Term Paper on

Distributed Algorithms for Wireless Sensor Network– A Survey

As a part of

Course IT630

By

Nirav S. Uchat

06305906

Under guidance of

Prof. Umesh Bellur – IIT Bombay

Computer Science and Engineering Department

IIT Bombay

Abstract

Wireless sensor network is a wireless network consisting of independent sensor, communicating with each other in distributed fashion to monitor the environment. Sensors are usually attached to microcontroller and are powered by battery. Application domain of WSN varies from habitat monitoring to monitor volcanic eruption. WSN are usually deployed in very harsh environment where node failure occurs frequently. The network should be resilient to node failure and should give long life which is of the order of 1-2 year with single charge. In this paper we will see different distributed algorithm used for neighborhood discovery, for ensuring converge and connectivity. We will also look at low energy clustering algorithm which is used to extend lifetime of WSN. It gives performance measure of these algorithm and their advantages. It also lists overhead associated with it. The goal of WSN is to have long life time and high reliability with maximum coverage.

Introduction

The relentless pace of technology growth has led to the emergence of variety of sensors and networked sensor platform. Today network sensors span the spectrum of cost, form-factor, resolution and functionality. TinyOS (Embedded Linux OS) can be implemented on small hardware and has almost same functionality as that of PC. Sensor network application spans across many domains including military application such as object tracking, sniper detection, hebetate monitoring etc. With criticality of application, the algorithm running on the system must be robust. Sensor node runes on battery power and node failure are common because of working environment. To counter this problem application uses distributed algorithm having fault tolerant mechanism. This survey paper discusses distributed algorithm for coverage and connectivity, for topology discovery and cluster based low power algorithm to reduce energy consumption. There are routing algorithm such as XMAC[1] (1) and BMAC which uses strobe preamble technique to reduce node waiting time. There are algorithms for localization which uses GPS for localizing sensor node. Sensor network may contain non-homogeneous hardware; running same distributed algorithm on it can be difficult but not impossible. We start with coverage and connectivity algorithm then will move to topology discovery (TopDisc) and will conclude with Cluster based low power algorithm - LEACH.

[1] XMAC performs better than BMAC

Integrated Coverage and Connectivity Algorithm

Energy is of paramount importance in wireless sensor network that operate on battery for long period. An effective approach for energy conservation can be scheduling sleep interval for non active node, while other node remains active to provide complete coverage of region under consideration. The network must be capable of configuring itself to required degree of coverage (k-covered)[2] and connectivity[3].

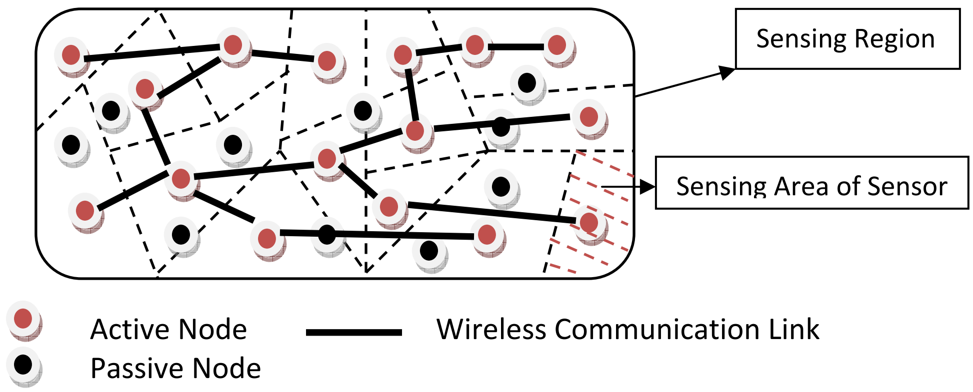

Figure 1 : Coverage and Connectivity Map

There are various algorithms that consider coverage and connectivity as isolated problem. But study shows that both are inherently interdependent. In this section we are going to discuss coverage configuration protocol (CCP) which ensures coverage and can dynamically adjust coverage degree, which is not possible in other protocols. Later we will combine CCP with SPAN[4] protocol to ensure both connectivity and coverage. This is done because CCP can't guarantee connectivity where as SPAN does.

