Int. J. Nonlinear Anal. Appl. 14 (2023) 1, 2603–2616
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/ijnaa.2022.27744.3696

Parsim: A parametric simulation application for wireless sensor networks based on NS2 simulator

Hossein Ghaffarian, Mahdi Sadeghizadeh*

Department of Computer Engineering, Quchan University of Technology, Quchan, Iran

(Communicated by Javad Vahidi)

Abstract

Wireless sensor networks (WSNs) are attractive technologies that have been highly regarded by researchers in recent years due to their wide applications in various fields. Due to the high cost of implementing wireless sensor networks, one of the most important issues in their use is simulation and extraction of the results to achieve maximum efficiency before implementing them. Therefore, simulation is essential to studying WSN. A large number of WSN simulators available nowadays differ in their design, goals, and characteristics. Of course, there are still challenges in simulating wireless sensor networks, the most important of which are the lack of ease of use and lack of full access to the relevant parameters in the existing cases. In this article, we first introduce the various available simulators for WSNs, together with guidelines for selecting an appropriate simulator. In the following, we present our proposed simulator based on NS2 with the aim of achieving an effective and efficient simulation by considering the ease of use and providing a complete list of relevant parameters.

Keywords: Wireless sensor networks (WSNs), Parametric simulation, Simulators, Network simulator version2 (NS2), Tool command language (TCL), Performance criteria
2020 MSC: 94C60, 00A72

1 Introduction

Wireless Sensor Networks because of their inherent advantages such as lower cost and easier deployment in the environment [1], play different roles in a wide range of applications, such as military surveillance [5], health care [11], target tracking [23], and smart homes [18] are highly desirable and cost-effective.

However, resource constraints, such as limited processing power, memory, and energy are the main challenges in WSN design and application, and subsequently address issues for researchers such as energy efficiency and extending network lifetime [26], enhancing reliability, effective routing [31, 13], and Establish security against attacks [28, 29, 21, 30, 27].

Due to the high cost of implementing WSNs, one of the most important issues in their use is simulation and extraction of the results to achieve maximum efficiency before implementing them. Therefore, simulation is essential to studying WSN. A large number of WSN simulators available nowadays differ in their design, goals, and characteristics

*Corresponding author
Email addresses: ghafarian@qiet.ac.ir (Hossein Ghaffarian), m.sadeghizadeh@qiet.ac.ir (Mahdi Sadeghizadeh)

Received: November 2021 *Accepted:* July 2022

[3, 32, 16, 39]. Of course, there are still challenges in simulating wireless sensor networks, the most important of which are the lack of ease of use and lack of full access to the relevant parameters in the existing cases.

In this article, we first introduce the various available simulators for WSNs, together with guidelines for selecting an appropriate simulator. In the following, we present our proposed simulator based on NS2 with the aim of achieving an effective and efficient simulation by considering the ease of use and providing a complete list of relevant parameters. We actually designed an application to simulate WSNs that allows the user to easily set a complete list of parameters in it. After setting the parameters, by running this application, a TCL code is generated that can be run in the NS2 simulator.

2 Related work

Considering the importance of the issue of simulating wireless sensor networks and its related benefits in reducing costs and increasing the efficiency mentioned in the previous section, a lot of work has been done in this area, which we will review and evaluate in the following.

In [8], a model for simulating WSNs is presented in which the network architecture model and the node structure architecture model are appropriately described. Figure 1 shows the network architecture model in which the sensor nodes sense the events of the environment monitored by the network. Then these nodes transmit the sensed data to the sink node by communicating with each other through the communication medium (radio wave channel). Finally, in the sink node, the main processing is done on the aggregated data and the necessary decisions are made.

