

# Coverage Protocol for Wireless Sensor Networks Using Distance Estimates

Mingze Zhang, Mun Choon Chan and A. L. Ananda  
School of Computing, National University of Singapore  
Email: {zhangmi3,chanmc,ananda}@comp.nus.edu.sg

**Abstract**—In wireless sensor networks, it is desirable to maintain good sensing coverage while keeping the number of active sensor nodes small to achieve long system lifetime. Existing coverage algorithms assume some form of localization and that complete coverage is needed. Our work is motivated by two observations. First, localization is expensive, error-prone and not required for a coverage algorithm. Instead, coverage algorithm can be designed using only distance information between two nodes. Second, complete coverage can lead to excessive redundancy, resulting in inefficiency in terms of active nodes needed.

In this paper, we first present a scheme that estimates the distance between any two neighboring nodes using only local information. We then present *CCP*, a configurable coverage protocol that requires only neighbor distance information. CCP can be configured such that at least  $\alpha$  portion of the area will be covered by active nodes with high probability, where  $\alpha$  is a tunable parameter. In addition, CCP achieves similar performance to OGDC [1] in terms of coverage and active nodes required. Compare to OGDC, CCP offers the flexibility to trade-off between coverage and nodes required. By setting the coverage objective to 90%, about 22% node savings can be achieved.

## I. INTRODUCTION

Low-cost sensor devices are failure-prone. In typical sensor networks, these devices are deployed in higher than necessary densities to meet various design specifications. In order to conserve energy and prolong network lifetime, at any time instance, only a portion of these sensors are active while others operate in “sleep” mode. However, if too many nodes are turned off, there may be a large area that cannot be monitored. Therefore, one important research challenge in wireless sensor network (WSN) is the coverage problem, in which the number of active sensors needed to cover the area of interest is to be minimized.

Sensor network coverage is an active research area. Most existing protocols [1]–[7] work on complete coverage (either 1- or  $k$ -coverage) and can be centralized [2] or distributed [1], [3], [4]. These protocols also assume

accurate location information to be available. Our work is motivated by two observations. First, global localization is expensive, error-prone and is not necessary for a coverage algorithm. Instead, coverage algorithm can be designed using only distance information between two nodes. Second, by requiring complete coverage, excessive overlap may occur. By reducing the coverage requirement, substantial savings can be achieved.

Localization can be performed in a number of ways. First, localization can be performed with a node in direct contact with three or more high power beacon/anchor nodes which know their accurate locations [8]–[10]. However, such systems can be expensive and the performance depends highly on the number of beacons available and accuracy of the distance estimation to the beacons. In GPS [8] and Cricket [9], each node has to be equipped with the appropriate receiver. For GPS, the devices are expensive and power consuming, and cheap GPS devices may have large errors [8]. Cricket [9] is designed for in-door use, is short range and requires line-of-sight. In [10], the localization scheme may require a fairly large number of anchor nodes to be heard by any node who needs accurate location. For example, more than 10 anchor nodes are needed to have an error of less than 50% of the communication range. In [11], the localization scheme requires nodes to estimate their distances to the beacon nodes which are out of communication range. According to [12], the errors can be up to 60 – 140% of communication range. Finally, localization can be calculated from distance information among the neighbors [13]–[15]. However, the process is not straightforward. Building the global location system with distance information requires the graph to be globally rigid [13], [15]. In fact, it is proven in [14] that even when both distance and angle information are available with a small amount of errors, the localization is still NP-hard. All of the methods are either expensive and not applicable to many networks, or may have a large error range which is not useful in design of coverage algorithms.

The contributions of this paper are as follow.

- First, we present a scheme that estimates the distance between any two neighboring nodes using only local information. Since no localization is performed, there is no need for anchor node or angle information. The estimation scheme assumes random node placement, homogeneous nodes and circular communication and sensing range. Simulation evaluation shows that the estimation error decreases with increasing node density and the mean estimation error approaches zero with sufficiently high node density. The impact of irregular communication is also investigated and simulation results show that while the absolute error increases with irregularity, the error is still relatively low.
- Next, we present a configurable coverage protocol, called CCP, that uses as input, distance between two nodes rather than actual position. Distance information among nodes is much easier to obtain than accurate global location information. CCP allows the trade-off between coverage and node usage. It can be configured to cover at least  $\alpha$  portion of the area with high probability. For complete coverage ( $\alpha = 1$ ), CCP is comparable to OGDC in terms of coverage and number of active nodes required. For 90% coverage, 22% node savings can be achieved.

The paper is organized as follow. In Section II, we present related work and in Section III, the network model and notations used are given. The distance estimation scheme is presented in Section IV, and the coverage protocol in Section V. We conclude in Section VI.

## II. RELATED WORK

Many topology and density control protocols have been proposed in the literature. Some protocols deal only with ensuring connectivity, while others integrate the issues of connectivity and coverage.

GAF [16] divides a region into rectangular grids, and ensures that the maximum distance between any pair of nodes in adjacent grids is within the transmission range of each other. Only the leader in each grid stays awake and relays packets. The leader election scheme in each grid takes into account battery usage at each node. SPAN [17] decides whether a node should be working or sleeping based on connectivity among its neighbors. The objective of both algorithms described is to maintain network connectivity and does not deal with coverage.

[18] defines a sensor coverage metric called surveillance that can be used as a measurement of quality of service provided by a particular sensor network. Centralized optimum algorithms that take polynomial time are proposed to evaluate paths that are best and least

monitored in the sensor network. [19] further investigates the problem of how well a target can be monitored over a time period while it moves along an arbitrary path with an arbitrary velocity in a sensor network. Localized exposure-based coverage and location discovery algorithms are proposed in [20].

In [3], the authors propose an algorithm that ensures complete coverage using the concept of “sponsored area”. Whenever a sensor node receives a packet from one of its working neighbors, it calculates its sponsored area (defined as the maximal sector covered by the neighbor). If the union of all the sponsored areas of a sensor node covers the coverage disk of the node, the node turns itself off. The sponsored area is defined by the nodes that are within sensing range of each other. Thus, the number of active nodes required are much larger than the optimal solutions.

