Sensor Deployment and Scheduling in Wireless Sensor Networks

T. Rajath(M. Tech)
Dept. of Electronics and Communication Engineering
Shree Vidyanikathan Engineering College
Tirupati

R. Nagendra (M.Tech,(Ph.D.)
Assisstant Professor (SL)
Dept of Electronics and Communication Engineering
Tirupati

Abstract:
Target coverage is considered as a major problem in case of wireless sensor network. This kind of condition can consequentially reduce the minimum energy consumption. One of the well known techniques for this problem is non disjoint set cover problem. In this paper we are trying to realize this target coverage problem in a wireless sensor network environment. Basically we need to consume the energy in various nodes so that the critical period can be handled. A Euclidean based approach is taken where the concept of non disjoint set cover is used for efficient energy consumption. Then a greedy algorithm based approach is proposed where we introduce a uncovered function to find the best solution among a no of solutions (i.e. the nodes) with an aim to maximize the network lifetime with minimum no of utilized sensor nodes.

Keywords: *Wireless sensor networks, Target coverage, Sensor deployment, K-coverage, Q-coverage*

1. Introduction

From last few years wireless sensor network [1] became an rising trend. It enables the sensor node to combine sensing, processing and communicating capabilities into small low cost sensor devices. Once these nodes get deployed , they self organize to form wireless sensor network (WSN) and communicate via wireless links to perform a specific task of real world [3,6].Availableness of sensor nodes with varieties of sensing capabilities results in hundred of applications including National Security[1,2], Habitat Monitoring[2,7,8],Environment observation and forecasting[2,19],Health Applications[1,2,10],Home and Office Applications[2,11].Therefore WSN's are becoming an practical research field with different activities carried out every year to research and solve different constraints.

Target coverage problem [3] is a major problem which is concerned with the coverage of specific targets by the sensor nodes. These nodes require energy for performing the coverage task. Since, the sensor nodes are usually battery powered; therefore judicious management of energy is an important concern so that coverage task can be performed for a maximum duration.

There are some solutions [3] to handle the problem of target coverage and the researchers are seeking solutions such as:-

1. No. of sensors
2. No. of targets
3. The distance between the sensor i and target k.
4. The distance between the sensor j and target k.

To handle target coverage it is hard to anticipate about the reduce in energy consumption when it is going to reduce, for how long it will last. So supplying of extra resources is not an efficient solution to this problem. This is the reason we need a set cover problem that can statically reduce minimum energy consumption when needed, so that it can perform the coverage task.

When going for handling target coverage problem, we have to face three major problems [3]:

1. Discovering of the sensor nodes and their allocation to the respective targets statically.
2. How to make the target coverage process non disjoint in order to quickly react to the sudden reduce in energy consumption and hence maintain the energy availability even during a critical period.
3. How to overcome the bottle necks in the sensor network infrastructure.

2. Related Work

Target coverage being a major problem and can reduce the minimum energy consumption. To find a solution and then deploying the technique to find the result is very important.

Manual control on this whole process would surely affect the energy consumption so we have to find a set cover problem that can find an solution instantaneously and appropriately.

Karaboga Proposed the artificial bee colony algorithm (ABC) is an optimization algorithm based on the intelligent foraging behaviour of honey bee swarm In the ABC model, the colony consists of three groups of bees: employed bees, onlookers and scouts. It is assumed that there is only one artificial employed bee for each food source. In other words, the number of employed bees in the colony is equal to the number of food sources around the hive. Employed bees go to their food source and come back to hive and dance on this area. The employed bee whose food source has been abandoned becomes a scout and starts to search for finding a new food source. Onlookers watch the dances of employed bees and choose food sources depending on dances.