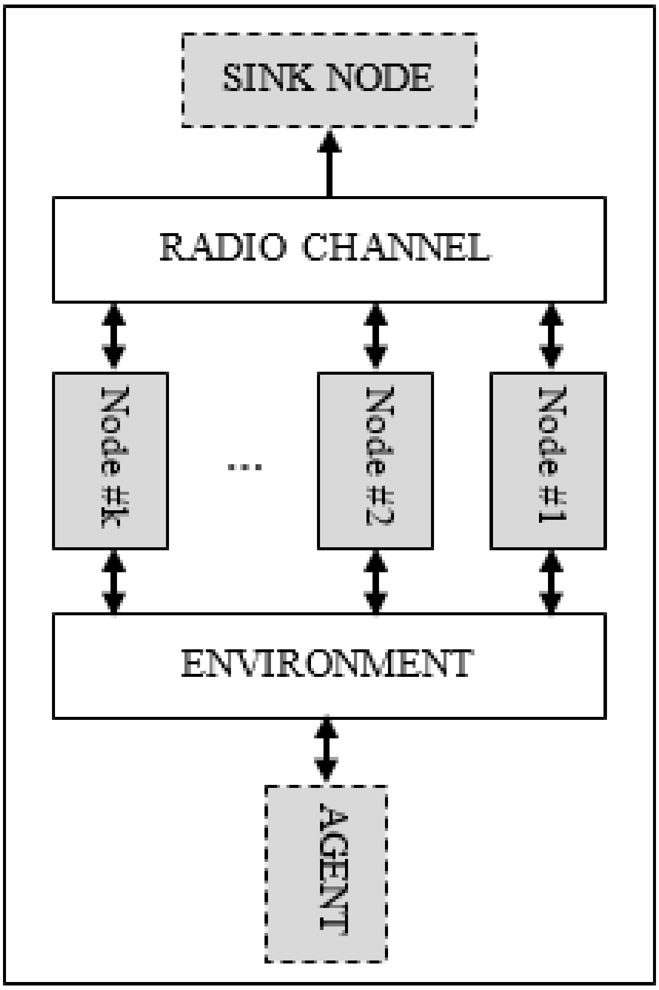

Figure 1: WSN architecture model

Figure 2 also shows the layered architecture model in the design and simulation of sensor nodes. The protocol-tier comprises all communication protocols including media access control (MAC) layer, routing layer and application

