Routing Simulation for Wireless Sensor Networks

Zdravko Monov[1] and Ludmila Karakehayova[2]

Abstract **– This paper describes ARCS – ad-hoc routing and connectivity simulator, a tool that is able to simulate the routing process in distributed sensor networks. After a configuration phase the nodes in the network are ready to route different types of messages. Two versions of the routing algorithm are currently supported. The simulation results produced by ARCS can be used to compare both protocols for a particular deployment. Using the simulator and its graphical module we can assess the capability of the network to route traffic.**

Keywords **– distributed sensor networks, ad-hoc networks, geographic routing.**

I. Introduction

Distributed sensor networks (DSN) are made up of a large number of small sensing nodes which cooperatively perform complex tasks. Environmental monitoring, healthcare, building automation, surveillance and rescue missions are applications where wireless ad-hoc networks provide benefits that we would not otherwise be able to obtain.

The interaction between the nodes is based on wireless communication. Sensor-actuator networks employ sensors to gather information and actuators to perform appropriate actions in a area of interest. Since the energy is a scarce and usually non-renewable resource, the functionality of distributed sensor-actuator networks must be viewed from low-power perspective. Normally, the nodes have a limited radio footprint and packets are forwarded in a multihop manner. When a node receives a packet it applies a routing algorithm to select a neighbor for forwarding. Different criteria can guide the local decision. One approach is to choose the closest to the destination neighbor.

Different protocols have been developed for routing in wireless ad-hoc networks [1]. Dynamic topologies, environmental radio propagation conditions, depletion of batteries and congestion make the experiments difficult and expensive in real-world systems [2]. To better understand the functionality of distributed sensor networks simulations are performed. Various simulators are developed to meet different criteria like energy efficiency, minimizing of bottle-necks or optimal area deployment [6]. They allow complex and precise simulations, while keeping the results reliable for any real world system. Designing a new simulator for a general DSN is a complex task that requires the consideration of many parameters.

[1]Zdravko Monov is with the Faculty of German Engineering Education and Industrial Management, Technical University of Sofia, Bulgaria, E-mail: ztm@tu-sofia.bg

[2]Ludmila Karakehayova is with the Technical University of Sofia, Bulgaria, E-mail: lvk@tu-sofia.bg

II. Hardware model

A typical node is built around a low-power microcontroller. Wireless transceivers create physical links between nodes. Hardware provides the following low-power mechanisms:

- Each node is capable of determining its coordinates,
- The receiver and transmitter can be individually enabled and disabled and
- The transmit power can be adjusted gradually.

III. ARCS overview

Our simulator, ARCS (ad-hoc routing and connectivity simulator), has five major components:

- DSN generator,
- network organization unit,
- route generator,
- routing protocol, and
- computation module.

ARCS' modularity allows easier switching between different protocols and simulation scenarios, while preserving the reliability of the results. The simulator can display the traveled distance for packets, providing valuable information for parameters like energy drain.

The network nodes are randomly distributed in a sensor field. Nodes have a specified wireless cell size. Also, the design entry includes the communication rate, packet data structure, timing and energy parameters. The computation module software is written in the C++ programming language.

Ad-hoc networks are usually homogeneous systems [10]. The internode communication is the largest energy consumer. There are two ways to achieve power-efficient interaction between nodes. First, nodes calculate the distance and tune their transmit power accordingly. Second, nodes stay in a sleeping mode as long as possible. Periodically, nodes wake up and receive the packets buffered for them. Since the internode communications must be synchronized, the nodes operation becomes time constrained [7]. The real-time functionality requires more than just having enough computing power on average - it requires enough power for the worst-case scenario. The network density emerges as the most demanding factor when both variable power levels and synchronization underlie the communication [11].

IV. NETWORK ORGANIZATION AND STRUCTURE

Each node is characterized by its ID and location. The current version of ARCS assumes a static topology. At the beginning of each simulation cycle the nodes are positioned using random coordinates. Deployments could be arranged to match both sparse and dense networks.

The network organization begins with broadcast messages which the nodes transmit after random delays. The messages include the node own ID and location. In addition, each node copies to the message its own database. Fig. 1 shows the network connectivity after a configuration phase. ARCS is able to keep track on the simulation time. By defining the input timing parameters a specific mathematical algorithm is inovked in order to determine the correct behavior of each node.

