Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2005 Proceedings

Americas Conference on Information Systems (AMCIS)

2005

Applications and Routing Management of Wireless Sensor Networks

Subhankar Dhar
San Jose State University, dhar_s@cob.sjsu.edu

Follow this and additional works at: http://aisel.aisnet.org/amcis2005

Recommended Citation
Dhar, Subhankar, "Applications and Routing Management of Wireless Sensor Networks" (2005). *AMCIS 2005 Proceedings*. 197.
http://aisel.aisnet.org/amcis2005/197

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2005 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

Applications and Routing Management of Wireless Sensor Networks

Subhankar Dhar
San Jose State University
dhar_s@cob.sjsu.edu

ABSTRACT

In this paper, we first describe some current and future applications of sensor networks. We present the conceptual framework of distributed routing strategies for wireless sensor networks. We show that under reasonable assumptions, this routing scheme guarantees 'shortest path property' which is quite desirable for sensor networks. We then discuss how this framework can be used to support distributed applications for sensor nodes acting as mobile devices. These schemes work well in low mobility conditions. We also discuss the performance of these heuristics.

KEYWORDS

Dominating set, energy-aware, sensor network, shortest path routing, wireless and mobile communications

INTRODUCTION

The growing importance of sensor networks can hardly be exaggerated. A sensor network is a collection of sensors that form a network and communicate with each other via wireless links. The past decade has shown a phenomenal growth in wireless communications and the future of sensor networks looks very promising from applications standpoint. The sensor nodes are small, low in power, and are inexpensive. These networks are usually multi-hop in nature and they have high redundancy in ambient conditions. Sensor networks are self-organizing and consist of hundreds or even thousands of nodes having limited battery power.

In a sensor network the physical layer deals with the communication issues like modulating and coding across the physical link. The MAC layer provides access control, channel assignment, power management and neighbor list management. The network layer is responsible for addressing and routing. When designing mobile sensor networks, several interesting and difficult problems arise due to shared nature of the wireless medium, limited transmission power (range) of wireless devices. Energy efficient routing is a very important issue for sensor networks as the battery power of the sensors are limited. One of the major goals of routing protocols is to maximize the lifetime of the network because as the energy level of nodes go down, the connectivity between the nodes become weak and eventually the network gets partitioned and cease to work.

Our contribution in this paper can be summarized as follows. We first describe some current and future applications of sensor networks. We present the conceptual framework of distributed routing strategies for wireless sensor networks. We show that under reasonable assumptions, this routing scheme guarantees 'shortest path property' which is quite desirable for sensor networks. We then discuss how this framework can be used to support distributed applications for sensor nodes acting as mobile devices. These schemes work well in low mobility conditions. We also discuss the performance of these heuristics.

SOME COMMERCIAL APPLICATIONS OF SENSOR NETWORKS

Some typical applications are as follows (I.F. Akyildiz et al., 2002; Hac 2003).

a) *Surveillance and security:* Sensor networks are used for monitoring highly secured places and buildings. More and more sensor networks will be deployed for defense and homeland security. Sensor networks can also be used in minefields.
b) *Environmental and habitat monitoring:* These inexpensive devices, once deployed, can offer accurate information about temperature, detect chemicals, critical environment conditions (e.g. generate wild fire alarms); monitor certain behavior patterns like movements of some animals, etc. These networks can be deployed in agricultural fields to monitor temperature, water levels, etc. to ensure ideal conditions for vegetation.
c) *Personal area and home networking*: Sensor networks are quite suitable for home as well as personal area networking applications. Smart sensor nodes can be embedded in various home appliances, refrigerators, micro-wave ovens and can be monitored and operated remotely when connected via external networks like Internet, etc. from outside.

