage as the area covered by the sensor and its one-hop and two-hop neighbors. Our algorithms determine the uncovered area and its percentage and calculate the best strategy for recovering the area.

Lazy failure detection.

We define the energy cost of sensor movement $E_m = k \cdot d$ as the distance moved d times a constant k, where k is determined by the particular vehicle. The energy cost of sensing E_s with radius r is $E_s = l \cdot r^2$ where l is a constant specified by the sensing equipment. The total cost of moving sensors to different locations with enlarged radii is $E_{total} = \sum_{i=1}^{n} E_{m,i} + \sum_{i=1}^{n} E_{s,i}$, where n is the number of sensors. We assume that initially the field is completely covered by sensors. To maintain network connectivity, a sensor periodically exchanges among its neighbors beacons containing the locations of all of its neighbors. Thus each sensor knows its two-hop neighbors and the percentage of area covered within two-hop range. When the percentage of coverage falls below certain threshold, a coverage maintenance scheme is used to recover the lost area. Assume that a failed node X has m neighbors, S_1, S_2, …, S_m. Let the area covered by each node be $A(S_i)$, for $i = 1, 2, \ldots m$. Let the area covered by X be $A(X)$. We need to find the combination of movements of the neighbors to new locations P_1, P_2, …, P_m and employments of different sensing range for S_1, S_2, …, S_m, such that the following conditions are fulfilled. 1. $A(X) \cap (\cup_{i=1}^{m} A(S_i))$ is maximized, and 2. $E_{total} = \sum_{i=1}^{m} E_{m,i} + \sum_{i=1}^{m} E_{s,i}$ is minimized. The complexity of the problem is exponential. We propose recovery schemes under various constraints and assumptions.

Different radii without mobility.

Our scheme allows different radii for different nodes with no mobility of the nodes. Each node is only responsible for recovering its one-hop neighbors since its two-hop neighbors are one-hop neighbors of some other nodes. We propose a scheme in which the node increases its sensing and communication range in $1+\varepsilon$ intervals until one of the two conditions are met: (1) with the new radius the coverage is restored to a certain percentage, or (2) it detects at least one other node recovering part of the lost area, and the union of their disks restores the coverage to a certain percentage. Different radii cause asymmetric communication, that is, node with smaller radius can receive message from node with bigger radius but not vice versa. In order to maintain connectivity we propose the clustering of nodes with smaller radii in a tree structure around the node with bigger radius.

Different radii with mobility.

For a sensor to recover a lost area while maintaining connectivity with its current neighbors, we propose a scheme that moves the sensor on the straight line connecting itself and the lost node and increases its radius in $1+\varepsilon$ intervals. The distance moved is equal to the radius increased such that the connection with neighbors is maintained.

For a node with radius r to cover a point d distance from the disk of the node, we calculate the energy cost for the two schemes as follows. For radius without mobility scheme, the energy cost $E_R = l \cdot (r + d)^2$. For radius with mobility scheme, the energy cost $E_M = l \cdot (r + \frac{d}{2})^2 + k \cdot \frac{d}{2}$. If $\frac{k}{l} > \frac{3}{2}d + 2r$, $E_M > E_R$; the node should only increase its radius. If $\frac{k}{l} < \frac{3}{2}d + 2r$, $E_M < E_R$; the node should increase the radius and move the same distance.

Acknowledgment. This research was partly supported by NSF grant CNS-0614929.

4. REFERENCES

[1] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava. Coverage problems in wireless ad-hoc sensor networks. *Proc. of the 20th IEEE INFOCOM*, pages 1380–1387, 2001.

[2] A. Sekhar, B. S. Manoj, and C. S. R. Murthy. Dynamic coverage maintenance algorithms for sensor networks with limited mobility. *Proc. of the 3rd IEEE Int'l Conf. on Pervasive Computing and Communications*, 2005.

[3] H. Zhang and J. C. Hou. Maintaining sensing coverage and connectivity in large sensor networks. *Ad Hoc & Sensor Wireless Networks*, 1:89–124, March 2005.