There is interesting relation between communication & sensing range of sensor, which says that, to ensure coverage and connectivity, $R_c > 2R_s$; where R_c is communication range and R_s is sensing range of a sensor. We will see effect of this relation during actual simulation. We will consider both case were $R_c > 2R_s$ & $R_c < 2R_s$.

CCP is a decentralized algorithm and it depends on local state of neighbor. This feature allows CCP to scale to large sensing application in which node can fail at runtime. Node failure is handled by dynamically changing coverage degree. We will now look at CCP algorithm which is also known as *K_s-Coverage Eligibility Algorithm*.

---

[2] Each point in region is in sensing area of K sensor, if one sensor fails other can still sense that region, which provide fault tolerance

[3] All active node can communicate with each other in single or multi hop, so that sensed data can be send back to base station for processing

[4] SPAN protocol ensures connectivity, not coverage

The goal is to ensure every point in sensing region is covered by K sensor. To achieve this, each node runs CCP algorithm independently and decide whether it is eligible to become active.

K_s-Coverage Eligibility Algorithm

Each node runs Eligibility algorithm for assuring K_s coverage. Rather than considering every point for coverage only intersection points are considered. Detail proof can be obtained from (2)

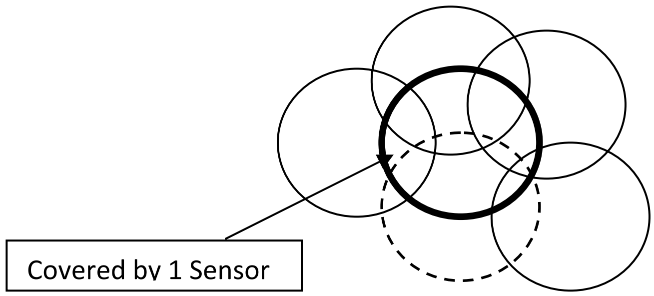

Figure 2 : K_s Eligibility Algorithm

In above diagram, dark circle sensor is ineligible for Ks=1, because all point in that region are covered by at least one active sensor. But for Ks > 1, it will become active. If sensor (with dashed circle) fails then dark circle sensor will become active to fulfill coverage criteria.

Each node maintains information of its sensing neighbor in *SN(v)* to find intersection point. The node is ineligible to become active if

a) There is no intersection point inside sensing circle of sensor v
b) There are K_s or more sensor located at position of v

Node sends its neighbor information in Hello Packet and may include up to two hop neighbor.

State Transition Diagram

A node in CCP can be in one of three states: SLEEP, ACTIVE or LISTEN

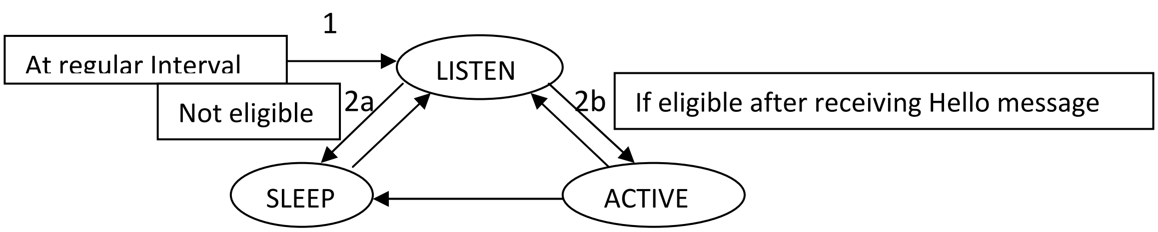

Figure 3 : State Diagram of K_s Eligibility Algorithm

SLEEP state: when T_s timer expires, node goes into LISTEN state

LISTEN state: It receives Hello message and runs eligibility algorithm, if it is eligible it starts T_j (Join timer), otherwise returns to the sleep state.