d) *Emergency and Health services*: Sensor networks will provide solutions to emergency services. In case of some disasters like flood, earthquake, sensor networks will be able to detect these conditions and generate alerts for proper actions. Other applications include remote integrated patient monitoring, drug administration in hospitals, etc.
e) *Military applications*: In battlefield, sensors can be deployed for communications among the soldiers in the field. Different military units are expected to communicate, cooperate with each other and within a specified area. In these kinds of low mobility environments, sensor is used for communications where virtually no network infrastructure is available. They are low in cost, self-organized, self-balancing and self-healing. It is easy to scale. They are also used for monitoring forces.
f) *Ubiquitous and embedded computing applications*: With the emergence of new generations of intelligent portable mobile devices, ubiquitous computing is becoming a reality. As predicted by some researchers (Weiser, 1993), ubiquitous computers will be around us, always doing some tasks for us without our conscious effort. These machines will also react to changing environment and work accordingly. These mobile devices will form a sensor network and, gather various localized information and sometimes inform the users automatically.
g) *Location-based services*: Sensors when integrated with location-based information provides useful location-based services. *GPS (Global Positioning System), a* satellite-based radio navigation system, is a very effective tool in highway for exchanging traffic information and vehicle location. This system when integrated with a sensor networks can be very effective for tracking vehicles and location discovery.
h) *Inventory control*: Sensor networks can be used in inventory control, manufacturing and other applications in a warehouse.

RELATED WORK

Routing protocols for sensor are active areas of research and several researchers have proposed several protocols/heuristics in this regard. Here we only describe the ones that have gained popularity in the recent years.

Spin (Heinzelman et al., 1999) is a flooding based protocol which has a broadcast system for data dissemination and works well for mobile sensors. It is also quite scalable and robust. The main drawback is nodes are always active and hence idle nodes consume energy.

The directed diffusion data dissemination protocol proposed in (Intanagonwiwat et al., 2000) uses the sink to send out interest, which is a task description, to all sensors. Each sensor node then stores the interest entry in its cache. The interest entry contains a timestamp field and several gradient fields. As the interest is propagated throughout the sensor network, the gradients from the source back to the sink are set up. When the source has data for the interest, the source sends the data along the interest's gradient path. The interest and data propagation and aggregation are determined locally. Also, the sink must refresh and reinforce the interest when it starts to receive data from the source. Note that the directed diffusion is based on data-centric routing where the sink broadcasts the interest. The main drawbacks of this protocol are (a) the gradient setup phase is expensive and (b) it is not energy aware as the best paths might be used too often.

Low-energy adaptive clustering hierarchy (LEACH) is a clustering-based protocol that minimizes energy dissipation in sensor networks (Heinzelman et al., 2000). The purpose of LEACH is to randomly select sensor nodes as cluster-heads, so the high energy dissipation in communicating with the base station is spread to all sensor nodes in the sensor network. Clusterhead selection is difficult to optimize in many situations.

Since our framework for routing is based on minimum connected dominating set, we will focus on works that use minimum connected dominating set. A *dominating set* in a graph is a set such that each node is either in the dominating set or adjacent to some node in the dominating set. It is well-known that the *MCDS* (Minimum Connected Dominating Set) problem i.e., finding smallest subset *D* of nodes, such that the subgraph induced by *D* is connected and *D* forms a dominating set is an NP-complete problem (Guha and Khuller, 1998). Several researchers have used minimum connected dominating sets to do induce a virtual backbone in ad hoc wireless networks (Alzoubi et al., 2002; Das and Bhargavan, 1997; Rieck et al., 2002). J. Wu and H. Li (Wu and Li., 2001) proposed a heuristic for selecting dominating set based on local neighborhood information and two rules that eliminate redundant nodes in the dominating set that result in a very small dominating set.

ROUTING FRAMEWORK

In a sensor network, energy efficiency is extremely important as the nodes have limited battery power. Hence, routing strategies that choose the shortest path from source to destination will be quite useful in this context. This work is largely motivated by previous work on ad hoc networks where some localized heuristics have been proposed using dominating sets. However, we found that the route through the dominating nodes does not always provide the shortest path and we proposed our own heuristics that produce a clustering-like structure based on *k*-hop connected dominating set, which also ensures that the route through the dominating nodes does always provide the shortest path (*k*-SPR property). This strategy will work well for low mobility environments like sensor networks.

