International Research Journal of Multidisciplinary Science & Technology
www.irjmrs.com ISSN : 2455-930X

A Energy-Efficient Probabilisitic Coverage Protocol in Wireless Sensor Networks

S.Senthil Kumar[1], R.Deepan[2]

[1]*Asst.Professor, Department of Computer Science and Engineering,Coimbatore*
[2]*Professor, Department of Computer Science and Engineering, Coimbatore.*

---***---

Abstract - *Wireless sensor networks are networks for collecting environment information in specific area using sensor nodes with low power. The power consumption of nodes determines the lifetime of the wireless sensor network. Thus, the design of low-power node is very important. The sensors sense physical phenomena in different ways, and thus, are expected to have different sensing models. Even for the same sensor type, the sensing model may need to be changed in different environments. Designing and testing a different coverage protocol for each sensing model is indeed a costly task. To address this challenging task, we propose a new probabilistic coverage protocol (denoted by PCP) that could employ different sensing models. We show that PCP works with the common disk sensing model as well as probabilistic sensing models, with minimal changes. In this paper, the analysis and design of our coverage protocol can be extended to the probabilistic K-coverage case. K-coverage is needed in several sensor network applications to enhance reliability and accuracy of the network.*

Key Words: Sensor networks, coverage in sensor networks, probabilistic coverage, coverage protocols.

1.INTRODUCTION

In Wireless Sensor Networks (WSN) the sensor nodes are usually battery-powered, for which the energy consumption must be minimized in order to extend their working time. In this paper the coverage and connectivity problems in wireless sensor networks have been discussed. Several empirical studies have shown that sensing and communication ranges of sensors are not regular disks. Rather, they follow probabilistic models. Yet, many current coverage and connectivity protocols continue to assume the disk model for ease of analysis, which may lead to incorrect operation of these protocols in real environments. A distributed coverage and connectivity maintenance protocol has been proposed that explicitly accounts for the probabilistic nature of communication and sensing ranges.

Through analytical analysis, it is shown that protocol guarantees a target packet Delivery rate in the network, while ensuring the monitored area is covered with a probability exceeding a given threshold. Using large-scale simulations, the protocol has been compared against others in the literature and showed that it activates fewer nodes, consumes much less energy, and signicantly prolongs the network lifetime. The robustness of the protocol has been demonstrated against random node failures, node location inaccuracy, and imperfect time synchronization. Power consumption is one of the fundamental concerns in wireless sensor networks. Sensors can last for a few weeks using their batteries. The solution is to deploy some extra sensors and distribute the workload between nodes to increase the lifetime. In this case, some protocols are needed to schedule activation and deactivation of nodes while keeping the coverage and connectivity quality. The protocols maintaining the area covered are often referred to as *Coverage* Protocols while Connectivity Protocols guarantee the communication quality between nodes.

The proposed protocol, called Probabilistic Coverage Protocol (PCP), works for the disk sensing model used in many of the previous works in the literature, e.g., [2], [3], [4], [5], [6], [7], [8]. Furthermore, an extensive simulation study of large-scale sensor networks is conducted to rigorously evaluate the protocol and compare it against other deterministic and probabilistic protocols in the literature. The simulation demonstrates that the proposed protocol is robust, and it can function correctly in presence of random node failures, inaccuracies in node locations, and imperfect time synchronization of nodes. The comparisons with other protocols indicate that our protocol outperforms them in several aspects, including number of activated sensors, total energy consumed, and network lifetime. A fully distributed version of the algorithm is designed and implemented.. The Simulation results show that the distributed algorithm converges faster and consumes much less energy than previous algorithms. The rest of the paper is organized as follows: The results & discussion is summarized in Section II. In Section II we formally define the probabilistic coverage problem and show how our protocol can solve it..In Section III, conclusion of this paper is given.

2.PCP WITH DISK SENSING MODEL

In this section, we present our new PCP, in the context of the disk sensing model because it is simpler.

It has been shown before, e.g., in [14], that covering an area with disks of same radius (rs) can optimally be done by placing disks on vertices of a triangular lattice, where the side of the triangle is 3rs. Optimality here means the minimum number of disks required. The idea of PCP is to activate a subset of deployed sensors to construct an approximate triangular lattice on top of the area to be covered. PCP starts by activating any sensor in the area, which we refer to as an activator. This sensor activates six other sensors located at vertices of the hexagon centered at that sensor. Each activated sensor, in turn, activates other sensors at vertices of its own hexagon. As illustrated in Fig. 1, this process continues till the activated sensors form a virtual triangular lattice over the whole area.

