Journal of Applied Science and Agriculture, 8(5) October 2013, Pages: 515-521

AENSI Journals

Journal of Applied Science and Agriculture

Journal home page: www.aensiweb.com/jasa/index.html

Wireless Sensor Network Simulator to Study Routing Protocols

Mojgan Eivazzadeh Kalejahi

[1]*Department of Computer Engineering, Osku Branch, Islamic Azad University, Osku, Iran*

ARTICLE INFO

Article history:
Received 13 October 2013
Received in revised form 17 November 2013
Accepted 22 November 2013
Available online 15 December 2013

Key words:
SimulationsOptimizedBroadcast
Protocols Sensor Networks
Wireless Sensor Networks

ABSTRACT

Background: Wireless sensor networks are applied in many applications. The routing protocols that are developed for sensor networks must be energy efficient and scalable. It is necessary to perform a detailed analysis of sensor networks for understanding of the performance of them. **Objective:** Network simulator scan inform us about the performance and behavior of these protocols on various network topologies. This research considers simulation of routing protocols at Sensor Networks Research Laboratory. **Results:** Sensor Simulator is a discrete event simulation framework for sensor networks built over OMNeT++. This frame work allows the user to debug and test software for distributed sensor networks. **Conclusion:** This research implements a comparative study on Directed Diffusion Routing Protocol on both ns2 and SensorSimulator.

To Cite This Article: Mojgan Eivazzadeh Kalejahi., Wireless Sensor Network Simulator To Study Routing Protocols. ***J. Appl. Sci. & Agric.,*** *8(5): 515-521, 2013*

INTRODUCTION

A Sensor Node consists ofone or more sensing elements (motion, temperature, pressure, etc.), a battery, and lowpower radio trans-receiver, microprocessor and limited memory, mobilizer (optional), aposition finding system. An important aspect of such networks is that the nodes are unattended, have limited energy and the network topology is unknown. Many design challenges that arise in sensor networks are due to the limited resources they have and their deployment in hostile environments.

Sensor nodes are deployed in environments where it is impractical or infeasible for humans to interact or monitor them. These unattended nodes may have effect on the efficiency of many military and civil applications such as target field imaging, distributed computing, intrusion detection, security and tactical surveillance, inventory control,disaster management and detecting ambient conditions. Some applications requiresensors to be small in size and have short transmission ranges to reduce the chances ofdetection. These size constraints cause further constraints on CPU speed, amount of memory, RF bandwidth and battery lifetime. Hence, efficient communication techniquesare essential for increasing the lifetime and quality of data collection and decreasing the communication latency of such wireless devices (Akyildiz *et al*., 2002).

Unlike the mobile ad hoc networks, sensor nodes are most likely to be stationary for the entire period of their lifetime. Even though the sensor nodes are fixed, thetopology of the network can change. During periods of low activity, nodes may go to inactive sleep state, to conserve energy. When some nodes run out of battery power anddie, new nodes may be added to the network. Although all nodes are initially equipped with equal energy, some nodes may experience higher activity as result of region they are located in. Communication pattern is intermittent and sensor applications are data-centricin nature. An important property of sensor networks is the need of the sensors to reliably disseminate the data to the sink or the base station within a time interval that allows theuser or controller application to respond to the information in a timely manner, as out of date information is of no use and may lead to disastrous results. Another important attribute is the scalability to the change in network size, node density and topology. Sensor networks are very dense as compared to mobile ad hoc andwired networks. This arises from the fact that the sensing range is lesser than the communication range and hence more nodes are needed to achieve sufficient sensing coverage. Sensor nodes are required to be resistant to failures and attacks.