In the ABC model, the colony consists of three groups of bees: employed bees, onlookers and scouts. It is assumed that there is only one artificial employed bee for each food source. In other words, the number of employed bees in the colony is equal to the number of food sources around the hive. Employed bees go to their food source and come back to hive and dance on this area. The employed bee whose food source has been abandoned becomes a scout and starts to search for finding a new food source. Onlookers watch the dances of employed bees and choose food sources depending on dances. The main steps of the algorithm are given below:

- Initial food sources are produced for all employed bees
- REPEAT
- Each employed bee goes to a food source in her memory and determines a neighbour source, then evaluates its nectar amount and dances in the hive
- Each onlooker watches the dance of employed bees and chooses one of their sources depending on the dances, and then goes to that source. After choosing a neighbour around that, she evaluates its nectar amount.
- Abandoned food sources are determined and are replaced with the new food sources discovered by scouts.
- The best food source found so far is registered.
- UNTIL (requirements are met)

In ABC, a population based algorithm, the position of a food source represents a possible solution to the optimization problem and the nectar amount of a food source corresponds to the quality (fitness) of the associated solution. The number of the employed bees is equal to the number of solutions in the population. At the first step, a randomly distributed initial population (food source positions) is generated. After initialization, the population is subjected to repeat the cycles of the search processes of the employed, onlooker, and scout bees, respectively. An employed bee produces a modification on the source position in her memory and discovers a new food source position. Provided that the nectar amount of the new one is higher than that of the previous source, the bee memorizes the new source position and forgets the old one. Otherwise she keeps the position of the one in her memory. After all employed bees complete the search process, they share the position information of the sources with the onlookers on the dance area. Each onlooker evaluates the nectar information taken from all employed bees and then chooses a food source depending on the nectar amounts of sources. As in the case of the employed bee, she produces a modification on the source position in her memory and checks its nectar amount. Providing that its nectar is higher than that of the previous one, the bee memorizes the new position and forgets the old one. The sources abandoned are determined and new sources are randomly produced to be replaced with the abandoned ones by artificial scouts.

3. Our Proposed Work

3.1 Destination Sequenced Distance Vector Routing Protocol

Destination-Sequenced Distance-Vector Routing (DSDV) is a table-driven routing scheme for ad hoc mobile networks based on the Bellman–Ford algorithm. It was developed by C. Perkins and P.Bhagwat in 1994. The main contribution of the algorithm was to solve the routing loop problem. Each entry in the routing table contains a sequence number, the sequence numbers are generally even if a link is present; else, an odd number is used. The number is generated by the destination, and the emitter needs to send out the next update with this number. Routing information is distributed between nodes by sending *full dump* sinfrequently and smaller incremental updates more frequently.

Packets are transmitted between the stations of the network by using routing tables which are stored at each station of the network. Each routing table, at each of the stations, lists all available destinations, and the number of hops to each. Each route table entry is tagged with a sequence number which is originated by the destination station. To maintain the consistency of routing tables in a dynamically varying topology, each station periodically transmits updates, and transmits updates immediately when significant new information is available. Since we do not assume that the mobile hosts are maintaining any sort of time synchronization, we also make no assumption about the phase relationship of the update periods between the mobile hosts. These packets indicate which stations are accessible from each station and the number of hops necessary to reach these accessible stat ions, as is often done in dist ante-vector routing algorithms.

Routing information is advertised by broadcasting or multicasting the packets which are transmitted periodically and incrementally as topological changes are detected – for instance, when stations move within the network. Data is also kept about the length of time between arrival of the jkt and the arrival of the best route for each particular destination. Based on this data, a decision may be made to delay advertising routes which are about to change soon, thus damping fluctuations of the route tables. The advertisement of routes which may not have stabilized yet is delayed in order to reduce the number of rebroadcasts of possible route entries that normally arrive with the same sequence number.

3.1.1 Example of DSDV in operation

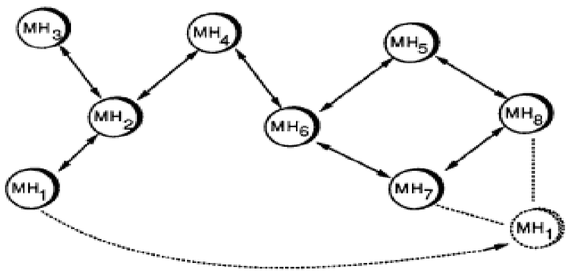

Figure 1: Movement in an adhoc network

Consider MH4 in Figure 1.Table 1 shows a possible structure of the forwarding table which is maintained at MH4. Suppose the address of each Mobile Host is represented as MHi Suppose further that all sequence numbers are denoted SNNN_MHi, where MHi specifies the computer that created the sequence number and SNNN is a sequence number value. Also suppose that there are entries for all other Mobile Hosts, with sequence numbers SNNN.MHi, before MH1 moves away from MH2. The install time field helps determine when to delete stale routes. With our protocol, the deletion of stale routes should rarely occur, since the detection of link breakages should propagate through the ad-hoc network immediately. Nevertheless, we expect to continue to monitor for the existence of stale routes and take appropriate action.