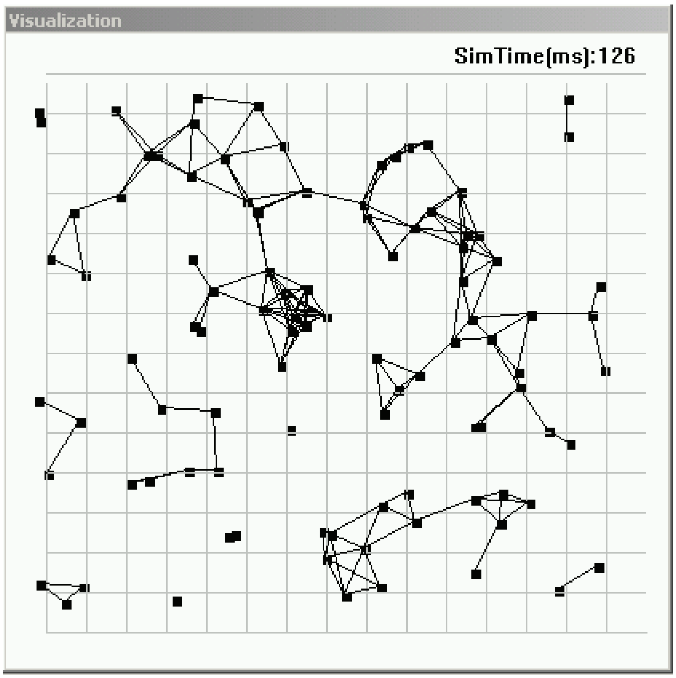

Fig. 1. Network connectivity after a configuration phase

V. ENERGY MODEL

The energy used to send a bit over a distance d may be written as Eq. 1:

$$E = ad^n \tag{1}$$

where a is a proportionality constant [8], [9]. The radio parameter n is a path loss exponent that describes the rate at which the transmitted power decays with increasing distance. Typically, n is between 2 and 4 [8].

$$E = ad^n + b \tag{2}$$

Eq. 2 emerges as a more realistic model. The b constant is associated with specific receivers, CPUs and computational algorithms. The power consumption of a turned on receiver is yet another constant, P_R.

VI. ARCS ROUTING

The route generator randomly selects two nodes as source and destination. The procedure is repeated until the specified number of messages is met. Reaching a dead-end (or void) interrupts the current routing as Fig. 2 presents.

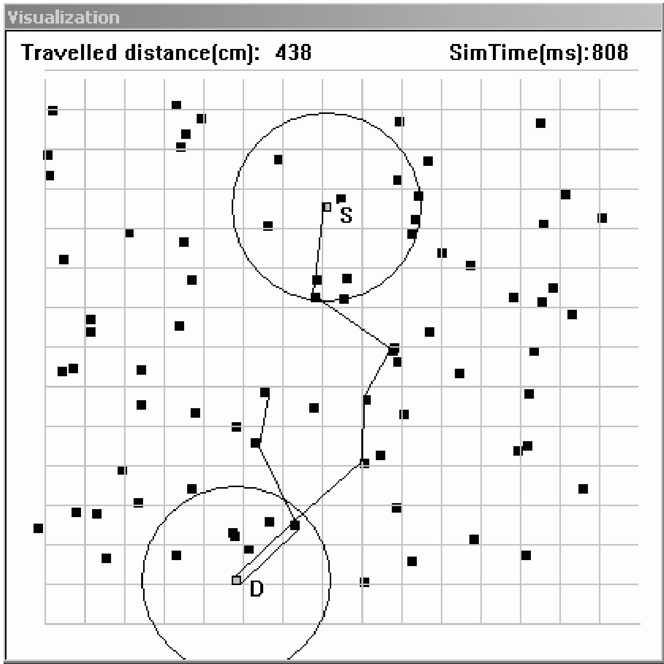

Fig. 2. Unsuccessful routing for the acknowledgement

The simulator is based on two versions of Most Forward within the transmission Range (MFR) routing protocol. MFR assumes that it will be most beneficial to transmit the packet to the closest to the destination node [3]. Fig. 3 indicates the routes when a modification of the MFR protocol is applied. While in the case of geographic routing we may need additional information, the physical position of the participating nodes, on the positive side is the opportunity to forward packets without pre-established routes [4, 5, 12].

The simulation requires only the destination to send an acknowledgement back. Usually, the acknowledgement travels over the same intermediate nodes. ARCS introduces another option – the reverse route is generated by applying MFR again. The difference in routes could be quite significant if the network deployment is reasonably dense. Also the simulation results state a negligible drop in the successful routings, while preserving the nodes from multiple transmissions. Using different routes may be benefical for both power efficiency and security.