The current work is focused to study the performance and behavior of these routing protocols on various network topologies. The report begins with an introduction to Wireless Sensor Networks and the importance of routing protocols in various Sensor Network Applications. Section 2 gives an overview on the existing simulators and a brief description of OMNeT++ Framework. Section 3 gives an overview on the design of Sensor Simulator Architecture and how various protocols can be added at different layers without much

Corresponding Author: Mojgan Eivazzadeh Kalejahi, Department of Computer Engineering, Osku Branch, Islamic Azad University, Osku, Iran
Tel.: +98 9143096003; E-mail: eyvazzade@iauosku.ac.ir (M. eivazzadeh).

dependency. Section4 gives an overview on Optimized Flooding Protocoland its implementation in the Simulator.

2- Literature review:

Network simulators are very important for analyzing various protocols designed for a network (wired or wireless) and its necessity is very well known in the field of research. Especially, the research challenges in wireless sensor networks brought many open issues to network designers. The techniques used for analyzing the performance ofany wireless networks are physical measurement, analytical methods and computer simulation. The constraints imposed on sensor networks, such as energy limitation, fault tolerance make the algorithms for sensor networks to be quite complex and usually defyanalytical methods that have been fairly effective for traditional networks. And physical measurement is not possible because of the unsolved research problems in the field of sensor networks. Hence computer simulations appear to be the only feasible approach than anything else (Chen *et al*., 2004).

This extension might be easy for traditional networks but not for sensor networks where the protocols are not very dominant and it is very unlikely that a single algorithm will be optimal under various circumstances. Also various simulation studies show that the memory utilization of ns2 is very high and increases for very large simulations. Since the application areas in sensor networks require many number of sensor nodes in a sensor field, the simulations in ns2 take lot of memory. Also another disadvantage posed by ns2 comes from its open source nature. The documentation is often limited and out of date with the current release of the simulator. The problems can be solved with the help of dynamic news groups and going through the source code. Also, the consistency of code is lacking as it is developed by many users. There are no tools describe simulation scenarios and analyze or visualize simulation trace files. The tools for ns2 are written with scripting languages. The lack of generalized analysis tools may lead to different people measuring different values for the same metric names (Sobeih *et al*., 2008).

OPNET modeler is another popular commercial platform for network modelingand simulation which allows the design and study of communication networks, devices, protocols, and applications with unmatched flexibility and scalability. This is used by many prestigious technology organizations to accelerate the research and development process. OPNET Modeler is based on a series of hierarchical editors that directly parallel the structure of real networks, equipment, and protocols. The wireless model uses a stage pipeline to determine connectivity and propagation among nodes. Modeler’s object oriented modeling and hierarchical editors mirror the structure of actual networks and network components (Mallanda *et al*., 2012). The difficulty with OPNET Modeler is to build the state machine for each level of the protocol stack. It is difficult to abstract such a state machine starting from a pseudo-coded algorithm. But state machines are the most practical inputfor discrete simulators. Hence, it is possible to reuse a lot of existing components (MAC layer, transceivers, links, etc.) improving the deployment process. But on the other hand, any new feature must be described as a finite state machine which can be difficultto debug, extend and validate (Tannenbaum, 2002).

Also it is commercial and is not available for public which becomes the biggest disadvantage for working on it. J-Sim is an open-source, component based network simulation environment developed entirely in Java by Ohio State University (initially and later by University ofIllinois). This along with the autonomous component architecture makes it a truly platform-neutral, extensible, and reusable environment. The Sensor Network Framework developed in J-Sim provides an object-oriented definition for target, sensor and sink nodes; sensor and wireless communication channels; and physical media such as seismicchannels, mobility model and power model (Durresi *et al*., 2005). The simulation analysis described in (Dommety and Jain, 1996) show that the execution speed of J-Sim is less compared to many other simulators andthis happens because of its implementation in JAVA. But the memory consumption of HSimis less compared to others and this advantage comes from its garbage collectors. GloMoSim developed initially at UCLA Computing Laboratory, is a scalable simulation environment for wireless and wired networks systems developed. It is designed using the parallel discrete-event simulation capability provided by a C-based parallel simulation language. It currently supports protocols for purely wireless networks and is built using a layered approach. Standard APIs are used between the different layers and allow the rapid integration of models developed at different layersby users. The difficulty with GloMoSim was to describe a simple application that bypasses most OSI layers. The bypass of the protocol stack is not obvious to achieve as most applications usually lie on top of it. The architecture is also not very flexible compared to other simulators.