We refer to the distance between the vertices of the triangular lattice as the maximum separation between active nodes, and it is denoted by s. The value of s is determined from the sensing range rs of sensors. Under the disk sensing model, the maximum separation is set to ps ¼ 3rs. The lattice is approximate because it is constructed in a distributed manner and is controlled by sensor deployment. The initial sensor deployment is not assumed to be on a lattice; it could be any distribution. In our simulations, we deploy sensors uniformly at random.

Nodes try to form a triangular lattice over the area

We now present some details of the proposed protocol. PCP works in rounds of R seconds each. R is chosen to be much smaller than the average lifetime of sensors. In the beginning of each round, all nodes start running PCP independent of each other. A number of messages will be exchanged between nodes to determine which of them should be on duty (i.e., active) during the current round, and which should sleep till the beginning of the next round. The time it takes the protocol to determine active/sleep nodes is called the convergence time, and it is desired to be as small as possible. After convergence, no node changes its state and no protocol messages are exchanged till the beginning of the next round.

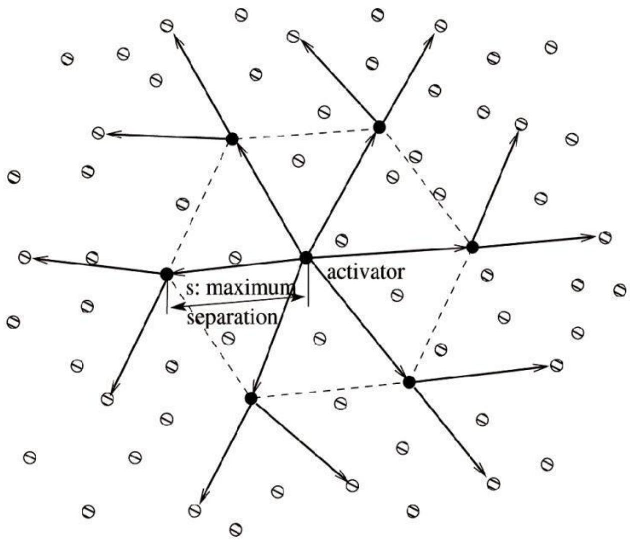

Fig. 1. A simplification of the node activation process in PCP. Activated

In PCP, a node can be in one of the four states: ACTIVE, SLEEP, WAIT, or START. In the beginning of a round, each node sets its state to be START and selects a random start-up timer Ts inversely proportional to its remaining energy level. The node with the smallest Ts will become active and broadcast an activation message to all nodes in its communication range. The sender of activation message is called the activator. The activation message contains the coordinates of the activator, and it tries to activate nodes at vertices of the hexagon centered at the activator, while putting all other nodes within that hexagon to sleep. A node receiving the activation message can determine whether it is a vertex of the hexagon by measuring the distance and angle between itself and the activator. The angle is measured starting from the positive x-axis and going anticlockwise. If the angle is multiple of $_3$=3 and the distance is s, then the node sets its state to ACTIVE and it becomes a new activator. Otherwise, it goes to SLEEP state.

3.METHODOLOGY

Each sensor is modeled as a disk and it is proved that the area is properly k-covered if the perimeter of all disks is k-covered. In this paper, efficient solutions for the problem of selecting the minimum number of sensor nodes from a set of already deployed ones to achieve k-coverage is presented.This paper uses k-coverage algorithm.We can understand the k-coverage algorithm with the help of Figure1.

A. K - coverage algorithm

SENDER

If n is a new message then

Set a delay time t

Store position (x1, y1) of n in COV_LIST

else if n is a duplicate message then.

REPEAT step

while (true) {

weight = 1, totalWeight = n, netSize = 1;
curCoverage = 0, state = TEMP; while
(netSize ≤ n) { After delay time t expire

Set CUR_coverage ← First record of COV_LIST.

while (CURR_coverage != NULL) do

dx = x2 - x1, dy = y2 - y1

Compute distance *d*

weight = atan2 (dx, dy)

if (dx < 0) then dx = (n + 360)

if (netSize × (weight/totalWeight) > rand()) {

state = ACTIVE;

reqCoverage = k − curCoverage;

Pa = reqCoverage/(neighbor Size − curCoverage);

broadcast an ACTIVATE message containing Pa and reqCoverage to neighbors;

}

wait for NOTIFY messages;

/* verify k-Covergae */

if (curCoverage ≥ k) { break; }

/* update variables for next iteration */

if (1/(n − netSize) > rand()) { weight = weight × 2; }

netSize = netSize × 2;

totalWeight = totalWeight + totalWeight/n;

Set CURRENT_COV← First record of CoV_LIST.