ACTIVE state: On receiving HELLO message in active state, if it finds it become ineligible it starts T_w (withdraw timer); if T_w expires it goes in to sleep state and start sleep timer.

CCP efficiency heavily depends on timer value; it is interesting to note that

a) *Join timer* value should be small for sensor that is responsible for covering large area
b) Both timer value are selected randomly, hence performance of CCP is highly randomized depending on application and timer value.
c) It uses heuristic technique to determine T_j & T_w

Integrating SPAN with CCP

Study shows that CCP is incapable of assuring connectivity. On other hand SPAN does. So we can integrate eligibility criteria SPAN with CCP to ensure both connectivity and coverage. We can achieve this by modifying eligibility criteria of CCP as

"Node becomes active if at least one pair of its neighbor cannot reach each other either directly or via one or two active node". That is to say that if node can't communicate with one of its neighbor via one or two node then that node becomes active.

Effect of Inequality $R_c > 2R_s$

When $R_c > 2R_s$, the hello message from each node contains its own location. When $R_c < 2R_s$, a node may not be aware of all sensing (active) node, and hence activate itself in spite some other node already sensing the same region, in such case number of active node may increase resulting in high energy usage.

Performance of Coverage Configuration protocol (2)

As stated earlier, CCP gives good performance when $R_c > 2R_s$. Author has done extensive simulation and it compares performance of CCP with other protocol.

1. Experiment shows that average coverage degree is near to 2 where as using Ottawa protocol it is around 6 to 7. This implies that CCP has small number of active node compare to Ottawa protocol[5]. This reduces energy consumption.
2. It is found that, CCP contains small number of nodes with high coverage degree. More precisely in 100 nodes deployed in 50mx50m with each patch of size 1mx1m, only one node was having coverage degree 4. In general it is found that coverage degree is around 2 to 3 when we desire to get $K_s = 1$.
3. In case of coverage percentage CCP always gives full coverage while in SPAN it decreases with increase in R_c/R_s ratio. Since node will be able to communicate with more node as ration increase this in turn makes some node to become inactive and hence it reduces coverage area.

[5] It is decentralized algorithm design to preserve coverage while turning off redundant node

As performance matrix shows that CCP+SPAN gives better result when Rc/Rs > 2, but it would be been more convincing if author gave timer value used in ineligibility algorithm. As stated earlier this values plays important role in deciding number of active node. Also the results discussed are purely application dependent and May not give same performance for other application.

Topology Discovery in Wireless Sensor Network

In this section we will look at topology discovery algorithm (TopDisc) (3), which is used for data management. TopDisc algorithm represents a logical organization of the node and provides a framework for managing sensor network. The main purpose of the topology discovery process is to provide network administrator with the network topology. Further it can be used to create connectivity and reachability map.

Our aim is to construct topology of the whole network from the perspective of a single node. The steps are

1. Node requiring the topology of the network initiates a "topology discovery request"
2. This request is received by all node in the network
3. All node response to this request with topology information

There are three way in which this information is converge back to inquiring node,

1. ***Direct Response:*** where each node replies to its parent. Some mechanism is needed for sending neighborhood information. This approach gives complete picture of network topology. But it has high communication overhead.

2. ***Aggregated Response:*** all nodes send topology request but wait for the children node to respond before sending reply to its own parent. Node can find its neighborhood information by listening to communication channel. Children should first respond to its parent and then parent can send this information to its own parent.

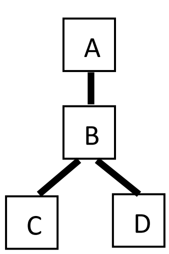

Figure 4 : Network Layout

a. Node C & D forward request, node B listen to it and deduce both as its children.
b. Node C & D reply back to node B
c. Node B aggregate this response and send it to node A
d. Node A get complete topology information

3. ***Cluster Response:*** in this network is divided into set of clusters with each cluster having cluster head. Each node is a member of at least one cluster. Cluster head receive responses from its cluster members and forward it to higher level. Cluster response gives reachability map that is to say that if all cluster head are reachable the all node are reachable form at lest one cluster head.