In [1], [4], the authors first prove that when communication range is at least 2 times the sensing range, a completely covered network is also connected. They then propose an algorithm called OGDC (Optimal Geographical Density Control). The sensor nodes decide whether they are or close to the optimal positions and decide whether they should turn on or off themselves distributively. As OGDC performs very close to the optimal solution for very high network density, we used it as a baseline comparison for CCP in the simulation section.

[5] describes a method to determine if an area is  $k$ -covered by checking only the intersection points on the perimeter of a sensing circle. The proposed method is extended to an algorithm that finds the set of nodes that provide  $k$ -coverage.

In [6], the authors analyze the number of random sensing neighbors (nodes within sensing range) required for some confidence of redundancy of the current node, as well as the probability of complete redundancy based on the number of random sensing neighbors. This approach is based purely on random point processes (Poisson Point Process), but it is still based on sponsored area (as in [3]) which may produce inefficient results.

In [7], the authors propose a way to totally eliminate the communication cost of coverage calculation. This is a grid-based approach, and only one node will be awake in each grid, and by doing so, nodes do not need to know the neighboring node information.

Most of the above algorithms and protocols (except [6]) require accurate location information to be stored on each sensor nodes, and they all work on complete coverage rather than a configurable coverage.

### III. ASSUMPTIONS AND NOTATIONS

We assume that the sensor nodes are randomly distributed in a very large 2D region with density  $\lambda$ . Thus, node distribution can be estimated as a Poisson point process. We assume uniform sensing model and uniform communication model, which means all the sensor nodes have same sensing range ( $r_s$ ) and communication range ( $r_c$ ). Events occurring inside sensing range will be detected and vice versa.

Due to obstacles, interference and fading effects, the unit communication model may not be true in practical. We have also evaluated the distance estimation algorithm using a more realistic, irregular communication model based on degree of irregularity (DOI) [10] through simulation in Section IV.

We use capital letter such as  $A$  to represent a region of interest, and  $N_A$  is the random variable for the number of nodes inside region  $A$ .  $n_A$  represents the actual number of nodes inside regions  $A$ . When the context is clear,  $A$  will also be used to represent the content(area) of a region.

### IV. NEIGHBOR NODE DISTANCE ESTIMATION

#### A. Problem Formulation

As explained in the previous sections, estimating the distance between two nodes can be easier and less error prone than global localization information. We propose a simple distance estimation algorithm which can provide enough accuracy for coverage protocol. The basic idea can be explained as follow. In Figure 1, the distance  $d$  between two nodes  $A$  and  $B$  is to be estimated. Let  $A$  and  $B$  be the region of communication circles of node  $A$  and node  $B$  respectively. The common region of region  $A$  and  $B$  is  $X$ . Also, let there be  $n_a$  nodes in  $A$ ,  $n_b$  nodes in  $B$ , and  $n_x$  nodes in  $X$ . Intuitively, when  $d$  is small,  $n_x$  is large and  $n_a$  and  $n_b$  are small. Conversely, when  $d$  is large (A and B within communication range),  $n_x$  is small, while  $n_a$  and  $n_b$  are large. Hence, by taking into account the values of  $n_a$ ,  $n_b$  and  $n_x$ ,  $d$  can be estimated.

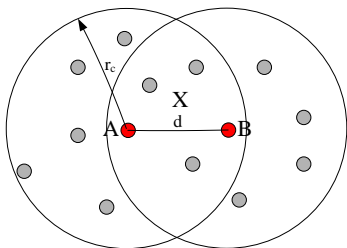


Fig. 1. The number of common neighbors of two nodes can be used to estimate the distance between the two nodes

As  $n_a$ ,  $n_b$  and  $n_x$  are correlated, the problem can be redefined as follow. Given  $(n_a - n_x)$  nodes in  $A - X$ ,  $(n_b - n_x)$  nodes in  $B - X$ , and  $n_x$  nodes in  $X$ , what is the estimated distance  $d$  between node  $A$  and node  $B$ ? In the following analysis, we let  $m$ ,  $n$ , and  $c$  denote  $(n_a - n_x)$ ,  $(n_b - n_x)$ , and  $n_x$  respectively to simplify the expressions.

#### B. Maximum Likelihood Distance Estimation

Maximum likelihood estimation is used to estimate the size of  $X$  and thus the distance  $d$ . The probability of having certain number of nodes inside an area given the value of the area is given in Equation 1.

$$p(N_A = n) = \frac{e^{-\lambda A} (\lambda A)^n}{n!} \quad (1)$$

We need to find the value of  $X$  which maximizes (let  $a = \lambda A = \lambda B$  and  $t = \lambda X$ ),

$$\begin{aligned} M &= p(c|X)p(m|A - X)p(n|B - X) \\ &= \frac{e^{-2a} e^{t^c} (a - t)^{m+n}}{c!m!n!} \end{aligned} \quad (2)$$

Maximizing Equation 2 we get

$$X = \frac{-(m + n + c - a) + \sqrt{(m + n + c - a)^2 + 4ac}}{2\lambda} \quad (3)$$

If  $X < X_{min}$ , we can set  $X = X_{min}$ , and if  $X > X_{max}$ , we can set  $X = X_{max}$ . Where  $X_{min} = (\frac{2\pi}{3} - \frac{\sqrt{3}}{2})r_c^2$ , and  $X_{max} = \pi r_c^2$ .

Results obtained using Equation 3 turn out to be fairly inaccurate when node density is low. This is because the number of nodes within communication range is too small to provide good accuracy, though the accuracy is much better for high node density. The approach taken to improve the estimation accuracy is to increase the number of samples through the use of multiple transmission power levels. By varying the transmission power, the sensor nodes can communicate with different sets of neighbors. This additional information helps to improve the estimation accuracy.