The modification of the MFR protocol consists in the different distances that are compared. The next node will be that one, that is farther away from the sender and its position is on the way to the destination.

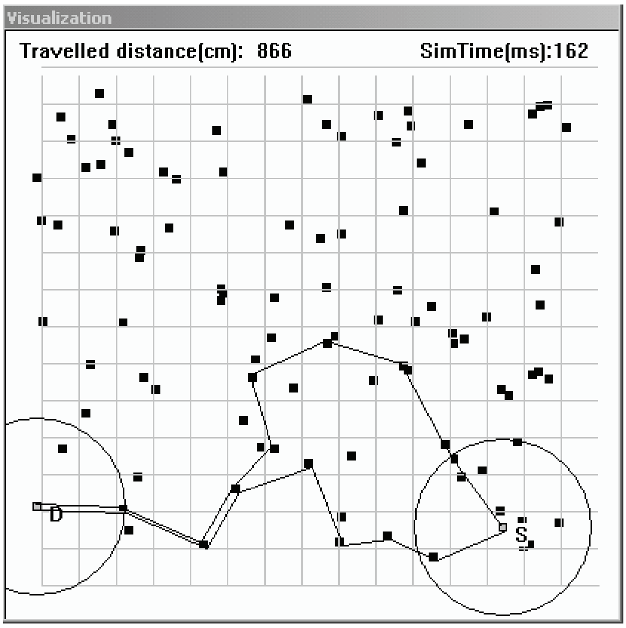

Fig. 3. MFR modification algorithm

For the sake of efficiency, the records in the address table are sorted using a specific criterion – for example, by the distance from the node. The sending node can now easy state which is the farthest node. Then a direction check must be done. If the selected hop doesn't match the corresponding direction, the next node is chosen, and so on. Once a suitable hop is found, the sender transmits the packet.

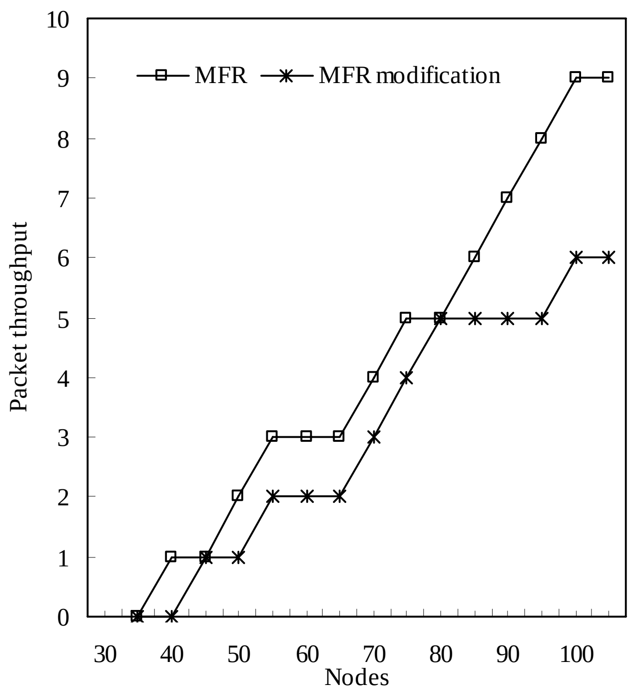

Fig. 4. The fraction of packets delivered for different density of the network

Using ARCS and its graphical presentation of the network connections, we realized that the modification of MFR protocol does not provide the needed efficiency, as Fig. 4 shows [6]. Here, a transmission range of 150 meters is assumed. The area is a square with size 10 km^2, and the baud rate is 1Mb/s. The routes are significantly longer. A longer route means that a larger number of nodes is involved in the routing process. Further more, this leads to increases in the timing parameters – respectively, the packet may not reach the destination in a specific period of time. The general form of the routes generated by the modificated MFR is usually too far from the straight line that connects the initial sender and the final destination.

The transmission range of the nodes also has a significant impact on the network connectivity. While a smaller range may reduce the required energy at overall, it lowers the packet throughput. Fig. 5 shows the results we obtained for different range sizes. We tested the MFR algorithm in terms of routing efficiency. The simulation conditions remain similar to the previous example – area of 10 km^2, transmission speed of 1Mb/s. The nodes were randomly distributed over the whole area. After passing the recognition phase, 10 messages were initiated in a random manner. By increasing the transmission range, more nodes become direct neighbors and therefore more packets are delivered successfully. The downside of a larger radius is that more nodes must wake up to participate during the data transmission. If the signal strength could be adjusted precisely enough, only a small subset of the neighbors would detect the transmission.