3- Architecture of sensor simulator:

The goal of the frame work provided by Sensor Networking Research Laboratory (Cavin *et al*., 2006) is to reduce the interdependence between modules and increase the reusability. The development of the simulator was done in such a way that the above goal is achieved. Though many students were working in the project, certain coding practices were followed. The following section gives the architecture of Sensor Simulator and adescription on the various modules implemented. The architecture closely models a Sensor Network scenario

(Cavin *et al*., 2006) which can be represented by the high-level representation shown below in Figure 1. The sensor model can be represented by the Sensor Nodemodel and the Power model.

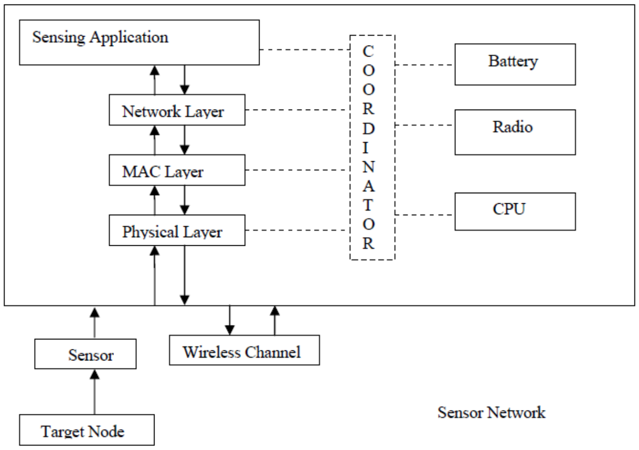

Fig. 1: Sensor Node Representation in a Network

The framework takes advantage of the design features of OMNeT++. The object –oriented approach makes the framework more flexible. It takes the advantage of modular simulation models as they can be reused flexibly. The Sensor Node model represents the wireless network protocol stack and the sensor applications. The power model represents the hardware of the sensor node: the CPU, Sensor and RF transceiver. The two models act in parallel and achieve the task assigned to a sensor node. The state of the hardware model is changed based on the operation carried out by the software model of a Sensor Node. The power model in a Sensor Node is hardware abstraction of sensor node. This interacts with sensor node model to estimate the power usage. The power model comprises of a single energy source and many consumers. The battery module is a provider with finite amount of energy. The consumers are radio, CPU and other sensing devices that may be added to the device as illustrated in Figure 1. The dashed connections between the CoOrdinatormodule to all other modules represent the direct communication between it to others. And the arrowed lines represent the gate connections between modules. Consumers report their power usage change to the energy source (battery) and the energy source updates the energy.

The simulator is designed in the form of a layered architecture and the communication between the different layers and modules are accomplished through message passing (Nicol, 2002). The implementation has a hierarchical structure where in the code is divided into base classes and sample classes. Any layer has a base class definition whichis a derived class from Layer Base. LayerBase defines the properties of any layer in the protocol stack. It defines the gate connections and the required parameters that are needed for any layer.

The base class derived from LayerBase has the properties for that particular layer.The sample classes derived from the base class has the implementation needed by users. Hence a user has to derive his class from the base class for his protocol at a particular layer. In this way, all the protocols in samples directory are independent of each other and can also be used collectively. These details are explained in the following section for all the layers. The following section also describes the different modules of the framework.

Common directory has Co Ordinator, packets structures for Network and Maclayer and other constants and attributes used for simulation. This directory is derived by all other directories of the samples directory.TargetBase class is the base class that represents the Target Node. It has the base class function alities that are essential for any target node which includes the position ofthe target node, its id, its speed etc. Target Node Simple extends the Target Base and has the functional implementation of any target node. Any target node module maintains gate connection with the sensor channel. The simple class generates the stimuli and passes the message to the sensor channel.