Take an example of two power level sensor nodes, as shown in Figure 2. The two sensor nodes have 2 communication radius  $r_{c1}$  and  $r_{c2}$  ( $r_{c1} \leq r_{c2}$ ), and communication covered areas by the two power levels are  $A_1 = B_1 = \pi r_{c1}^2$ , and  $A_2 = B_2 = \pi r_{c2}^2$ . By adjusting the power levels, there are 4 combination of estimations,  $A_1$  with  $B_1$ ,  $A_2$  with  $B_2$ ,  $A_1$  with  $B_2$ , and lastly  $A_2$  with  $B_1$ . For the case of  $A_1, B_1$  and  $A_2, B_2$ , the estimation is the same as the previous section.

For the two cases with different communication radius, again, the maximum likelihood estimation method can be used.

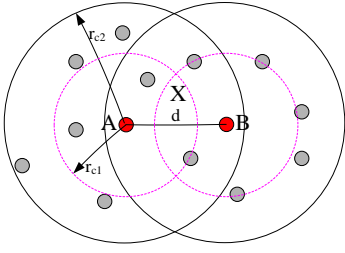


Fig. 2. 2 Power level sensor nodes

$$\begin{aligned} M &= p(c|X)p(m|A_1 - X)p(n|A_2 - X) \\ &= \frac{e^{-\lambda(A_1+A_2)} e^{t^c} (a-t)^m (b-t)^n}{c!m!n!} \end{aligned} \quad (4)$$

Where  $t = \lambda X$ ,  $a = \lambda A_1$ , and  $b = \lambda A_2$ .

The maximization of  $M$  is same to maximize  $\ln M$  and can be simplified to solve the following equation,

$$\begin{aligned} t^3 + (m+n+c-a-b)t^2 + (ab-ac \\ - bc - an - bm)t + abc = 0 \end{aligned} \quad (5)$$

The equation can be easily solved by any approximation algorithms or cubic formula.

The final estimates can be calculated as the average of the estimates on four possible combinations.

### C. Evaluation

A set of simulations are run to evaluate the performance of distance estimation schemes. We compare the performance of using one and two transmission power levels.  $r_c$  is normalized to 1 and the communication ranges with two power levels are 0.5 and 1. The results are show in Figure 3. It can be clearly seen that with low node density, the estimation based on multiple transmission power gives significant improvements on estimation accuracy. The performance of the estimation improves with the increasing node density.

The mean distance estimation error can be reduced to 2% of  $r_c$  for node density larger than 5 using two transmission power levels. At such node density, the 98% percentile values is less than 10% of  $r_c$ . When  $r_c = 2r_s^1$ , the mean distance estimation error is 4% of  $r_s$ . Hence, the distance estimates can provide enough accuracy for coverage problems.

Next, we relax the assumption on the perfect disk communication model. Instead, we adopt the model

<sup>1</sup>This is a meaningful assumption, because  $r_c \geq 2r_s$  is a necessary and sufficient condition for a completely covered network to be connected [1], and meanwhile keeping  $r_c$  as small as possible can reduce energy cost.

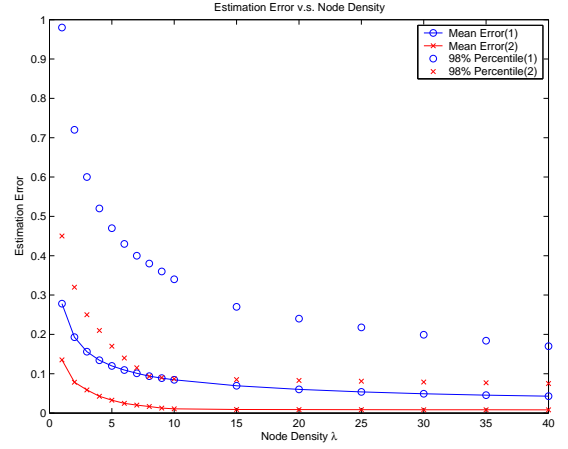


Fig. 3. Distance Estimation Error (98% Percentile and Mean) v.s. Node Density. Single and dual power levels are indicated as (1) and (2) respectively

suggested in [10]. In this model, there is a lower bound and upper bound on signal propagation. Beyond the upper bound, all nodes are out of communication range; and within the lower bound, every node is guaranteed to be within communication range. In between lower and upper bound, degree of irregularity (DOI) is used to denote the irregularity of the radio pattern. DOI is defined as the maximum radio range variation per unit degree change in the direction of radio propagation. Figure 4 shows an example of radio irregularity with the value of DOI 0.05 and 0.2 respectively (DOI of value 0 is the same as the perfect disk model).

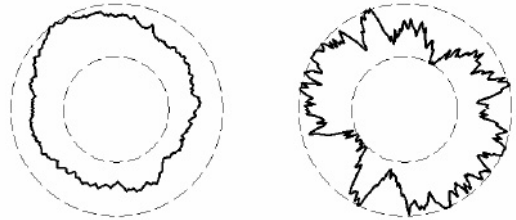


Fig. 4. Radio Pattern Examples with DOI=0.05 and 0.2 respectively

Figure 5 shows how the estimation (two power levels) error varies with DOI (assume upper bound is 1 and lower bound is 0.5 for the first power level, and upper bound is 0.5 and lower bound is 0.25 for the second power level). It can be observed that estimation error increases almost linearly with DOI. With a relatively high (DOI=0.2) irregularity in communication range, with two power level of estimation, the average error can still be tolerable for coverage applications (about 15% of  $r_c$ ).