A sensor network is typically modeled as a connected graph with bi-directional links. For the sake of simplicity, the network nodes are presumed to be identical in nature and to have the same transmission radii. In (Dhar, Rieck and Pai., 2003) and (Rieck, Pai and Dhar., 2002), the authors proposed an heuristics for constructing a k-hop connected, *k*-hop dominating set for *G*, where *k* is a fixed integer greater than one. A set of nodes is “*k*-hop dominating" (also called “*k*-dominating") if each node in the network is within *k* hops of a node in the set. “*k*-hop connected” means that given any two nodes *u* and *v* in the set, there is a path beginning with *u* and ending with *v*, such that the hop count between consecutive nodes along the path that belong to the set never exceeds *k*. In fact the sets obtained in (Dhar, Rieck and Pai., 2003) and (Rieck, Pai and Dhar., 2002) have the special property that there exists a path as just described between any two nodes *u* and *v* from the graph, and that this path is a shortest (possible) path connecting *u* to *v*, and that the hop counts from *u* to the first member of the set and from the last member of the set to *v* do not exceed *k*. We refer to this property as the ‘shortest path property’. To construct the set, (Dhar, Rieck and Pai., 2003) introduces the notion of a *k*-SPR set, as follows.

A set *S* of nodes of *G* will be called a *k-SPR set* if given any pair of nodes *u* and *v* of *G* such that $\delta(u,v) = k+1$, there exists a $w \in S$ with $\delta(u,w) + \delta(w,v) = k+1$ and $w \neq u$, $w \neq v$.

Here $\delta(u,v)$ means the hop count from *u* to *v*, and *k*-SPR stands for ‘*k*-shortest path routing’. Under reasonable assumptions, a *k*-SPR set can be shown to be a *k*-hop connected *k*-hop dominating set that has the shortest path property. Deciding whether or not a set is a *k*-SPR set is a local issue. To be more specific, it is possible to consider a certain test that can be conducted based solely on the induced subgraph of a given *k+1*-hop neighborhood. Each node can conduct this test for its own *k+1*-hop neighborhood. The set as a whole is a *k*-SPR set if and only if each node perceives it to be so locally. This follows easily from the definition and (Corollary 1, Dhar, Rieck and Pai., 2003). An alternative way to think about a *k*-hop connected k-hop dominating set is simply as a connected dominating set for the so-called *k-hop closure* G_k graph. This is the graph obtained from the original graph G by adding edges between any pair of nodes that are within *k* hops in G.

Problem Formulation

It is well known that the Set Covering Problem is essentially a problem concerning bipartite graphs that can be stated as follows. Suppose that *H* is a bipartite graph, consisting of two sets of nodes *A* and *B*, where edges only connect nodes from A to nodes from B. Also assume that for every node in B, there is at least one edge connecting it to a node in A. The goal is to find a minimal (or at least small) subset C of A such that every node is B is “covered by” (*i.e.* adjacent to) some node in C.

The problem of finding a *k*-SPR set can be translated into the problem of finding a “covering set” C for the following bipartite graph. Let A be the set of all the nodes in the network. Let B be the set of all unordered pairs *{x,y}* of nodes in the network satisfying $\delta(x,y) = k+1$. Put an edge between a node v from A and a pair *{x,y}* from B if *v* does not equal x or y, but v does lie along some shortest (possible) path connecting x and y. A subset C of A covers B if and only if it is a *k-SPR* set, as is straightforward to check. For more details, see (Dhar, Rieck and Pai., 2003). The following definition will be useful, although it really is just another name for “node degree” in the context of the bipartite graph *H*.

The *covering number* of each node w in A is the number of *{x,y}* pairs in B that share an edge with w. Also, we say that w covers the pair *{x,y}*.

Techniques to generate sets with shortest path property

In order to produce small *k*-SPR sets in a distributed way, we consider certain subgraphs as follows.

The *k-local view* of a node v consists of all the k-hop neighbors of *v*, together with all edges between these, except for the edges that connect two nodes at a distance *k* from *v*.