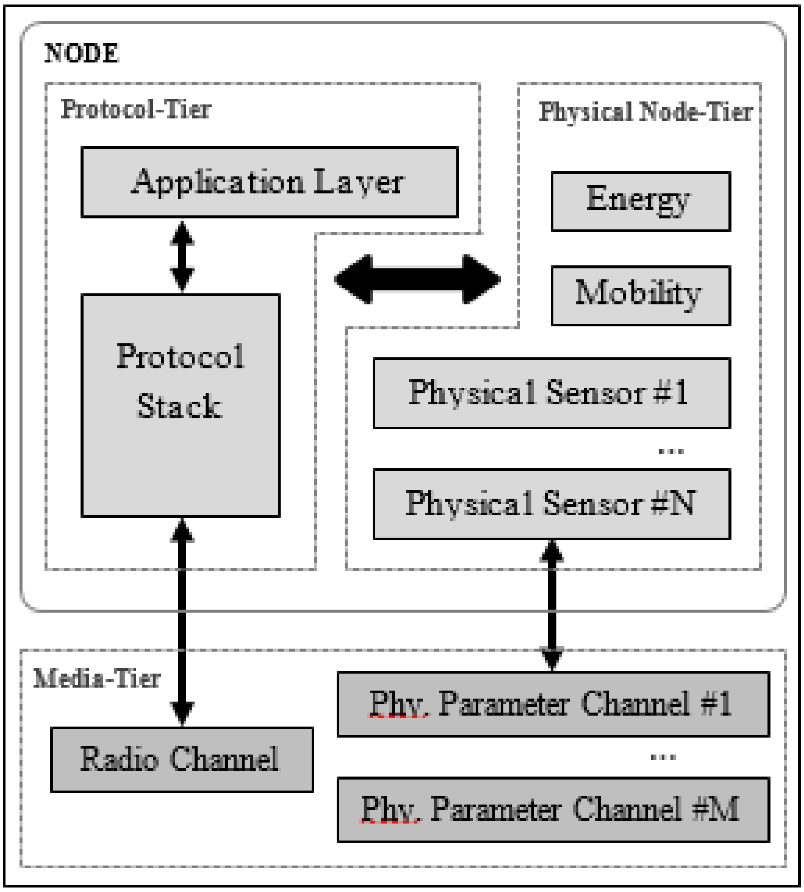

Figure 2: Layered architecture of node model

layer. The physical-tier of the node provides the hardware platform and its effects on equipment performance. The actual composition and structure of this section depends on the relevant application in the wireless sensor network. As can be seen in Figure 2, common components of this layer include physical sensors, the energy module and the mobility module. In the media-tier, the connection of the node with the real environment is also handled.

After determining the network architecture model and its nodes, we need to specify the relevant protocols for each of the assumed layers in wireless sensor networks. Reference [7] provides complete examples of these protocols. These protocols will be selected for the relevant network depending on the type of network application and how the sensor nodes interact with the surrounding environment. In fact, to properly and dynamically simulate a wireless sensor network, we must consider all the necessary parameters mentioned above.

In the continuation of this section, we will introduce and review the existing simulators for simulating wireless sensor networks.

2.1 WSN simulators

Network Simulator 2 (NS-2) [40]

NS2 is one of the most popular network simulators that is widely used in research laboratories to simulate and evaluate the performance of network protocols. This simulator was developed in 1989 based on C++ with the object-oriented discrete event technique. NS2 uses the Object-oriented Tool Command Language (OTcl) to configure the simulation of a given network by assembling and configuring the objects as well as scheduling discrete events. In fact, this simulator, with the support of a wide range of communication protocols at all layers, allows us to test, analyze and evaluate network applications before their practical implementation in the real environment. In addition to the above, special protocols for WSNs such as directed diffusion or SMAC are also provided which allows it to be simulated [2]. However, the use of ns-2 to simulate WSNs is limited due to its scalability and lack of an application model [7]. NS-2 uses Network Animator (NAM) as a graphical visualization tool, which can just reproduce an NS-2 trace. After the trace file as simulation output is created, Scripting languages such as AWK (Aho Weinberger Kernighan) script and PERL script can be used to calculate the performance metrics such as throughput, and packet loss, and delay.

OPNET [41, 12]

OPNET Modeler is a discrete event, object-oriented, general purpose network simulator. Modeler was introduced in 1987 as the first commercial network simulator. Originally, the software was developed for military purposes, but it has grown to be the world's leading commercial network simulation and modeling tool. OPNET is a large and powerful software with a wide variety of possibilities. OPNET can be used as a research tool and also as a network design/analysis tool. OPNET was originally built for the simulation of fixed networks, and therefore, it contains extensive libraries of accurate models from commercially available fixed network hardware and protocols. Recent versions also include wide possibilities for wireless network simulations including support for Zigbee-compatible 802.15.4 MAC. The strength of OPNET in wireless network simulations is the accurate modeling of the radio transmission. Different characteristics of physical-link transceivers, antennas, and antenna patterns are modeled in detail. With the Wireless suite for Defense extension, OPNET can model 3D outdoor scenarios and take into account different kinds of obstacles like terrain shape and buildings. OPNET can also be used to define custom packet formats. A weak point is that there exist only a few ready models for recent wireless systems.

J-Sim [14, 35]

A component-based simulation environment developed entirely in Java. It provides real-time process-based simulation. The main benefit of J-sim is its considerable list of supported protocols, including a WSN simulation framework with a very detailed model of WSNs, and implementation of localization, routing, and data diffusion WSN algorithms. J-sim models are easily reusable and interchangeable offering the maximum flexibility. Additionally, it provides a GUI library for animation, tracing, and debugging support and a script interface, named Jacl. J-Sim claims to scale to a similar number of wireless nodes than NS-2 (around 500) with two orders of magnitude better memory consumption but a 41% worse execution time.