Finally, in order to execute the estimation algorithm,

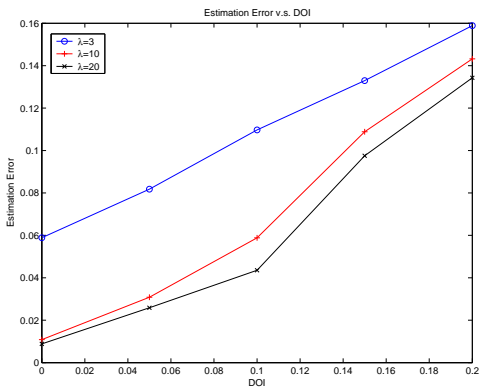


Fig. 5. Mean Distance Estimation Error v.s. DOI

there is still the need to estimate the node density. Our simulation result shows that for a randomly deployed sensor field with sufficiently high node density ( $\lambda \geq 10$ ), the local node density can be approximated to within 10% if the densities of all 1-hop neighbors are averaged. Hence, it is possible to estimate the node density locally even if this information is not available.

## V. CONFIGURABLE COVERAGE PROTOCOL

In this section, we present the configurable coverage protocol. CCP only makes use of the distance information among the neighboring nodes. It can be built on top of distance estimation scheme proposed in Section IV but will also work with any other distance estimation schemes or absolute co-ordinates localization schemes as long as the error is constrained to be within a small portion of  $r_s$ . In emerging wireless technology like Ultra-wide Band or UWB, distance between nodes can be easily measure due to the physical layer properties and the information is available through the MAC layer [21].

CCP allows the users to specify the coverage objective  $\alpha$ . In order to ensure that the coverage objective will be met, we need a way to compute or estimate in a distributed manner (with only distance information), the vacancy of the network. The approach used in CCP is shown in Figure 6. Given a set of active nodes, the area is divided into non-overlapping triangles (without considering boundary effects), and the vertices of these triangles are the active nodes. The basic idea for selecting an additional node to be active is that the ratio of the size of the vacancy or white area ( $V_j$ ) inside the triangle to the area of the triangle ( $T_j$ ) should be less than or equal to  $1 - \alpha$ . For a large WSN, by ensuring that coverage objective is met locally, the global coverage which is computed as  $1 - \frac{\sum V_j}{\sum T_j}$  will be satisfied too.

In CCP, each node distributively elects itself based on the existing edges/triangles that have already been

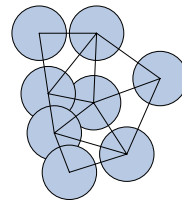


Fig. 6. Illustration of Coverage and Vacancy Estimation

formed and the vacancy values of possible new triangles if it is active. Each node will start a timer based on the vacancy value of the new triangle formed by itself and existing edges, and once a node decides to be active, it will broadcast power on information first and other nodes will implicitly cancel their timers.

Note that in order to ensure the correctness of CCP, it is necessary that active nodes are added one at a time and this is built into the protocol design. By adding only one active node at a time, a unique sequence of (active) nodes addition is obtained. Such a sequence will generate a unique set of triangles formed by adding a new active node to two existing active nodes. This set of unique triangles covers the entire area of interest (excluding boundary effects) and the triangles do not overlap, ensuring that there is no double counting of vacant and covered area.

### A. Vacancy Inside Triangle

While the vacancy may be easily identified graphically or visually, computing the exact values of  $V_j$  using only distance information among nodes is more complicated. Before we formally describe CCP, it is essential to have a look at how the vacancies inside the triangles can be calculated.

1) *Triangle Vacancy Calculation*: Given the distances between each pair of the sensor nodes are  $d_1$ ,  $d_2$  and  $d_3$ , the area of the triangle is  $T(d_1, d_2, d_3) = \sqrt{s(s-d_1)(s-d_2)(s-d_3)}$ , where  $s = \frac{1}{2}(d_1 + d_2 + d_3)$ . The common coverage between any pair of the nodes with distance  $d$ , where  $d < 2r_s$ , is given by  $f(d) = 2r_s^2 \arccos(\frac{d}{2r_s}) - \frac{d}{2} \sqrt{4r_s^2 - d^2}$ . The vacancy of the different cases shown in Figure 7 can be calculated easily. The percentage of vacancy inside the triangle can then be calculated by  $V/T$ . However, for some other cases as will be listed in section V-A.2, the vacancy cannot be calculated easily. We call these exceptional cases. CCP tries to avoid such cases during selection of active nodes.

2) *Exceptional Cases of Vacancy Calculation*: Note that for the cases shown in Figure 7, the sensing nodes are in “good” positions where the angles of the triangle

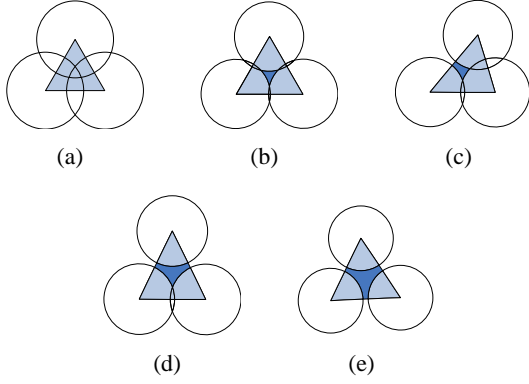


Fig. 7. Triangle Vacancy Calculation. (a)  $V = 0$  (b)  $V = T - \frac{1}{2}\pi r_s^2 + \frac{1}{2}(f(d_1) + f(d_2) + f(d_3))$  (c)  $V = T - \frac{1}{2}\pi r_s^2 + \frac{1}{2}(f(d_1) + f(d_2))$  (d)  $V = T - \frac{1}{2}\pi r_s^2 + \frac{1}{2}f(d_1)$  (e)  $V = T - \frac{1}{2}\pi r_s^2$

are “balanced”. These cases can be easily identified using the distance information and the vacancy inside the triangle calculated in a very simple and standard way. Figure 8 shows the exceptional cases where the simple formula does not apply.