Each node maintains its *d*-local view by obtaining the necessary link-state information from its neighbors. This requires just *k* rounds of message passing, during which each node sends a message to its adjacent neighbors. We now state the following theorem that has been proved in (Dhar, Rieck and Pai., 2003).

Theorem. *For any node x, y, v in G, let* $\delta(x,v) + \delta(y,v) = k+1$*, the distance* $\delta(x,y)$ *can be computed solely from a knowledge of the (k+1)-local view of v. Moreover, all the shortest paths connecting x to y lie inside the (k+1)-local view of v.*

The following corollary is a useful consequence of Theorem 1.

Corollary 1. Given $u, v \in$ A (*i.e*. nodes in G), and a pair $\{x,y\} \in B$ such that u and v are both adjacent to $\{x,y\}$ in H, the four nodes $u, v, x, y,$ as nodes of G, are within a distance $k+1$ of each other.

k-Shortest Path routing (*k*-SPR) heuristics

k-SPR Based on Highest ID (k-SPR-I)

Let H be the bipartite graph as discussed earlier. Each node in the original graph is randomly assigned a unique positive integer or ID. For each pair $\{x,y\} \in$ B, a node v $\in$ $A_{\{x,y\}}$ is elected into the *k-SPR* set to cover this pair if v has the highest ID among the nodes in $A_{\{x,y\}}$. The node v is in a position to decide this question from its local information. The resulting set produced by the k-SPR-I heuristics is clearly a *k*-SPR set.

k-SPR Based on Covering Number (k-SPR-C)

In the k-SPR-I heuristics, given a pair $\{x,y\}$ (from B), the node that covered $\{x,y\}$ with the highest ID was elected for inclusion into the k-SPR set. Here, instead we elect the node having the highest *priority*. The priority of each node is defined to be the ordered pair of numbers *(covering_number, ID)*, lexicographically ordered. This variant of *k*-SPR-I will be referred to as k-SPR-C ("C" for "covering number").

The *k*-SPR-C heuristics requires an additional $k+1$ round of local broadcast. After the first $k+1$ rounds, each node is aware of its own $(k+1)$-local view, and is able to compute its own covering number. The subsequent $k+1$ rounds of broadcast are used to allow each node to transmit its covering number to each of its $(k+1)$-hop neighbors.

k-SPR Based on Energy (k-SPR-E)

The k-SPR-E heuristics can be described as follows. We obtain k-SPR sets by assigning battery levels as weights to each node. Our k-SPR-E ("E" for "energy") algorithm is identical to k-SPR-C, except that nodes are elected so as to have the *highest* possible weights, rather than the highest possible covering numbers. That is, it elects nodes so as to maximize (residual energy, ID). It is a priority based k-SPR set selection where priority is based on each node's remaining battery life.

Next, the k-SPR-E and k-SPR-C algorithms can be blended to form two other algorithms: "k-SPR-CE" and "k-SPR-EC". When using k-SPR-CE to elect a network node to cover an election pair in H, the candidate nodes are first compared based on covering number, just as in k-SPR-C. In the case of a tie, the node weights are considered in order to break the tie, just as IDs are used to break a tie in k-SPR-C. If comparing weights also leads to a tie, then IDs are used to break this tie. So here elections are based on the triple (covering number, residual energy, ID), lexicographically ordered. This means choosing the largest covering number, and in case of a tie, choosing the lowest node weight, and in the case of another tie, choosing the highest node identifier. Similarly the k-SPR-EC algorithm is (lexicographically) based on the triple (residual energy, covering number, ID), and so node weights are of principal concern for the *k*-SPR-EC algorithm.

GREEDY k-SPR HEURISTICS (k-SPR-G)

In order to further reduce the set size produced by the earlier heuristics, we applied a greedy heuristics, which can be implemented in a distributed manner. Here each node begins in an "undecided state", maintains information about its $(k+1)$-local view, and executes the following steps. From Corollary 1 we know that a node v can "see" all the node pairs $\{x,y\}$ it covers, and also the rest of the nodes that cover such pairs. The fact that the distributed heuristics below terminates and that the set of "selected" nodes is a *k-SPR* set follows immediately from the discussion of similar distributed greedy heuristics, as found in (Jia et al., 2001). All of these, including ours, are in essence just distributed implementations of the greedy heuristics for the Set Covering Problem.