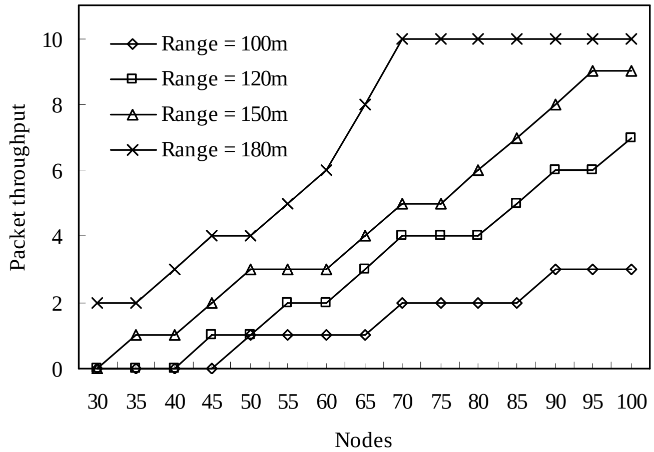

Fig. 5. Routing success rate as a function of the transmission range

VII. Conclusion

In this paper we described ARCS, an ad-hoc routing simulator suitable for wireless sensor networks. We provided simulation results that indicate how the packet throughput scales with the network density and communication range. The results showed that the original MFR protocol provides better connectivity for both sparse and dense networks compared to the modified MFR.

REFERENCES

[1] Hu Y. C. and Perrig, A., A survey of secure wireless ad hoc routing, IEEE Security & Privacy, May/June, 2004, pp. 28-39.

[2] Karakehayov Z., Design of distributed sensor networks for security and defense, In Proceedings of the NATO Advanced Research Workshop on Cyberspace Security and Defense: Research Issues, (Gdansk, Poland, September 6-9, 2004). Springer, 2005, pp. 177-192.

[3] H. Takagi and L. Kleinrock, “Optimal transmission ranges for randomly distributed packet radio terminals”, IEEE Trans. Commun., vol. 32, no.3, March, 1984, pp. 246-257.

[4] Karakehayov Z., "Low-power communication for wireless ad-hoc networks", Proceedings ELECTRONICS'2003 International Conference, Sozopol, 2003, pp. 77-82.

[5] D. Puccinelli and M. Haenggi, Wireless sensor networks: applications and challenges of ubiquitous sensing, IEEE Circuits and Systems Magazine, third quarter 2005, pp. 19-29.

[6] Karakehayov, Z., Monov, Z., Karakehayova, L., ARCS: A simulator for distributed sensor networks, Proceedings of the ELECTRONICS'2005 International Conference, Sozopol, September, 2005, pp. 91-96.

[7] Zdravko Karakehayov and Zdravko Monov, Target-aware timing modelling for wireless ad-hoc networks, Proceedings International Scientific Conference Computer Science’2006, October, 2006, Istanbul, pp. 54-59.

[8] J. L. Gao, Energy efficient routing for wireless sensor networks, Ph.D. dissertation, University of California, Los Angeles, 2000.

[9] J. M. Rabaey, M. J. Ammer, J. L. Silva, D. Patel and S. Roundy, "PicoRadio Supports Ad Hoc Ultra-low Power Wireless Networking", IEEE Computer, vol. 33, pp. 42-48, July 2000.

[10] R. Szewczyk, E. Osterweil, J. Polastre, M. Hamilton, A. Mainwaring and D. Estrin. Habitat monitoring with sensor networks. Communications of the ACM, Special Issue on Sensor Networks, June 2004, Vol. 47, pp. 34-40.

[11] Z. Karakehayov, Design of distributed sensor networks for security and defense, In Proceedings of the NATO Advanced Research Workshop on Cyberspace Security and Defense: Research Issues, (Gdansk, Poland, September 6-9, 2004), edited by J. S. Kowalik, J. Gorski and A. Sachenko, Springer, NATO Science Series II: Mathematics, Physics and Chemistry, Vol. 196, 2005, pp. 177-192.

[12] M. Witt and V. Turau, BGR: Blind geographic routing for sensor networks, In Proceedings of the Third International Workshop on Intelligent Solutions in Embedded Systems, Hamburg, Germany, May, 2005.