TopDisc algorithm uses cluster response method for topology discovery. Our goal is to find minimal number of cluster head which cover entire region. This paper discusses two approaches for TopDisc algorithm which are *three color algorithm* and *four color algorithm*. Goal is to find minimal set cover in distributed fashion.

Three Color Algorithm:

To find cluster head, this algorithm assign three color to each node as it proceeds

- White: Undiscovered Node, node which hasn't received any topology discovery request
- Black: Cluster head node, which reply with its neighborhood set to topology discovery request
- Gray: node which is covered by at least one black node, i.e. it is neighbor of a black node

Working of Algorithm:

- Node initiating topology discovery request is assigned color black and broadcast the request
- All white nodes receiving this request turn gray.
- Each gray node forward request to its neighbor with delay inversely proportional to its distance[6] from black node from which it received the request.
- When white node receive request from gray node it turn black with some random delay. During this time if white node receives any request from black node then it turns gray.
- Once the node are gray or black they ignore topology discovery request

Four Color Algorithm:

This algorithm uses same logic as that of three colors. It differs in color assignment i.e. when white nose receive topology discovery packet it turn dark grey rather then turning black. This is to achieve less number of black nodes. This algorithm gives non-overlapping isolated cluster with less number of black nodes.

---

[6] This heuristic gives nearly same result as that of greedy set cover problem.

TopDisc Response Mechanism:

In first phase we assign color to each node and form the cluster, we have following information in our hand at the end of first phase

- Black node is cluster head
- Grey node knows its cluster ID
- Each node knows its parent black node
- Each black node knows to which black node it should forward the information
- All node have their neighborhood information

Using this information we move to topology learning phase

- All nodes send data to its parent black node. Parent starts timer and wait for response from its children.
- The timer value of children should be less than that of parent.
- Parent receives data from its children and sends aggregated response to its own parent.

Performance of TopDisc Algorithm

Author has done extensive simulation on NS-2 simulator (4). Results shows there is more number of black node in TopDisc then centralized algorithm. But this can be tolerable rather then running centralized algorithm. It also shows that as communication range increases number of black node decreases. But since simulation is done on software, the performance may vary during actual implementation.

Shortcomings of this algorithm

The author compares algorithm complexity with greedy solution to set cover problem (log n), but greedy solution to NP-complete problem will not always give good results. It would be interesting if one can prove upper bound on this algorithm which gives satisfactory result for all type of network configuration. Author has use "delay inversely proportional to its distance from black node" heuristic for deciding node becoming black, while proving that it assumes each sensor has circular sensing region which is not true. There are papers on radio irregularity (5) (6), which list all mistaken axioms and radio irregularity in radio communication.

One of the drawbacks of this algorithm is cluster head must be energy rich since it has to manage all types of communication in the network, and if cluster head fail then whole process should be repeated, which is energy consuming. The timer value is manually selected and this should be done very precisely. As said earlier, there is no in built failure handling mechanism in this algorithm.

Extension to this algorithm
One can extend this idea and can design the framework for sensor network management Protocol (sNMP).

LEACH - Low Energy Adaptive Clustering Hierarchy (7)

In last protocol we saw that there was no randomness in the cluster head selection, which resulted in cluster head to become bottleneck in the algorithm. Here we will discuss algorithm which uses dynamic selection of cluster head at run time to distribute load evenly on all the nodes. It uses priory information of node to make decision to make itself as cluster head. Once the cluster head is selected and *local cluster*[7] is formed, data fusion and compression can be used to send aggregated response to base station.

LEACH Algorithm:
This algorithm works in two phase, first is cluster selection phase and second is steady-state were data is send to base station using compression and aggregation technique.