Step 1: Each node v gathers information about its $(k+1)$-local view. This requires $k+1$ rounds of message passing. Let B_v denote all the node pairs $\{x,y\}$ covered by v. Let C_v be the set of all nodes that cover some node pair in B_v (so $v \in C_v$).

Step 2: v computes its current covering number $|B_v|$ and sends this information, along with its status (undecided, selected or not selected), in a message to each node in C_v. (Note that the first time this step is executed, all nodes are undecided, and the last time a node executes this step, it will be in one of the two decided states.)

Step 3: If *v* has entered one of the two decided states (selected or not selected), then it essentially terminates its participation in this heuristics, except to help route messages between other nodes. Otherwise, if it is still undecided, then ....

Step 4: *v* waits until it receives messages as in Step 2 from each node in C_v. For each such node *u* that has become decided, *v* removes *u* from C_v, and if *u* has become selected, *v* also removes any pairs from B_v that *u* covers. Accordingly, *v* re-computes its covering number as necessary.

Step 5: If *v*'s covering number is now zero, then v enters the "not selected" state, and loops back to Step 2. Otherwise....

Step 6: *v* checks to see if its own priority is the highest among all the nodes of C_v. If it is, then *v* enters the "selected" state. In either case, it loops back to Step 2.

Greedy Refinement of k-SPR-I with *n* covering nodes (k-SPR-C_nG)

This new heuristics is a variation made on *k-SPR-C*. In *k-SPR-C*, for every pair of nodes with distance *(k+1)*, say {*x,y*}, a node in the set $A_{\{x,y\}}$ that has the highest priority is admitted to the *k-SPR* set. In the initial selection phase of *k-SPR-*C_n*G*, for each pair *{x,y}* in set *B*, *n* such nodes are admitted to the initial set.

Obviously, the initial set *A* for *k-SPR-*C_n*G* is considerably bigger than the set obtained from *k-SPR-C*. Having a smaller initial however set limits the scope of optimization in the second phase. Thus, from the optimization point of view, it is more desirable to have a large initial set, although a large initial set requires more time and messages to process. The purpose of *k-SPR-G* is to produce a very small *k-SPR* set by using the largest initial set possible, i.e. the set of all nodes in the graph. However, the processing time is large and many messages need to be passed. *k-SPR-*C_n*G* takes far less time and involves far fewer messages, but results in a large set.

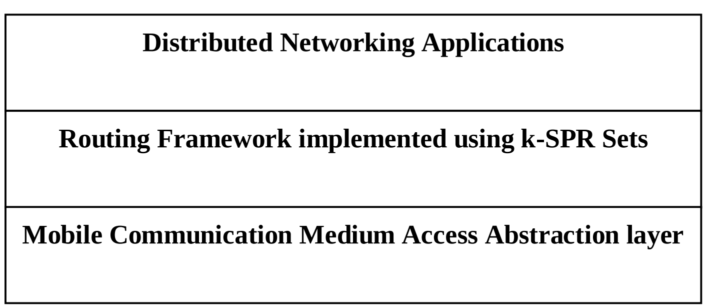
SUPPORT FOR DISTRIBUTED APPLICATIONS

We now discuss how the routing framework can be used to support distributed networking applications provided by sensor networks. This framework sits on the top of the Mobile Communication Medium Access Abstraction layer which will be described later. This framework will allow a system architect, the necessary interfaces including APIs (Application Programming Interface) to run distributed applications for a mobile environment using smart sensors.

Mobile Communication Medium Access Abstraction Layer

This layer provides all the access interfaces that a mobile device (sensor node) would have for communicating to another mobile device wirelessly. It will abstract out the wireless communication features of a mobile device. Thus the layer will provide interfaces to be able to listen to a particular frequency and transmit data on a certain frequency. When there is a nearby mobile device that transmits data at the listening frequency, data will be received by the first mobile device.