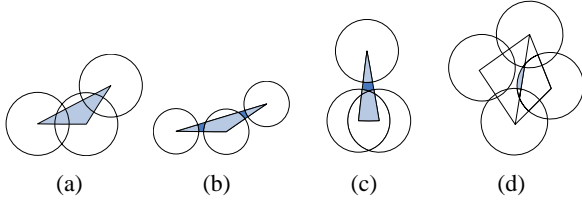


Fig. 8. Exceptional Cases of Triangle Vacancy Calculation

For exceptional cases (a),(b) and (c) shown in Figure 8, the problem comes from the fact that one edge of the triangle crosses all three circles. In addition, it can also be observed that in these cases, the angles inside the triangles are highly imbalance. In cases (a) and (b), one of the angles is very large while in case (c), one of the angles is very small.

In exceptional case (d) shown in Figure 8(d), the vacancy in the left triangle is actually affected by one of the node in right triangle. The vacancy in the left triangle is smaller than the vacancy compute using the calculation stated in the previous section. In this situation, the vacancy is over estimated and the global objective  $\alpha$  can still be satisfied. It can also be observed that case (d) is always linked to case (a) and (b).

All of the exceptional cases are not desirable. In particular, in cases (a) to (c), the vacancy is difficult to compute. In fact, we would further argue that these cases should also be avoided because they potentially increase the number of active nodes that are needed for the same coverage objective. The inefficiency of cases (a) and (b)

can be explained using an example shown in Figure 9(a). Node  $A$  and  $B$  are known active nodes, if node  $C$  decides to be active because the vacancy in triangle  $ABC$  is smaller than the predefined value, then node  $E$  will not be selected based on the edge  $AC$  because there is a very large vacancy in triangle  $EAC$ . A node that is closer to edge  $AC$  has to be elected, which is node  $D$  in this example. On the other hand, as shown in Figure 9(b), if node  $E$  decides to be active based on edge  $AB$ , the final results will be triangle  $ABE$  and  $BCE$ , which is better than the example in Figure 9(a) because the former example tends to have more active nodes than the later one, even though in both cases, the average objective is met. Thus, when both  $E$  and  $C$  hears information about edge  $AB$ ,  $E$  should elect itself first,  $C$  is undesirable.

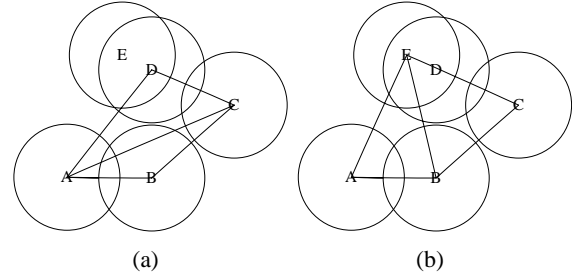


Fig. 9. Illustration of Inefficiency Caused by Exceptional Cases 6 and 7

Case (c) only happens when node  $A$  and  $B$  are too close to each other. For a sufficiently high node density, case (c) is not likely to happen. It is also undesirable because the amount of redundancy is high.

In conclusion, in order to design an efficient distributed algorithm for configurable coverage, the exceptional cases should be avoided because they provide less efficient solutions and the vacancy for these cases are hard to calculate. However, depending on the actual placement, it may not be possible to avoid these cases completely. Nevertheless, for most node density of interest where complete coverage is possible, these cases are rare. Hence, even when these cases are included and no vacancy is assumed (instead of computing the actual vacancy), the error is small.

### B. Node Selection Constraint

As previously mentioned, in CCP, active nodes are added one at a time. In the new node selection process, the set of active sensor nodes must be connected at all times (connectivity constraint) and the exceptional cases analyzed in previous section shall be avoided as much as possible (angle constraint). Thus, during the selection process, nodes that satisfy both connectivity and angle

constraints are considered first. If both constraints cannot be met at the same time, then connectivity constraint naturally takes priority over the angle constraint.

1) *Connectivity Constraints*: CCP tries to elect a subset of sensor nodes that cover  $\alpha$  portion of the environments, it does not consider the connectivity of the network formed by the active sensor nodes. To maintain network connectivity in CCP, a node should only volunteer itself if it is able to communicate with both end vertices of the edge. Thus, each edge of the triangles is connected, and the whole network is then connected.

2) *Angle Constraints*: From observation, the exceptional cases in Figure 8 occur only when there are small (or large) angles inside the triangle. These small or large angles will cause imbalance in the length of edges, and thus may cause the imbalance in vacancies in adjacent triangles. In order to avoid the exceptional cases, small (or large) angles in the triangles should be avoided. Therefore, CCP selects the node not only based on the vacancy values inside the triangle, it also tries to select the triangle that maximizes the minimum angle. Note that this is different from the concept of Delaunay triangulation.

As discussed, the exceptional cases in Figure 8(a) and 8(b) are undesirable. For a dense network, it is better to eliminate all such possibilities to form a triangle of such cases, i.e., the nodes that will form exceptional triangles will not perform any action. As shown in Figure 10(a), considering the connectivity constraint and avoiding the exceptional cases, only the nodes in the shadowed area should compete for the active nodes. The minimum angles formed by the competing nodes and the edge should be  $\beta_1$ . Any node that has an angle smaller than  $\beta_1$  will just ignore the new triangle and edge message.

The value of  $\beta_1$  can be calculated by,

$$\beta_1 = \arcsin\left(\frac{r_s}{d}\right). \quad (6)$$

Note that the value of  $\beta_1$  can be up to  $\frac{\pi}{2}$  when  $d$  is close to 0. Thus, even when network is dense, such constraint shall not be performed when  $d$  is small.