if ((CURRENT_COV–>angle = = 0) and

(CURRENT_COV–>angle–>next = = 160) then

}

if (state 6= ACTIVE) { state = SLEEP; }

wait until end of round;

}

RECEIVER

/* upon receiving a message msg */

if (msg.type == ACTIVATE and msg.Pa > rand()) } /* chosen to be active */

/* wait random time to reduce collision */

send a NOTIFY message to msg.source after int rand(0, msg.reqCoverage) × Tm

sec;

state = ACTIVE;

}

update (neighborSize, curCoverage); /* based on msg.source */

4.RESULTS AND DISCUSSION

An efficient approximation algorithm has been proposed which achieves a solution of size within a logarithmic factor of the optimal. A key feature of our algorithm is that it can be implemented in a distributed manner with local information and low message complexity.

- We consider the more general k-coverage (k ≥ 1) problem.
- Where each point should be within the sensing range of k or more sensors.
- Covering each point by multiple sensors is desired for many applications, because it provides redundancy and fault tolerance.

K-coverage is necessary for the proper functioning of other applications, such as

- Intrusion detection, data gathering, and object tracking.
- We start describing our solution approach with the following definition of set systems and hitting sets.

To illustrate, consider an intrusion detection system in military applications, where k-coverage is essential to identify intruding objects of different sizes. A tank, for instance, is detected by many sensors, while a soldier is detected by only a few. A high degree of coverage makes the classification more precise, because of errors in the measurement and the susceptibility of sensors to failure and power shortage.

- Converges much faster than the others.
- Activates near-optimal number of sensors.
- Significantly prolongs (almost doubles) the network lifetime because it consumes much less energy.

Finding the minimum number of nodes to 1-cover the set of all sensor locations is equivalent to finding the minimum dominating set for the graph. The algorithm takes as input the set of sensor locations X, sensing range of sensors r, and required degree of coverage k. If the algorithm succeeds, it will return a subset of nodes to activate in order to ensure k-coverage.Figures.3,4 show the simulation results.

The algorithm may only fail if activating all sensors is not enough for k-coverage because of low density. The minimum required density can be calculated as follows. If every point is to be k-covered, it has to be in the sensing range of at least k sensors. Thus, for each node p there should be at least k other nodes inside a disk of radius r centered at p.

- o In sensor network applications are ensuring area coverage and Maintaining the connectivity of the network
- o selecting the minimum number of sensors to activate to achieve coverage as NP hard problem

  - Random node failure
  - Imperfect time synchronization
  - Location in accuracy

Coverage with various degrees (k-coverage), where every point is sensed by at least k sensors, has also been studied, see the survey in [14]. The problem of verifying k-coverage is studied in [3]. Each sensor is modeled as a disk and it is proved that the area is k-covered if the perimeter of all disks is k-covered. An improved modeling is presented in [10], where the authors use the concept of order-k Voronoi diagrams [13] to build a verifier algorithm. In [12], the authors first propose a k-coverage determination algorithm and then present a distributed sleep control protocol to achieve k-coverage by exchanging several types of messages. In [14], the authors formulate the k-coverage problem of a set of n grid points as an integer linear programming to determine the minimum cost of sensors to cover all grid points.

We have implemented our PCP protocol in NS-2 and in our own packet-level simulator in C++.]. Some results from the NS-2 implementation with reasonable network sizes (up to 1,000 nodes) are presented. Most results, however, are based on our own simulator because it supports much larger networks, which we need to rigorously evaluate our protocol

We use the following parameters in the experiments, unless otherwise specified. We uniformly at random deploy 20,000 sensors over a 1 km $_3$ 1 km area. We use two sensing models: The disk sensing model with a sensing range of rs ¼ 15 m and the exponential sensing model with sensing capacity decay factor $_3$ ¼ 0:05, and we set rs ¼ 15 m as the threshold value below which sensing is achieved with probability 1. We employ the energy model in which is based on the Mote hardware specifications. In this model, the node power consumption in transmission, reception, idle, and sleep modes is 60, 12, 12, and 0.03 mW, respectively. The initial energy of a node is assumed to be 60 Joules, which allows a node to operate for about 5,000 seconds in reception/idle modes.