The Sensor Node module describes the behavior of a sensor node in the simulation. It is a compound module with different layers of the protocol stack as its submodules. The Sensor Node module definition and the class represent all the componentsof the sensor node.

One of the features that the Sensor Simulator incorporates in a sensor node moduled escription is the addition of a coordinator module that acts as an interface between allthe other sub-modules or layers in a sensor node. As such, the Coordinator Module has the functionalities that coordinate the activities of the hardware and the software modulesin a sensor node. The module has to be extended with an added functionality for accessing the properties of the newly added hardware modules or consumers. It has reference to all the layers of a sensor node and all the layers in the sensor node may access the Coordinator. Hence with the help of this coordinator module, any layer may access and update the properties of the other layer. When there is any transmission orreception of a packet, the physical layer has to inform the battery about the decrease in it senergy accordingly. This is not done directly between physical layer module and the battery module but is done through the coordinator. Hence physical layer module indicates the coordinator module and this in turn indicates the battery module. The important advantage of this feature is that the individual layers need not have a reference to each of the implementations of a layer. It can have reference only to this coordinat or module.

3- Comparative study on directed diffusion:

The initial version of Sensor Simulator is verified by making comparisons with ns2. The performance of the simulator is tested in terms of execution speed and memory consumption. This is done by implementing Directed Diffusion routing protocol with a similar setup as that of ns2. The implementation details are maintained same in both of them. The reason to choose Directed Diffusion was that it being a very well known and useful protocol implementation for a new simulator. The comparisons with a protocol that is under research will help further study on it.

Directed Diffusion is a new data dissemination paradigm for sensor networks and is data centric. Data generated by the nodes have attribute value pairs. Interests are generated by nodes for the named data and the data matching the interest is drawn down to wards that node (Intanagonwiwat *et al*., 2003). Intermediate nodes cache, or transform the data.

The details of the protocol are as follows: The nodes that generate queries are called subscriber nodes and generate queries at a regular interval. The subscribernodeinitially generates beacon messages and gets the information about neighbors from beaconreplies and then forwards the query. Each node follows this procedure and useGeographic Routing to forward the query to the region. If a node in the path does not have any neighbors or all its neighbors are away from the region, then it sends a message to its parent node that it is a dead-end. The parent node on updating the cost of the unreachable node, forwards the query in an alternate route towards the region. In the specified region, the interest is recursively flooded. The interest cache is maintained at each of the nodes in the path with its gradient of interest to each of the neighbors. The nodes in the region that have the specified properties of the interest send out data. These nodes are referred to as Publishers (Cavin *et al*., 2006).

The data is marked as Exploratory to reinforce the path that was taken by the interest. On receiving the data marked as Exploratory by the subscriber, positive reinforcement message is sent out by the Subscriber node. Each node on path forwards this message thus reinforcing the path to the region. When the node reinforces a path, its cost to the region is known and this cost is sent back to its source node, which updates the cost information of that node to the particular region of interest. Thus the path with the highest cost is always maintained, reinforcing the route. The data from the region follow the path established by the reinforced messages. The nodes in the region send out data at the rate that is specified in the query. Data caching is implemented in intermediate node sand so the data requested by different subscribers from the same region maybe satisfied by the common node in the path thus reducing the traffic and redundant messages. The data marked as exploratory are sent to identify better paths and reinforce at regular intervals. Also the neighbor-updating procedure phase is carried out, i.e. at regular intervals the beacon messages are broadcast and beacon-reply messages are sent by neighbors thus maintaining latest neighbor information (Cavin *et al*., 2006).