Another constraint in angle is shown in Figure 10(b). When a node decides to become active and form a new triangle, it will broadcast the power on message. All other nodes that are within the communication range of this node will hear this broadcast and try to cancel their timers. It is thus essential for every node that is trying to compete for the new vertex to hear this message. These nodes should be constrained in the shadowed area in Figure 10(b), in which every node is able to directly communicate with other nodes. The shadowed area is limited by angle  $\beta_2$ , which is calculated by,

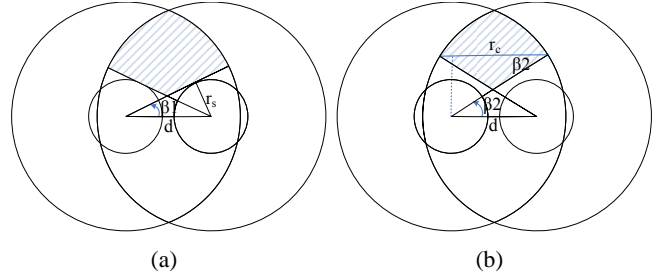


Fig. 10. Angle Constraints

$$\beta_2 = \arccos\left(\frac{r_s + d}{2r_s}\right). \quad (7)$$

Nodes that can form an angle large than  $\max(\beta_1, \beta_2)$  meet the angle constraints and are preferred. For a sufficiently dense network, one or more nodes will be able to meet this angle constraints in most cases.

3) *Rigidity Constraints*: It is possible that given only distance information, the relative position of a node to some of its neighbors cannot be determined (i.e., the local distance graph is not rigid, the node can possibly be on either side of an edge), especially when there are errors in distance estimation. In CCP, any node who cannot form robust quadrilateral<sup>2</sup> to existing triangles will not elect itself as an active node.

It should be mentioned that, the angle constraints in previous section also help in dealing with distance errors because maximizing the minimum angle is able to help the protocol tolerate more distance errors without affecting the robustness of the local distance graph [15].

### C. Protocol Description

In this section, we present the CCP protocol.

1) *Selection of Starting Node*: At the initial selection phase, all nodes are in the “UNDECIDED” state. A node should volunteer to be the starting node with probability  $p$ . The value of  $p$  should be a small value such that it is not likely to have many volunteer starting nodes in each round of selection.

When a node decides to be a starting node, it first waits for a random time  $t_s$  uniformly distributed within  $[0, t_{smax}]$ .  $t_{smax}$  can be any reasonably large values, for example, 20 times the MTU transmission time. This waiting time is used to reduce the probability of having multiple starting nodes but is not crucial for the correctness of CCP. If the node does not hear any messages

<sup>2</sup>A robust quadrilateral is a complete distance graph of 4 vertices, which means the distance information between any pair of the 4 nodes is known.

from neighboring nodes within  $t_s$ , it will change its state to "ON" and broadcast the power on message. If it receives any power on messages from neighbor nodes, it will simply cancel the timer.

2) *First Edge and First Triangle Formation*: After the first starting node broadcasts the power on message, all neighbors around the starting node will set a timer  $t_1$ . If the timer fires, the node will change its state to "ON". The value of  $t_1$  is based on the distance to the starting node  $d$ .

When a node turns "ON", it broadcast power on message together with the edge information. The edge information includes the local unique id of the two end nodes as well as the length of the edge.

Upon receiving the edge information, the neighboring nodes will set a timer  $t_2$ . If the timer fires, the node turns "ON" and form the first triangle. The value of  $t_2$  depends on the vacancy as well as the angles inside the triangle it forms.

The node will broadcast the power on message together with the triangle information. The information includes the id of the three vertices and the length of the three edges. This message also has information about the new edges generated by this triangle (there are normally two new edges). All nodes will save the triangles formed associated with itself (i.e. if a node is a vertex of the triangle, it will save this triangle information). All nodes that hear the triangle information and locate at the same side with the broadcasting node will cancel their timers.

3) *Node Selection Process*: Upon receiving the triangle and new edge message, only those nodes that are located at different side of the new edge with the triangle will perform actions. Each node will first examine whether it has any triangle associated with itself and share a common vertex with the new edge. If there is, it will then look at the edge connecting itself and the common vertex, to see whether the edge has two triangles associated with it. The node will take no action if there are already two triangles associated with this edge. If there is only one triangle associated with the edge, and it satisfies the vacancy requirement, it will announce an creation of a new triangle with only one new edge immediately. This approach always tries to close the region around the common node first.

Otherwise, all other nodes set timer  $t_2$  based on the vacancy and angles to the new edges. The node that fires first turns itself "ON" and announce the existence of a new triangle with two new edges. All nodes that hear the new triangle information will cancel their timer  $t_2$ . Based on the triangle information broadcast by its neighbors, when a node notices that it is within one of the triangles formed, it turns itself "OFF". The protocol terminates

when all nodes are either in the "ON" or "OFF" states.

#### D. Discussion of CCP

1) *Starting Node Probability  $p$* : The value of  $p$  should be small enough so that in the ideal case, only one node in the whole network becomes starting node. This can be a value of say  $\frac{1}{N}$ .

2) *Timer  $t_1$* : The timer  $t_1$  should be based on the distance to the initial startup node. Based on the heuristics used in CCP, the optimal distance should be the edge length of the equilateral triangle which exactly has vacancy of  $1 - \alpha$ .

The value of  $t_1$  is then calculated by  $t_1 = a(d_o - d)$  if  $d < d_o$ , and  $t_1 = a(d - d_o) + c$  if  $d > d_o$ , where  $a, c$  are constants and  $c$  is used to degrade the distances that are larger than optimal (which may cause more vacancy).

3) *Timer  $t_2$* : The value of  $t_2$  can be calculated by the vacancy, as well as the minimum angle. The value of  $t_2$  is computed as  $a|\frac{V}{A} - \alpha| + \frac{b}{\min(a_1, a_2, a_3)} + c$ , where  $a_1, a_2, a_3$  are the angles of the triangle,  $a, b$  and  $c$  are constants.  $c$  is the penalty for the nodes that have vacancy larger than predefined value. It is 0 for the nodes that have vacancy smaller than predefined value.

4) *Joint of Different Sets of Sensor Nodes*: The above protocol description only considers the situation that there is only one starting node. Once there are more than one starting nodes, if there are no special consideration on this, most probably there will be multiple sets of active sensors at the end of the algorithm.

When a node hears broadcast of triangle message from another sets of sensor nodes (differentiated by the id defined by the starting node), it will consider the joining of the new edges associated with itself and the new edges associated with the triangle if any of the possible triangles satisfy the vacancy requirements.