GloMoSim [10]

GloMoSim is a scalable simulation environment for wireless and wired network systems, which uses the parallel discrete-event simulation capability provided by Parsec [42], a cbased simulation language for sequential and parallel execution of discrete-event simulation models. Both, GloMoSim as well as Parsec, were developed by the Parallel Computing Lab. at UCLA. GloMoSim offers basic functionality to simulate wireless networks, even for ad hoc networks (e.g., AODV, DSR). Several proposals for WSN protocols have been tested with it. Recently, a development kit for WSN has been released, sQualnet [36].

TOSSIM [17]

TOSSIM is a Bit-level discrete event simulator and emulator for TinyOS operating system, and it was developed at UC Berkeley. It simulates the behavior of a sensor (sending/receiving messages via radio waves and processing information). This simulator is written in NesC Language which provides a component-based programming model. TOSSIM simulates the TinyOS network stack at the bit level, allowing experimentation with low-level protocols in addition to high-level application systems. TOSSIM is based on the assumption that each node in the network must run exactly the same code, which makes it less flexible. TOSSIM does not model energy consumption, so PowerTOSSIM Simulator and PowerTOSSIM z are an improvement that extends the simulator to model energy consumption [24].

SENS [37]

SENS is a simulation tool consisting of components for the physical environment, network communication, and applications. Module implementations are based on data and characteristics obtained from real sensor networks. Although power utilization analysis is supported, phenomena detection capabilities are limited only to sound. The environment component yields models of various types of surfaces that influence the radio and sound propagation parameters. The simulator provides network models, responsible for packet losses, collisions, and propagation delays. NesC code can be used directly on it.

Other Simulator

In addition to the above, many other simulators have been presented for WSNs, each of which has been created for a specific purpose and case study, the most important of which is mentioned for the knowledge of researchers:

- JiST/SWANS [6]
- Ptolemy II [4]

- EmStar [9]
- ATEMU [25]
- Prowler [33]
- NetTopo [34]
- SENSE [38]
- Shawn [15]
- TRMSim-WSN [20]
- WsnSimPy [19]
- COOJA [22]

3 Proposed application for WSNs simulation

Given the wide applications of WSNs in various fields, as well as the high cost of its implementation in the environment, it is very important to provide a suitable simulation platform for it. In fact, by providing a suitable simulation platform, the necessary studies can be performed on the WSNs without wasting the relevant costs, and by evaluating the results of the simulations, the desired improvements and corrections can be applied to it, and finally based on its achievements implemented an effective WSN with maximum efficiency. As mentioned in the previous section, various simulators have been designed for WSNs, each of which is designed to cover specific applications. However, a general simulator is still needed with the ability to provide a complete list of parameters to set the assumed network that can be used easily and without the need for specialized knowledge by individuals. In fact, our main goal in presenting this article is to provide a simple and complete tool for effective simulation of WSNs in the shortest possible time so that it is possible to change and correct its parameters based on performance needs. This simulation tool is designed in the form of an application in which the possibility of selecting and setting a complete list of parameters for simulating a WSN in the simplest possible way is provided. Some of these parameters are dedicated to the selection of relevant protocols that in the simulation of the assumed WSN should be used in different layers. These protocols are presented separately for the different layers in Figure 3.

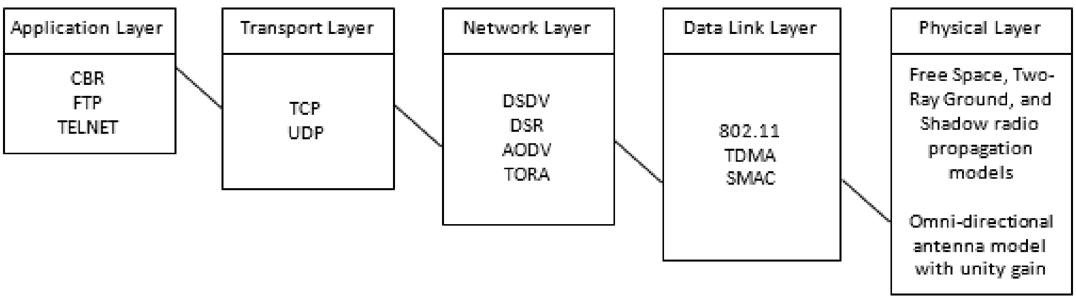

Figure 3: Protocols for different layers in wireless sensor networks

In the proposed application, as shown in Figures 4 to 6, the parameters related to network topology setting, physical layer and WSN nodes settings are presented, respectively, and the user can easily select them without specialized knowledge to configure the desired WSN network.