4-Implemention of raw:

As discussed in the introduction Section that sensor networks are energy constraint, energy efficient communication techniques are very critical for increasing the lifetime of sensor nodes. Hence the design of a good power management protocol for wireless sensor networks needs to consider the following attributes; energy efficiency and scalability to the change in network size, node density and topology where as latency, fairness and bandwidth, which are generally the primary concerns in traditional wireless voice and data networks, are secondary in sensor networks.

The Random Asynchronous Wakeup (RAW), a power management scheme is explicitly designed for wireless sensor networks, focusing on the above discussed issues. With reduced energy consumption, the protocol achieves good scalability and low latency. This is achieved by reducing idle listening; by making the sensors operate at very low-duty cycle modes. And a low duty cycle increases latency and reduces throughput.

RAW uses the concept of Stateless Non deterministic Geographic Forwarding (SNGF). The Protocol consists of two schemes; routing based on forwarding sets and random wakeup scheme. The routing protocol is designed to take advantage of the fact that sensor networks are densely deployed. Unlike conventional routing protocols, a packet can be forwarded to any of the several paths existing between the source and destination nodes, where the path lengths of these paths are comparable to the shortest path. The Random Wakeup Scheme allows a node to be active during a randomly chosen fixed interval in each time frame.

4-1- The Protocol:

Each sensor gets up periodically, transmits a beacon message indicating that it isready to receive or forward a message. It waits for duration txfor a reply. If it gets an RTS from any of its neighbor in that duration, it receives the message and extends the duration of its idle time. Then it checks if it can forward the message to any of its neighbor. If no neighbor in the forwarding set is awake it waits until a neighbor is awake.

Then it forwards the message to that node and goes to sleep again.Time axis is divided into fixed-length time frames of length *T*. Let Tsetupbe the time taken by a sensor to send a beacon message once it is awake and receive a reply consisting of it neighbor information. txis the duration that the sensor waits for an RTS.

4-2-Receive Transmit:

A sensor in this state performs the tasks of receiving and transmitting packets. It should be observed that a sensor will be awake until it could forward the packet, after which it goes back to Ready to Receive state.

4-3-Performance of RAW:

The performance of the protocol is verified on a standard network simulator ns2.

Various scenarios were tested with MAC 802.11 at the MAC Layer with a simple Propagation Model. The simulations were carried on a 5RX5R network with a density of6 nodes per RXR region and hence 150 nodes. The transmission radius can be varied accordingly for different network topologies. The model parameters and limits on transmission bit rates and energy ratings are set according to Crossbow MICA2 sensor nodes (Intanagonwiwat *et al.*, 2003). Nodes were deployed randomly in the rectangular region. The energy consumption for switching the radio from idle to sleep modes and vice versa is assumedto be negligible and hence not considered. The raw available bandwidth for each node is set to 1Mbps. The functionality of 802.11 is changed accordingly so that the node will be able to with stand sleep and idle schedules.

The simulations shown in Figure 2 are for a network of 150 nodes with a schedule period *Ti* being 0.5sec. There were 10 source nodes that generate packets at adate rate of 1-5 packets/sec and the performance is tested when there are one and two destination nodes. It is observed that for a fixed density and with an increase in traffic, the awake time of the nodes does not vary much with the average latency being increased considerably. This can be seen in Figure 3.