The abstraction layer will also provide a way for the mobile device to specify its location. This allows emulation of a moving mobile device in an area filled with other mobile devices. Loss of data due to interference or collision can be simulated within this layer. The access methods will be very similar to Media Access Layer (MAC) of the TCP/IP protocol stack.



Routing framework implemented using k-SPR sets

This layer will use the access interfaces from the layer below and implement a routing scheme at each mobile device. The routing scheme will use knowledge of the dominating sets computed using distributed communication across all the mobile devices. This layer will provide standard TCP/IP or UDP style access interfaces to the layer above. Typical methods would include opening a socket, connecting a socket to a remote port, sending and receiving data from sockets and, closing and shutting down connections.

PERFORMANCE EVALUATION

The main goal of our performance analysis is to find and compare the size of *d*-hop connected *d*-hop dominating sets produced by various versions of our *k-SPR* heuristics - *k-SPR-I, k-SPR-C, k-SPR-G, k-SPR-IG, k-SPR-CG,* k-SPR-C_nG. Each heuristics is designed in a distributed manner where each node will gather its *d*-hop neighborhood information by exchanging messages with its direct neighbors. These heuristics are implemented and experiments are run on a single machine.

Experimentation

For each experiment, a random disk graph is used to model the topology of a sensor network. A disk graph is a graph in which a node is connected to all other nodes within a radius defined for the graph. A random disk graph is a disk graph in which nodes are positioned randomly. The radius of a disk graph represents the transmission range of a node in the sensor network assuming that all nodes in the network use the same transmission power. Also all links are assumed to be bi-directional.

In this experimentation, we tried to keep the density of the network constant and thus the degree of each node constant too, to some extent. Given a node density and the number of nodes in a network, the size of simulation area was computed and *x* and *y* coordinates of each node was randomly chosen within this boundary.

We ran experiments for each heuristics with different values of *d* and different number of nodes. The heuristics considered were two *k-SPR* heuristics - *k-SPR-I* and *k-SPR-C* where the covering number in bipartite graph was used as the priority; and two greedy refinement heuristics - *k-SPR-G,* and k-SPR-C_nG.

Results

In terms of the set size, the greedy refinements of *k-SPR* heuristics performed better than any of *k-SPR* heuristics considered, which is not surprising considering that *k-SPR* heuristics produce a *d*-hop connected *d*-hop dominating set with the shortest path property, an extra property that entails some overhead. Therefore, our aim is to produce a *k-SPR* set whose size is as small as possible.

Nodes	k-SPR-I	k-SPR-C	k-SPR-G	k-SPR-C_2G
100	19.9	14	8	8.1
200	45.7	38.3	21.4	21.8
300	68.3	56.4	31.4	31.1
400	101.9	80.3	45.6	45.5
500	123.8	97.7	55.7	57.2

Figure 1. Set size produced by various heuristics for k = 5

Fig. 1 shows the average size of the sets produced by each heuristics when $d = 5$ *and* $n=2$ *for* k-SPR-C_nG, while the number of nodes varies from 100 to 500.

The first thing to notice is that the sets produced by basic *k-SPR* heuristics without the greedy refinement, i.e. *k-SPR-I* and *k-SPR-C*, are larger than the set produced by the greedy heuristics. The greedy refinement produced a significant reduction in the set size; after applying the greedy refinement, *k-SPR-I* and *k-SPR-C* sets were reduced in size by 35-45%. Among various *k-SPR* heuristics, *k-SPR-G* performed the best; the sets generated by *k-SPR-G* are lowest in size, while retaining the shortest path property.

The heuristics, k-SPR-C_2G, is a variation on *k-SPR-C* and produced a set that is very close to *k-SPR-G*. It should be noted that, unlike other greedy refinement heuristics, *k-SPR-G* does not have an initial selection process where a subset of nodes in the network with some desired attribute, e.g. high covering number, is selected before optimization heuristics is applied to the set.

k-SPR-C_2G while producing a set with similar size to *k-SPR-G*, makes use of priority in their initial selection process. Making use of priority is significant if attributes other than the shortest path property (e.g. remaining battery power) need to be considered in the set selection process.