Cluster head Selection:
In cluster head selection, node uses priory information to decide whether or not to become cluster head for current round. This is local decision and it depends on prior history of node under consideration.

$$T(n) = \begin{cases} \frac{P}{1-P*\left(r \bmod \frac{1}{P}\right)} & \text{if } n \in G \text{ else } 0 \end{cases}$$

Where p is the desired percentage of cluster head (0.05) in the network, r is current round and G is set of nodes which have not been cluster head in the last $\frac{1}{P}$ rounds. If value of T(n) is greater then some predefined threshold then that node becomes cluster head.

Now each cluster head will broadcast message using CSMA MAC protocol and all non-cluster head will listen to this message and will select one node as cluster head whose signal strength is

[7] Non-cluster head will listen to the signal strength from of broadcasted message of all cluster head and will select node which has highest signal strength as its cluster head

maximum, since that node will require less energy to transmit its data to cluster head. Thus using this technique cluster is formed and second phase commence.

One of the main feature of this algorithm is it uses TDMA for scheduling non-cluster node to send data to cluster head. So that node will turn off its radio while other node is schedule for transmission. This reduces energy consumption and intern extends node life time.

Transmission phase:

In this phase node uses TDMA to send data and may use compression to reduce energy consumption. Study shows that "*communication requires more energy then computation*". So it's better to do computation and reduce the data volume before transmission.

Use of TDMA do reduce energy usage but consider a situation where you have deployed intrusion detection using camera sensor and intruder is sensed at some sensor, it may be case that at the time of intrusion that sensor is not schedule for data transmission, in such case that event will not be sensed at all by base station or may be with delay of some second (we are using highly resource constrained device e.g. Telos (8)). One has to come up with dynamic scheduling algorithm depending on real time requirement.

Performance of LEACH:

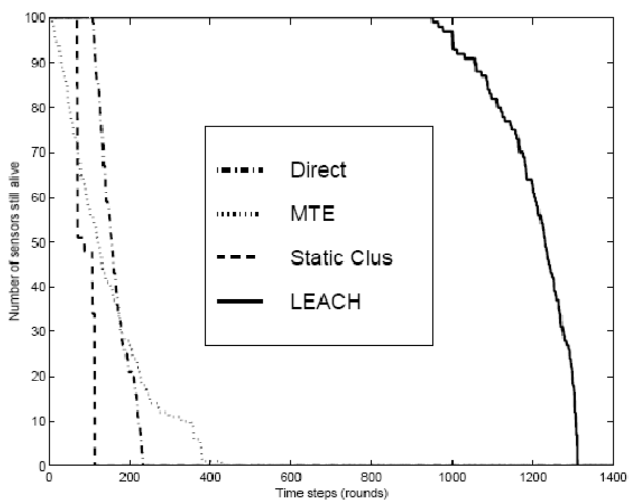

Figure 5 : Active Nodes in LEACH with increasing time

- MTE is minimum-transmitting-energy algorithm where node sends data to other node where it requires lest transmitting energy
- Node in LEACH last longer then other algorithm
- Other results shows that selecting 5% of node as cluster head gives excellent result in terms of energy consumption
- It performs better than static algorithm

Shortcoming of LEACH:

- Cluster head selection phase node is not considering other factor in to account while deciding its eligibility as cluster head.
- It uses only priory information of that node

- If cluster head fails then there is no mechanism for other node to select other cluster until next phase of cluster head selection comes
- In this algorithm cluster head will become bottleneck and one should take care while considering number of possible cluster head, if this value is not selected properly then network will die out early due to cascading effect.
- There is randomness in selection of cluster head which may result in some node getting repeatedly selected; to counter this problem adaptive back-off technique can be used (9).
- As point out earlier, TDMA scheduling will not be adequate.
- Author has done all simulation in MATLEB; performance may vary during actual implementation
- Value "5% node as cluster head" is application dependent. Performance may vary depending on deployed application.