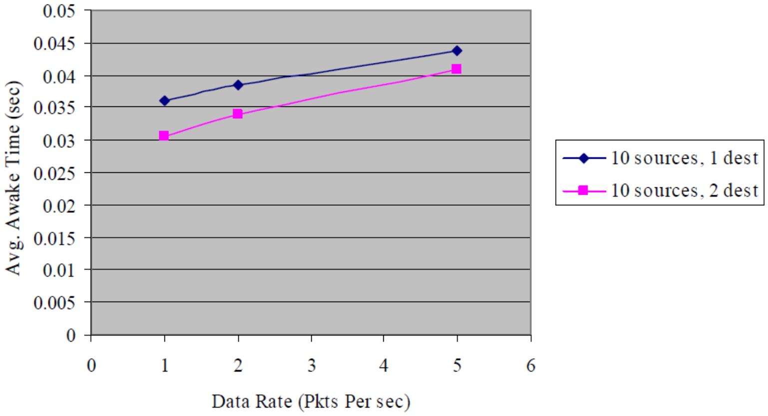

Fig. 2: Effect of Awake Time of Nodes with Traffic

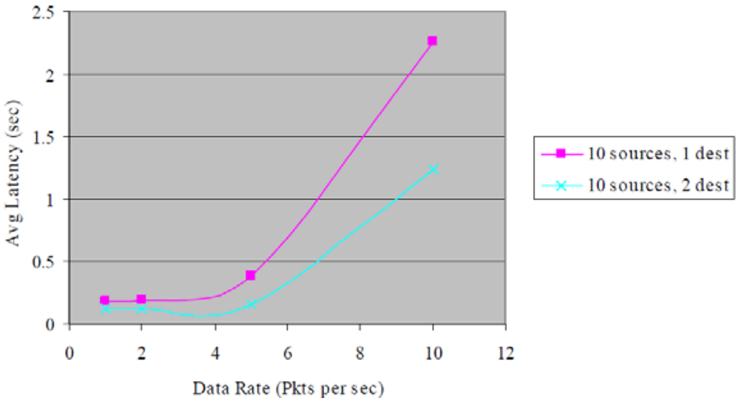

Fig. 3: Effect on Latency with Traffic

The performance of the protocol is also seen when the schedule period changes.

As seen in Figure 4, for a network size of 5RX5R with 500 nodes, density being 20nodes per RXR region, the algorithm is tested for a schedule period of 0.25sec, 0.5 secand 1.0 sec. Data is generated by one source at a rate of one packet/sec and is simulatedfor 50 simulation seconds. The average latency is calculated for these simulations. All the parameters remain the same as noted above.

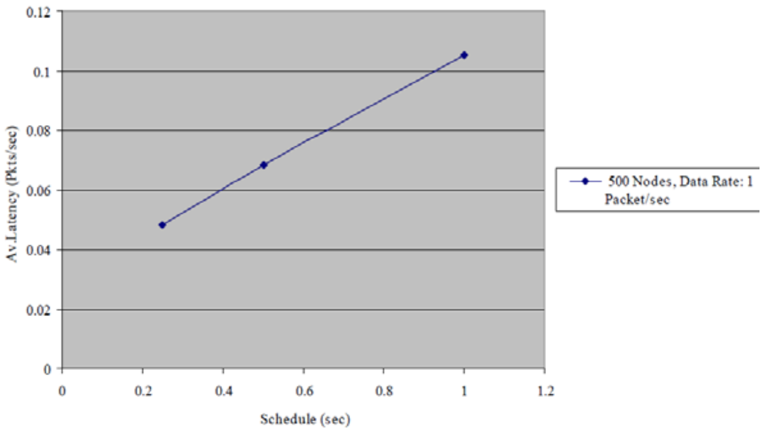

Fig. 4: Effect of Schedule Period on the performance of protocol

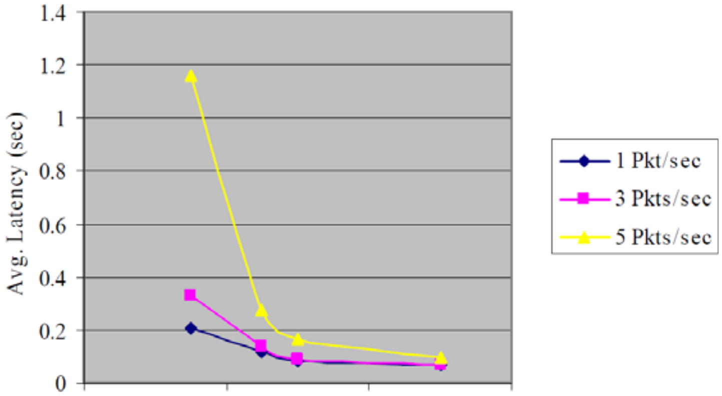

Fig. 5: Performance of the protocol for various densities.