#### E. CCP Performance Evaluation

1) *Simulation Setup*: In all sets of simulations, we normalize the sensing radius  $r_s$  to be 1. The communication range  $r_c$  is set to be 3. The world size is a  $30 \times 30$  square. The communication range is set to 3 times larger than sensing range so that the CCP is able to select the nodes that leave some vacancy. We set  $a = b = 0.5$  for CCP as the weights of vacancy and angle constraint respectively in all simulations.

The relative localization scheme in the simulation assumes that the nodes are able to dynamically change the transmission power levels. Two power levels are used to estimate the distances, one is with  $r_c = 2$  and one is  $r_c = 1$ . Note that the value of  $r_c = 3$  is used for CCP packet transmission, it is not used in distance estimation.



The performance matrix is defined by the average vacancy as well as the number of active nodes to monitor the environments.

2) *Performance of CCP and OGDC*: In the first set of experiments, we compare the performance of CCP and OGDC with both algorithms using the same distance estimate obtained using the scheme described in Section IV. To make CCP comparable to OGDC, we set the coverage objective  $\alpha$  to 1. In addition, we modify OGDC protocol to use distance information rather than position.

The simulation results is shown in Figure 11. It can be observed that CCP with  $\alpha = 1$  has very similar performance to OGDC. Overall, OGDC has a slightly better performance because CCP does not try to minimize node redundancy but simply tries to select the nodes that leave no vacancy and satisfy the angle constraints. However, the performance degradation is small. Using the same distance estimates, the vacancy achieves by OGDC is less than 0.2% lower and the number of nodes needed is reduced by less than 1%.

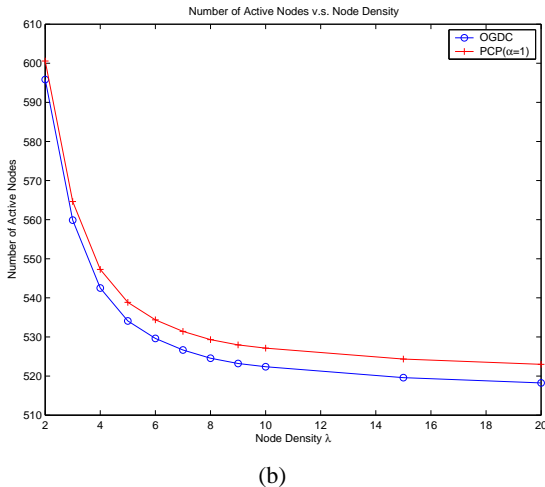
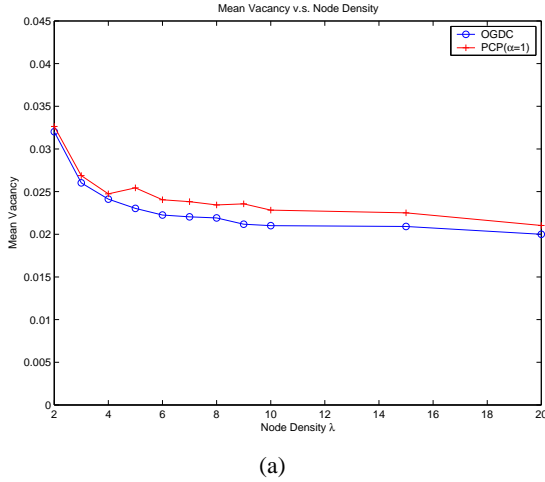
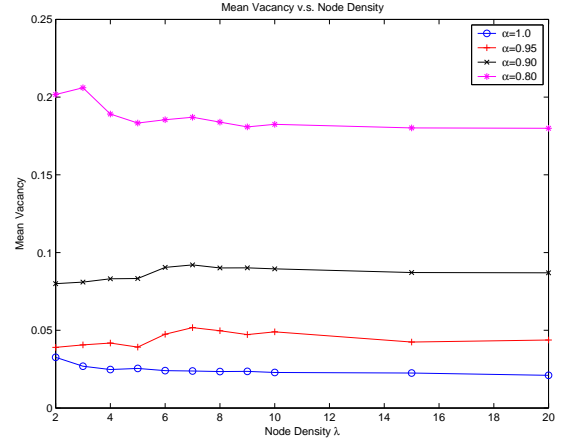
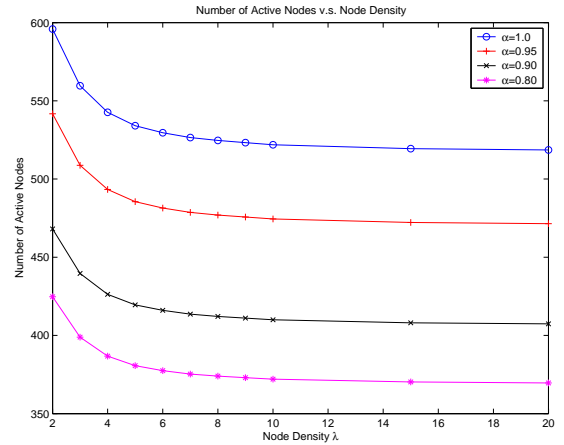


Fig. 11. Comparison between OGDC and CCP



(a)



(b)

Fig. 12. CCP with Coverage Objective  $\alpha = 1, 0.95, 0.9, 0.8$

It is interesting to note that for both OGDC and CCP ( $\alpha = 1$ ), there is always some amount of uncovered area (about 2% – 3%) in the network. The vacancy is a result of the distance estimation error. In addition, when node density is low, the amount of vacancy increases due to insufficient coverage. Therefore, in the presence of location or distance estimation error, it may not be meaningful to demand complete coverage even when network density is high. In our simulation setup, only coverage objective of 0.98 or below can be achieved for both OGDC and CCP.