Extension to LEACH:

- More efficient cluster head selection method can be employed, such as getting information from neighbor and then deciding whether to become cluster head or not. This will give better cluster creation in network.
- Rather then sending data directly to base station from cluster head, routing can be use to balance the load across several mode. Still this is subject to extensive simulation.
- Fault tolerance mechanism can be integrated for cluster head failure.

Conclusion

In this survey paper we looked at three distributed algorithms

- Coverage and Connectivity
- Topology Discovery
- LEACH – Cluster based Low Power Algorithm

We also discuss shortcoming of each algorithm and possible extension to it. As nodes in Sensor network are resource constrained device, distributed algorithm must be light on computation and it should be design with low power usage. Results show that distributed algorithm is resilient to failure then corresponding centralized algorithm. Implementing all three algorithms in a single application may not be feasible, selective algorithm can be used depending on requirement. All algorithm uses wireless transmission for communication which is low power radio 802.15.4 (10) and has low data rate compared to 802.11 but consume very less power. There is tradeoff between reliability and system life time, higher reliability implies more number of active nodes at a time which increases power requirement and decreases system life time.

Before deciding which distributed algorithm to uses for sensor network designer has to first go through WSN initialization task which includes

- Studying geographical region of deployment
- Deciding which sensor to use
- Whether to use single tier or multi tier architecture (11)
- Deciding which protocol to use
- How to run same algorithm on different hardware – which may require cross compiling
- Reliability Vs Lifetime tread-off
- Lifetime Vs latency – in case of multi tier network

Designing efficient distributed algorithm is an important task, but implementing it on small resource constrained device is very challenging.

Bibliography

1. *X-MAC: A Short Preamble MAC Protocol for Duty-Cycled Wireless Sensor Networks.* **Buettner, G. Yee, E. Anderson, R. Han.** s.l. : SenSys, 2006.

2. *Integrated Coverage and Connectivity Configuration in Wireless Sensor Network.* **Xiaorui Wang, Guoliang Xing, Yuanfang Zhang.** St. Louis : ACM, 2003.

3. *A Topology Discovery Algorithm for Sensor Networks with Applications to Network Management.* **Nath, Budhaditya Deb and Sudeept Bhatnagar and Badri.** s.l. : IEEE CAS workshop, 2002.

4. NS-2 Simulator. [Online] http://www.isi.edu/nsnam/ns/.

5. *Impact of Radio Irregularity on Wireless Sensor Networks.* **Gang Zhou, Tian He, Sudha Krishnamurthy, John A. Stankovic.** Charlottesville : MobiSYS'04, 2004.

6. *The Mistaken axioms of wireless-network research.* **D. Kotz, C. Newport, C. Elliott.** s.l. : Dartmouth College Computer Science Technical Report, 2003.

7. *Energy-Efficient Communication Protocol for Wireless Microsensor Networks.* **Balakrishnan, Wendi Rabiner Heinzelman and Anantha Chandrakasan and Hari.** Maui, Hawaii : HICSS, 2000.

8. *Telos: Enabling Ultra-Low Power Wireless Research.* **Joseph Polastre, Robert Szewczyk, and David Culler.** s.l. : IPSN/SPOTS, 2005.

9. *A Distributed Clustering Algorithm with an Adaptive Backoff Strategy for Wireless Sensor Networks.* **HE, Yongtao CAO and Chen.** Shanghai : IEICE Transactions on Communications, 2005.

10. *Performance Evaluation of of the IEEE 802.15.4 MAC for Low-Rate Low-Power Wireless Netrok.* **Gang Lu, Bhaskar Krishnamachari,Cauligi Raghvendra.** Los Angeles : IEEE, 2004.

11. *The Case for Multi-tier Camera Sensor Networks.* **Purushottam Kulkarni, Deepak Ganesan and Prashant Shenoy.** s.l. : NOSSDAV, 2005.