The simulations observed in Figure 5 shows that for a network of same rectangular region (5RX5R), when the density is increased from 4 nodes per RXR regionto 10 nodes, 12 nodes and 20 nodes per RXR region, the average latency for the over all simulation is decreased and this shows the performance of the protocol for a network of very large densities. The scenario is tested for varying data rates from one packet/sec to five packets/sec. The schedule period is maintained as 0.5 sec for this set of simulations.

Conclusion:

This research provides the implementation of various routing protocols for the Simulator developed at LSU. The comparisons of Directed Diffusion protocol in Sensor Simulator with ns2 validates the implementation details of various modules developed in the simulator. The study of the various routing protocols with 802.11 adds to the modules developed for the simulator and enables the further analysis. This research also provide senxtensions to the 802.11 MAC Layer, Physical Layer and the Energy Module that are developed in the initial version. The critical task of the MAC layer to consider the sleepand idle switching of a sensor node is carefully designed as this being the necessary taskfor a sensor node in a sensor network application. Script files for collecting the routing protocol statistics are included in the simulator. The simulator and the support provided makes it very easy to develop and test protocols very fast and obtain results for largesimulations at a reasonable amount of time. Simulations were carried for a sensor network of 2000 nodes and also with a density of 11 nodes per transmission region. This shows the scalability achieved by the simulator. The simulations show that the performance of simulator in terms of execution time remains the same for large numberof nodes also. The comparisons made with ns2 validate this. The simulation analysis shows that the algorithms Broadcast Protocol for Sensor Networks, Efficient Coordination Protocol for Sensor Networks and Random Asynchronous Wakeup

Protocol are efficient in terms of energy and network life time.These implementations in the simulator expand the set of protocols developed for it.

REFERENCES

Akyildiz, I.F., W. Su, Y. Sankarasubramaniam, E. Cayirci, 2002. Wireless Sensor Networks: A Survey. Computer Networks, 38: 393-422.

Cavin, D., YoavSasson, Andre Schiper, Distributed Systems Laboratory, On theaccuracy of MANET simulators.

Chen, G., J. Branch, M.J. Pflug, L. Zhu and B. Szymanski, 2004. SENSE: A SensorNetwork Simulator. Advances in Pervasive Computing and Networking. B. Szymanksi and B. Yener, Springer, pp: 249-267.

Dommety, G. and R. Jain, 1996. Potential networking applications of global positioningsystems (GPS).Tech. Rep. TR-24, CS Dept., The Ohio State University.

Durresi, A., V. Paruchuri, R. Kannan, S.S. Iyengar, 2005. "Optimized Broadcast Protocolfor Sensor Networks, IEEE Transactions on Computers, 54(8): 1013-1024.

Intanagonwiwat, C., R. Govindan, D. Estrin, J. Heidemann and F. Silva, 2003. Directeddiffusion for wireless sensor networking.*IEEE/ACM Transactions on Networking*, 1(1): 2-16.

Mallanda, C., A. Suri, V. Kunchakarra, S.S. Iyengar, R. Kannan, A. Durresi, Simulating Wireless Sensor Networks with OMNeT++ , submitted to IEEETransactions on Computers.

Nicol, David, M., 2002. Dartmouth College, Comparison of Network Simulators Revisited.

Sobeih, A., Wei-Peng Chen., C. Jennifer, Hou, Lu-Chuan Kung, Ning Li, Hyuk Lim., Hung-Ying Tyan, and Honghai Zhang, J-Sim: A Simulation and EmulationEnvironment for Wireless Sensor Networks.

Tannenbaum, A., 2002. Computer Networks.Prentice Hall.