The last component of the proposed application is the possibility of defining and adjusting parameters and performance evaluation criteria so that the user can evaluate the assumed criteria on the simulated network according to the required performance criteria. As shown in Figure 7, these performance metrics are as follows:

- **Average Energy Consumption:** This criterion shows the average energy consumption in network nodes.

$$Average\ Energy\ Consumption = \frac{\sum_{i=1}^{nodes} Initial\ Energy_i - Residal\ Energy_i}{No.of\ nodes} \quad (1)$$

- **End-to-End Delay:** This criterion is the time it takes to send a packet over the network from the source to the destination.
- **Network Throughput:** This criterion expresses the amount of data received in the entire network in the unit of time and is calculated by the following formula.

$$Throughput = \sum_{f=1}^{MaxFlow} \frac{No.of\ received\ packets * Packet\ size * 8}{flow\ time} \tag{2}$$

- **Traffic rate:** In order to more accurately examine the stability of the network, in addition to the transmission delay and throughput, it is better to use the traffic rate criterion. This criterion expresses the amount of data sent in the entire network in the unit of time.

$$TrafficRate = \sum_{f=1}^{MaxFlow} \frac{No.of\ sent\ packets * Packet\ size * 8}{flow\ time} \tag{3}$$

- **Packet Delivery Ratio (PDR):** This criterion specifies the amount of data received relative to the data transmitted over the entire network.

$$Packet\ Delivery\ Ratio = \frac{No.of\ received\ packets}{No.of\ sent\ packets} * 100 \tag{4}$$

- **Routing Overhead (RO):** This criterion describes the amount of overhead caused by routing data between sensor nodes in the network.

$$Normalized\ Routing\ Overhead = \frac{No.of\ Routing\ packets}{No.of\ Received\ packets} \tag{5}$$

- **Packet Loss Ratio:** This criterion determines the percentage of packets removed relative to send packets.

$$Packet\ Loss\ Ratio = \frac{Packet\ Loss}{No.of\ sent\ packets} * 100 \tag{6}$$

After setting and selecting the various parameters to simulate the desired WSN, which are presented in the components of the proposed application, and finally its execution, the following outputs are created:

- **Output related to WSN network scenario (scenario.Tcl file):** This output, which is a Tcl file, defines simulation scenario of the assumed WSN network according to the set parameters in proposed Application. By running this file on the NS2 simulator, a trice file is created that presents all the events related to the simulation of the assumed WSN network in a specific format.
- **Output related to WSN performance evaluation (Performance-Eval.awk):** This output, which is actually a code in GAWK language, is used based on the performance evaluation parameters specified in the proposed application (tab performance evaluation) to generate reports and extract the relevant results in the simulated WSN. By executing this code on the trice file created in the previous step, all the data needed to plot diagrams related to the evaluation criteria specified in the proposed application are extracted. Figure 8 shows the complete WSN simulation process described in the steps above.

As shown in Figure 8, the NAM tool in the NS2 simulator provides a visual interpretation of the assumed WSN topology with its animated display at runtime.

4 Simulation and results

In order to evaluate the proposed application, we have provided a sample simulation in this section. In this simulation, the basic network parameters are determined according to the nature of WSNs, existing requirements and the usual applications of these networks. We also presented two various scenarios in the simulation. In the first scenario, 60 nodes in 4 clusters and in the second scenario, 100 nodes in 5 clusters are considered. In both scenarios