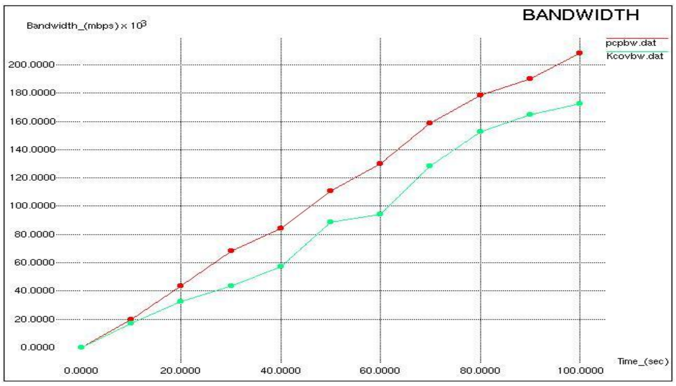

Fig. 2: Bandwidth performance of k-coverage

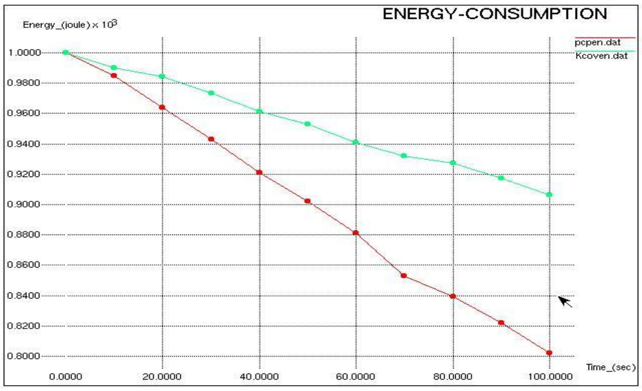

Fig. 3: Enery consumption of k-coverage

5.ENERGY CONSUMPTION:

These measures the energy expended per delivered data packet. It is expressed as

$$\frac{\sum ENERGY\ EXPENDED\ BY\ EACH\ NODE}{TOTAL\ NUMBER\ OF\ PACKETS\ DELIVERED}$$

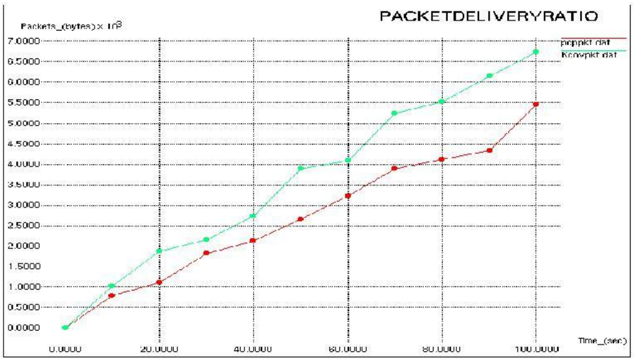

Fig. 4: Packet delivery ratio of k-coverage

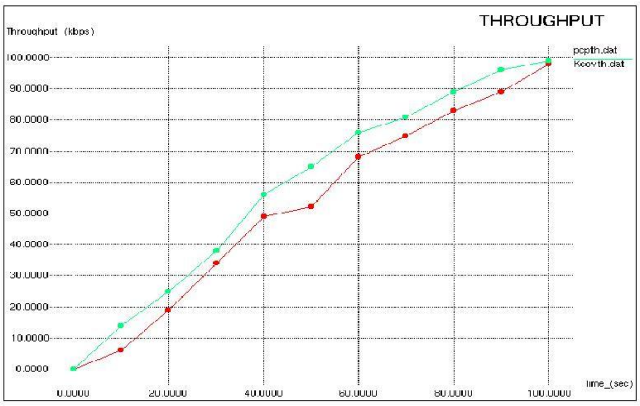

Fig. 5: Throughput performance of k-coverage

6.PERFORMANCE AND EVALUATION:

To evaluate the performance of routing protocols, both qualitative and quantitative metrics are needed. Most of the routing protocols ensure the qualitative

7.PACKET DELIVERY RATIO:

Packet delivery ratio (PDR) measures the percentage of data packets generated by nodes that are successfully delivered, expressed as

$$\frac{TOTAL\ NUMBER\ OF\ PACKETS\ SUCCESFULLY\ DELIVERED}{TOTAL\ NUMBER\ OF\ PACKETS\ SENT} X100\%$$

8.END-END LATENCY:

End-End latency measures the average time it takes to route a data packet from the source node to the hub. it is expressed as

$$\frac{\sum INDIVIDUAL\ DATA\ PACKET\ LATENCY}{TOTAL\ NUMBER\ OF\ PACKETS\ DELIVERED}$$

9.ENERGY CONSUMPTION:

These measures the energy expended per delivered data packet. It is expressed as

$$\frac{\sum ENERGY\ EXPENDED\ BY\ EACH\ NODE}{TOTAL\ NUMBER\ OF\ PACKETS\ DELIVERED}$$

10.THROUGHPUT:

It is defined as the number of packed at destination side at a particular time. It means

$$\frac{NUMBER\ OF\ PACKET\ RECEIVED}{TIME\,(Sec)}$$

11.CONCLUSION

In this paper, a fully distributed, probabilistic coverage protocol has been proposed. A key feature of our protocol is that it can be used with different sensing models, with minimal changes. The protocol has been analyzed and showed that it converges fast and has a small message complexity. The analytical results are verified using simulations. The k-coverage problem is modelled as a set system for which an optimal hitting set corresponds to an optimal solution for k-coverage. An approximation algorithm has been proposed for computing near-optimal hitting sets efficiently. Simulation results show that the distributed algorithm converges faster and consumes much less energy than previous algorithms. The analysis and design of the coverage protocol can be extended to the probabilistic k-coverage case. K-coverage is needed in several sensor network applications to enhance reliability and accuracy of the network. Using probabilistic sensing models in the k-coverage case is expected to yield even higher savings in the number of activated sensors. Another extension is to consider probabilistic communication models, in addition to the probabilistic sensing models, in the design and operation of the protocol. The simulation demonstrates that PCP is robust, and it can function correctly in presence of random node failures, inaccuracies in node locations, and imperfect time synchronization of nodes.

REFERENCES

[1] Mohamed Hefeeda and Hossein Ahmadi, "Energy-Efficient Protocol for Deterministic and Probabilistic Coverage in Sensor Network", IEEE Transaction on Parallel and Distributed Systems, Vol.25, No.5, May 2014.

[2] Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless Sensor Networks: A Survey", Computer Networks, vol. 38, no. 4, pp. 393-422, Mar. 2015.

[3] G. Xing, X. Wang, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated Coverage and Connectivity Configuration for Energy Conservation in Sensor Networks," ACM Trans. Sensor Networks, vol. 1, no. 1, pp. 36-72, Aug. 2010.

[4] Z. Zhou, S. Das, and H. Gupta, "Connected K-Coverage Problem in Sensor Networks," Proc. Int'l Conf. Computer Comm. And Networks (ICCCN '04), pp. 373-378, Oct. 2004.

[5] H. Zhang and J. Hou, "Maintaining Sensing Coverage and Connectivity in Large Sensor Networks," Ad Hoc and Sensor Wireless Networks: An Int'l J., vol. 1, nos. 1/2, pp. 89-123, Jan. 2005.

[6] S. Shakkottai, R. Srikant, and N. Shroff, "Unreliable Sensor Grids: Coverage, Connectivity, and Diameter," Ad Hoc Networks, vol. 3, no. 6, pp. 702-716, Nov. 2005.

[7] X. Bai, S. Kumar, D. Xuan, Z. Yun, and T. Lai, "Deploying Wireless Sensors to Achieve Both Coverage and Connectivity,"Proc. ACM MobiHoc, pp. 131-142, May 2006.Z. Abrams, A. Goel, and S. Plotkin, "Set K-Cover Algorithms for Energy Efficient Monitoring in Wireless Sensor Networks," Proc. Int'l Symp. Information Processing in Sensor Networks (IPSN '04),pp. 424-432, Apr. 2004.

[8] D. Tian and N. Georganas, "Location and Calculation-Free Node Scheduling Schemes in Large Wireless Sensor Networks," Elsevier Ad Hoc Networks, vol. 2, pp. 65-85, 2004.

[9] T. Wu and K. Ssu, "Determining Active Sensor Nodes for Complete Coverage without Location Information," Int'l J. Ad Hoc and Ubiquitous Computing, vol. 1, nos. 1/2, pp. 38-46, 2005.

[10] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. Srivastava,"Coverage Problems in Wireless Ad-Hoc Sensor Networks," Proc.IEEE INFOCOM, pp. 1380-1387, Apr. 2001.

[11] D. Mehta, M. Lopez, and L. Lin, "Optimal Coverage Paths in Ad-Hoc Sensor Networks," Proc. IEEE Int'l Conf. Comm. (ICC '03), May 2003.

[12] X. Li, P. Wan, and O. Frieder, "Coverage in Wireless Ad Hoc Sensor Networks," IEEE Trans. Computers, vol. 52, no. 6, pp. 753-763, June 2003.

[13] L. Lazos and R. Poovendran, "Stochastic Coverage in Heterogeneous Sensor Networks," ACM Trans. Sensor Networks, vol. 2, no. 3, pp. 325-358, Aug. 2006.

[14] Huang and Y. Tseng, "The Coverage Problem in a Wireless Sensor Network," ACM Mobile Networks and Applications(MONET), special issue on wireless sensor networks, vol. 10,no. 4, pp. 519-528, Aug. 2005.