CONCLUSION

The commercial and business applications of sensor networks are on the rise. More and more sensor networks will be deployed in the future. However, there are some interesting research issues that are wide open. The routing protocols in sensor networks will have to deal with energy efficiency, scalability, and low latency among many other issues. In this paper, we have shown how shortest path routing can be used to route messages efficiently in sensor networks. Our main motivation was the well studied problem of routing in an energy efficient manner, by using nodes with strong batteries as much as possible, and by avoiding nodes with weak batteries as much as possible, while still keeping the routing path as short as possible. By using our *k*-SPR method, which can assign weights to nodes as an arbitrary function of the strength of the corresponding batteries, it is possible to come up with a variety of approaches to this problem. We also discussed how the routing framework can be used to support distributed networking applications provided by sensor networks.

REFERENCES

1. I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci , (2002) Wireless Sensor Networks: A Survey, *Computer Networks*, Vol 38, pp 393-422.
2. K.M. Alzoubi, P. Wan and O. Frieder, (2002) New Distributed Algorithm for Connected Dominating Set in Wireless Sensor Networks, *Proc. 35th Hawaii Int'l Conf. System Sciences*, Hawaii, January 2002, pp. 1-7.
3. B. Das and V. Bharghavan, (1997) Routing in Ad-Hoc Networks Using Minimum Connected Dominating Sets, *Proc. IEEE Int'l Conf. Communications (ICC'97)*, June, pp. 376-380
4. S. Dhar, M. Q. Rieck and S. Pai, (2003) On Shortest Path Routing Schemes For Wireless Ad-Hoc Networks, *Proc. 10th Int'l Conf. High Performance Computing (HiPC'03),* Springer-Verlag, Hyderabad, December, pp. 130-141.
5. S. Guha and S. Khuller, (1998) Approximation algorithms for connected dominating sets, *Algorithmica*, 20(4), April, pp. 374-387.
6. Anna Hac, (2003) Wireless Sensor Network Designs, John Wiley & Sons.
7. W.R. Heinzelman, J. Kulik, H. Balakrishnan, (1999) Adaptive protocols for information dissemination in wireless sensor networks, *Proceedings of the ACM MobiCom'99*, Seattle, Washington, pp. 174–185.
8. W. R. Heinzelman, A. Chandrakasan, H. Balakrishnan, (2000) Energy-efficient communication protocol for wireless microsensor networks, *IEEE Proceedings of the Hawaii International Conference on System Sciences*, January, pp. 1–10.
9. C. Intanagonwiwat, R. Govindan, D. Estrin, (2000), Directed diffusion: a scalable and robust communication paradigm for sensor networks, *Proceedings of the ACM MobiCom'00*, Boston, MA, pp. 56–67.
10. S. Kutten and D. Peleg, (1998) Fast Distributed Construction of Small *k*-Dominating Sets and Applications, *J. Algorithms*, 28(1), July, pp. 40-66.
11. L. Jia, R Rajaraman and T Suel, (2001) An Efficient Distributed Algorithm for Constructing Small Dominating Sets, *Proc. Ann. ACM Symp. Principles Distributed Computing,* August, pp. 33-42.
12. S. Rampone, (2001) Probability-driven Greedy Algorithms for Set Cover. *VIII SIGEF Congress "`New Logics for the New Economy"*, Naples, Italy, September.
13. M. Q. Rieck, S. Pai and S. Dhar, (2002) Distributed Routing Algorithms for Wireless Ad Hoc Networks Using d-hop Connected d-hop Dominating Sets, *Proc. 6th Int'l Conf. High Performance Computing: Asia Pacific*, Tata McGraw-Hill, Bangalore, December, pp. 443-450.
14. J. Wu and H. Li, (2001) A Dominating-Set-Based Routing Scheme in Ad Hoc Wireless Networks, *Telecommunication Systems*, 18(1-3), pp. 13-36.
15. M. Weiser. (1993) Some Computer Sciences Issues in Ubiquitous Computing. *Communications of the ACM* 36(7), July.