Destination	NextHop	Metric	Sequence number	Install	Flags	Stable_data
MH_1	MH_2	2	S406_MH_1	T001_MH_4		Ptr1_MH_1
MH_2	MH_2	1	S128_MH_2	T001_MH_4		Ptr1_MH_2
MH_3	MH_2	2	S564_MH_3	T001_MH_4		Ptr1_MH_3
MH_4	MH_4	0	S710_MH_4	T001_MH_4		Ptr1_MH_4
MH_5	MH_6	2	S392_MH_5	T002_MH_4		Ptr1_MH_5
MH_6	MH_6	1	S076_MH_6	T001_MH_4		Ptr1_MH_6
MH_7	MH_6	2	S128_MH_7	T002_MH_4		Ptr1_MH_7
MH_8	MH_6	3	S050_MH_8	T002_MH_4		Ptr1_MH_8

Table 1 Structure of the MH4 forwarding table

From table 1, one could surmise, for instance, that all the computers became available to MH4 at about the same time, since its install-time for most of them is about the same. One could also surmise that none of the links between the computers were broken, because all of the sequence number fields have times with even digits in the units place. Ptrl-MHi would all be pointers to null structures, because there are not any routes in Figure 1 which are likely to be superseded or compete with other possible routes to any particular destination.

Destination	Metric	Sequence number
MH_1	2	S406_MH_1
MH_2	1	S128_MH_2
MH_3	2	S564_MH_3
MH_4	0	S710_MH_4
MH_5	2	S392_MH_5
MH_6	1	S076_MH_6
MH_7	2	S128_MH_7
MH_8	3	S050_MH_8

Table 2: Advertised route table by MH4

Now suppose that MH1 moves into the general vicinity of MH5 and MH7, and away from the others (especially MH2). The new internal forwarding tables at MH4 might then appear as shown in table 3.

Destination	NextHop	Metric	Sequence number	Install	Flags	Stable_data
MH_1	MH_6	3	S516_MH_1	T810_MH_4	M	Ptr1_MH_1
MH_2	MH_2	1	S238_MH_2	T001_MH_4		Ptr1_MH_2
MH_3	MH_2	2	S674_MH_3	T001_MH_4		Ptr1_MH_3
MH_4	MH_4	0	S820_MH_4	T001_MH_4		Ptr1_MH_4
MH_5	MH_6	2	S502_MH_5	T002_MH_4		Ptr1_MH_5
MH_6	MH_6	1	S186_MH_6	T001_MH_4		Ptr1_MH_6
MH_7	MH_6	2	S238_MH_7	T002_MH_4		Ptr1_MH_7
MH_8	MH_6	3	S160_MH_8	T002_MH_4		Ptr1_MH_8

Table 3 MH4 forwarding table (updated)

Only the entry for MH1 shows a new metric, but in the intervening time, many new sequence number entries have been received. The first entry thus must be advertised in subsequent incremental routing information updates until the next full dump occurs. When MH1 moved into the vicinity of MH5 and MH7, it triggered an immediate incremental routing information update which was then broadcast to MH6. MH6, having, determined that significant new routing information had been received, also triggered an immediate update which carried along the new routing information for MH1. MH4, upon receiving this information, would then broadcast it at every interval until the next full routing information dump. At MH4, the incremental advertised routing update would have the form as shown in table 4.

Destination	Metric	Sequence number
MH_4	0	S820_MH_4
MH_1	3	S516_MH_1
MH_2	1	S238_MH_2
MH_3	2	S674_MH_3
MH_5	2	S502_MH_5
MH_6	1	S186_MH_6
MH_7	2	S238_MH_7
MH_8	3	S160_MH_8

Table 4 MH4 advertised table (updated)

In this advertisement, the information for MH4 comes first, since it is doing the advertisement. The information for MH1 comes next, not because it has a lower address, but because MH1 k the only one which has any significant route changes affecting it. As a general rule, routes with changed metrics are first included in each incremental packet. The remaining space is used to include those routes whose sequence numbers have changed.