3) *Performance of CCP with  $\alpha < 1$* : In the second set of experiments, we evaluate the performance of CCP if the objective  $\alpha$  is set to a value less than 1. The results for  $\alpha = 1, 0.95, 0.9$  and  $0.8$  are shown in Figure 12. We have observed that for values of  $\alpha$  between 0.98 and 1.0, there is little difference in terms of average vacancy and number of active nodes needed. As a result, they are not shown in Figure 12.

From Figure 12(a), we can see that CCP is able to

meet the coverage objectives most of the time. There are two reasons why the objective may not be met. First, the network density is too low and there are insufficient nodes. Second, due to distance estimation errors. Nevertheless, it can be observed that even when  $\lambda = 2$  and the distance error is about  $0.1r_c$ , the mean vacancy is still very close to the objective.

In Figure 12(b), when  $\alpha$  is decreased from 1.0 to 0.95, the number of active nodes required is about 91% of the total nodes required when  $\alpha = 1$ . The decrease in nodes required for  $\alpha$  values of 0.9 and 0.8 are 22% and 29% respectively. The results can be explained as follow. When  $\alpha$  is decreased to 95% the savings (9%) is limited by the number of nodes that contribute less than 5% of additional normalized coverage. The biggest savings (12%) comes from moving from 95% to 90% coverage when many more redundant nodes can be found. However, when coverage objective is further decreased to 80%, the amount of redundancy is already low and further savings is only 7%. Further reduction in coverage objective will not be an effective way to reduce nodes required.

## VI. CONCLUSION AND FUTURE WORKS

In this paper we presented a simple distance estimation scheme, based on which the distances among the neighboring nodes can be estimated. The error of the estimation is sufficient for coverage studies. We also proposed a configurable coverage protocol which uses only distances among the neighboring nodes. CCP is able to estimate the vacancies distributively and the global coverage objective  $\alpha$  can be maintained.

Using simulation, we investigated the effects of distance estimation error on coverage density control protocols (OGDC and CCP). CCP performs very similar to OGDC for complete coverage. By relaxing the constraints of complete coverage, CCP is able to generate a subset of sensor nodes which is smaller than the number of nodes required for a complete coverage.

We have identified several interesting problems for possible future work. First, by using the vacancy estimation scheme, we would like to compute the vacancy of any given network efficiently. Second, through local adjustments, nodes should be able to perform local repair whenever there is a local failure without involving global topology change.

## REFERENCES

[1] H. Zhang and J. Hou, "Maintaining coverage and connectivity in large sensor networks," in *International Workshop on Theoretical and Algorithmic Aspects of Sensor, Ad hoc Wireless and Peer-to-Peer Networks*, February 2004.

[2] S. Slijepcevic and M. Potkonjak, "Power efficient organization of wireless sensor networks," in *ICC 2001*, June 2001.

[3] D. Tian and N. D. Georganas, "A coverage-preserving node scheduling scheme for large wireless sensor networks," in *First ACM International Workshop on Wireless Sensor Networks and Applications*, 2002.

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

[5] C.-F. Huang and Y.-C. Tseng, "The coverage problem in a wireless sensor network," in *Proceedings of WSNA'03*, September 2003.

[6] Y. Gao, K. Wu, and F. Li, "Analysis on the redundancy of wireless sensor networks," in *Proceedings of WSNA'03*, September 2003.

[7] R. Iyengar, K. Kar, and S. Banerjee, "Low-coordination topologies for redundancy in sensor networks," in *Proceedings of ACM MobiHoc'05*, May 2005.

[8] "Introduction to the global positioning system for GIS and TRAVERSE," Corvallis Microtechnology, Inc., 1996. [Online]. Available: <http://www.cmtinc.com/gpsbook/index.htm>

[9] N. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system," in *Proceedings of ACM MobiCom'02*, 2002.

[10] T. He, C. Huang, B. Blum, J. Stankovic, and T. Abdelzaher, "Range-free localization schemes for large scale sensor networks," in *Proceedings of ACM MobiCom'03*, 2003.

[11] D. Niculescu and B. Nath, "Ad hoc positioning system (aps)," in *Proceedings of GlobeCom'01*, 2001.

[12] H. A. Oliveira, E. F. Nakamura, A. A. Loureiro, and A. Boukerche, "Error analysis of localization systems for sensor networks," in *Proceedings of the 13th annual ACM international workshop on Geographic information systems*, 2005.

[13] T. Eren, D. Goldenberg, W. Whiteley, Y. R. Yang, A. S. Morse, B. D. O. Anderson, and P. N. Belhumeur, "Rigidity, computation, and randomization in network localization," in *Proceedings of IEEE INFOCOM'04*, 2004.

[14] A. Basu, J. Gao, J. S. B. Mitchell, and G. Sabhnani, "Distributed localization using noisy distance and angle information," in *Proceedings of ACM MobiHoc'06*, 2006.

[15] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robusted distributed network localization with noisy range measurements," in *Proceedings of ACM SenSys'04*, 2004.

[16] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing," in *Proceedings of ACM MobiCom'01*, July 2001.

[17] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An energy-efficient operation in multihop wireless ad hoc networks," in *Proceedings of ACM MobiCom'01*, July 2001.

[18] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava, "Coverage problems in wireless ad-hoc sensor networks," in *Proceedings of InfoCom'01*, July 2001.

[19] S. Meguerdichian, F. Koushanfar, M. Potkonjak, G. Qu, and M. B. Srivastava, "Exposure in wireless ad-hoc sensor networks," in *Proceedings of ACM MobiCom'01*, July 2001.

[20] S. Meguerdichian, S. Slijepcevic, V. Karayan, and M. Potkonjak, "Localized algorithms in wireless ad-hoc networks: location discovery and sensor exposure," in *Proceedings of MobiHoc'01*, 2001.

[21] S. Gezici, Z. Tian, G. B. Biannakis, H. Kobayashi, A. F. Molisch, H. V. Poor, and Z. Sahinoglu, "Localization via ultra-wideband radios," *IEEE Signal Processing Magazine*, 2005.