In this example, one node has changed its routing information, since it is in a new location. All nodes have transmitted new sequence numbers recently. If there were too many updated sequence numbers to fit in a single packet, only the ones which fit would be transmitted. These would be selected with a view to fairly transmitting them in their turn over several incremental update intervals. There is no such required format for the transmission of full routing information packets. As many packets are used as are needed, and all available information is transmitted. The frequency of transmitting full updates would be reduced if the volume of data began to consume a significant fraction of the available capacity of the medium.

4. Conclusion and Future Work

In this paper , at first we have defined the problem of target coverage in a sensor network community and have analyzed the probable solutions to this problem. A simple overview of non disjoint set cover problem showed an efficient way to solve the problem.

For realization of target coverage in a wireless sensor network we introduced two approaches: one is Euclidean based approach and another one is the Greedy algorithm based. The Euclidean based approach describes how we can implement the concept of Euclidean for target coverage problem. In the Greedy algorithm based approach, the process is executed through the following steps, by defining a uncovered function to evaluate the best solution.

Our future work involves the implementation of both the approaches in a wireless sensor network environment. The Euclidean based approach will be compared with the Greedy Algorithm based approach and a comparison will be done. In my next paper a simulation study will be performed for both the approaches considering the parameters –uncovered node, remaining battery life of a sensor, the sensor target distance and the no. of target/sensor.

References

1. Alan Mainwaring, Joseph Polastre, Robert Szewczyk, David Culler, and John Anderson. "Wireless sensor networks for habitat monitoring." In ACM International Workshop on Wireless Sensor Networks and Applications (WSNA'02), Atlanta, GA, September 2002.
2. Cerpa, J. Elson, D. Estrin, L. Girod, M Hamilton, and J. Zhao. "Habitat monitoring: Application driver for wireless communications technology". In Proceedings of the 2001ACM SIGCOMM Workshop on Data Communications in Latin America and the Caribbean, April 2001.
3. Edoardo Biagioni and Kent Bridges. "The application of remote sensor technology to assist the recovery of rare and endangered species". In Special issue on Distributed Sensor Networks for the International Journal of High Performance Computing Applications, Vol. 16, N. 3, August 2002.
4. I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, "Wireless Sensor Networks: A Survey", Elsevier Computer Networks, vol.38, no.4, pages 393-422, Mar. 2002.
5. Loren Schwiebert, Sandeep K. S. Gupta, and Jennifer Weinmann. "Research challenges in wireless networks of biomedical sensors". In Mobile Computing and Networking, pages151-165, 2001.
6. M. Cardei, M. T. Thai, Yingshu Li and Weili Wu, "Energy-Efficient Target Coverage in Wireless Sensor Networks,". 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2005), pp: 1976-1984, vol. 3, 13-17 March 2005.
7. Mani B. Srivastava, Richard R. Muntz, and Miodrag Potkonjak. "Smart kindergarten: sensorbased wireless networks for smart developmental problem-solving environments." In Mobile Computing and Networking, pages 132-138, 2001.
8. PurnimaKhuntia and Prasant Kumar Pattnaik "Some Target Coverage Issues of Wireless Sensor Network" International Journal of Instrumentation, Control & Automation (IJICA), Volume 1, Issue 1, 2011
9. PurnimaKhuntia and Prasant Kumar Pattnaik "Target Coverage Management Protocol for Wireless Sensor Network" Journal of Theoretical and Applied Information Technology, 15th January 2012. Vol. 35 No.1
10. Sanjaya Kumar Padhi and Prasant Kumar Pattnaik, "A Novel Distributed Protocol For Randomly Deployed Clustered Based Wireless Sensor Network" Journal of Theoretical and Applied Information Technology, Vol 15. No. 1, 2010.
11. Sung-YeopPyun and Dong-Ho Cho" Power-Saving Scheduling for Multiple-Target Coverage in Wireless Sensor Networks" IEEE Communications Letters, Vol. 13, NO. 2, FEBRUARY 2009