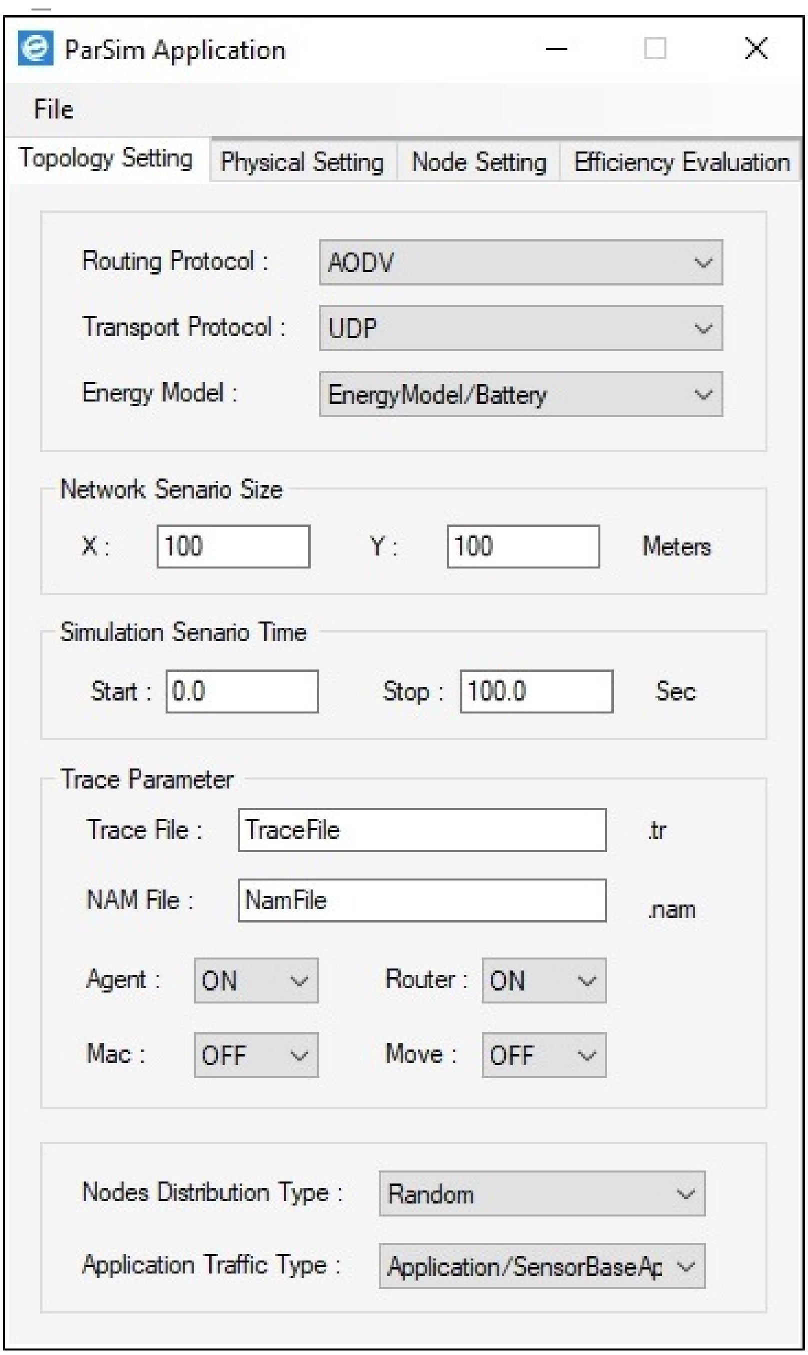

Figure 4: Parameters related to WSN topology setting

size of network is an area of 100 * 100 m2 with CBR traffic and packet size of 70 bytes. The simulation parameters used in our simulation model are summarized in the Table 1.

Figure 9 shows a visual interpretation of the simulated WSN topology based on parameters in table 1 at the runtime along with its animated display that is provided by the NAM tool available in the NS2 simulator. As shown in Figure 9, the dark blue nodes represent the Cluster head nodes and the black nodes represent the base station. Also, nodes of the same color have been used to show the normal nodes of each cluster. In the simulations for the distribution of WSN nodes in the environment, we presented and evaluated both random distribution and manual distribution methods. Figure 9 is an example of a manual distribution that has a more uniform distribution density in the environment than the random distribution.

The results of the simulations performed based on the performance evaluation criteria introduced in the previous

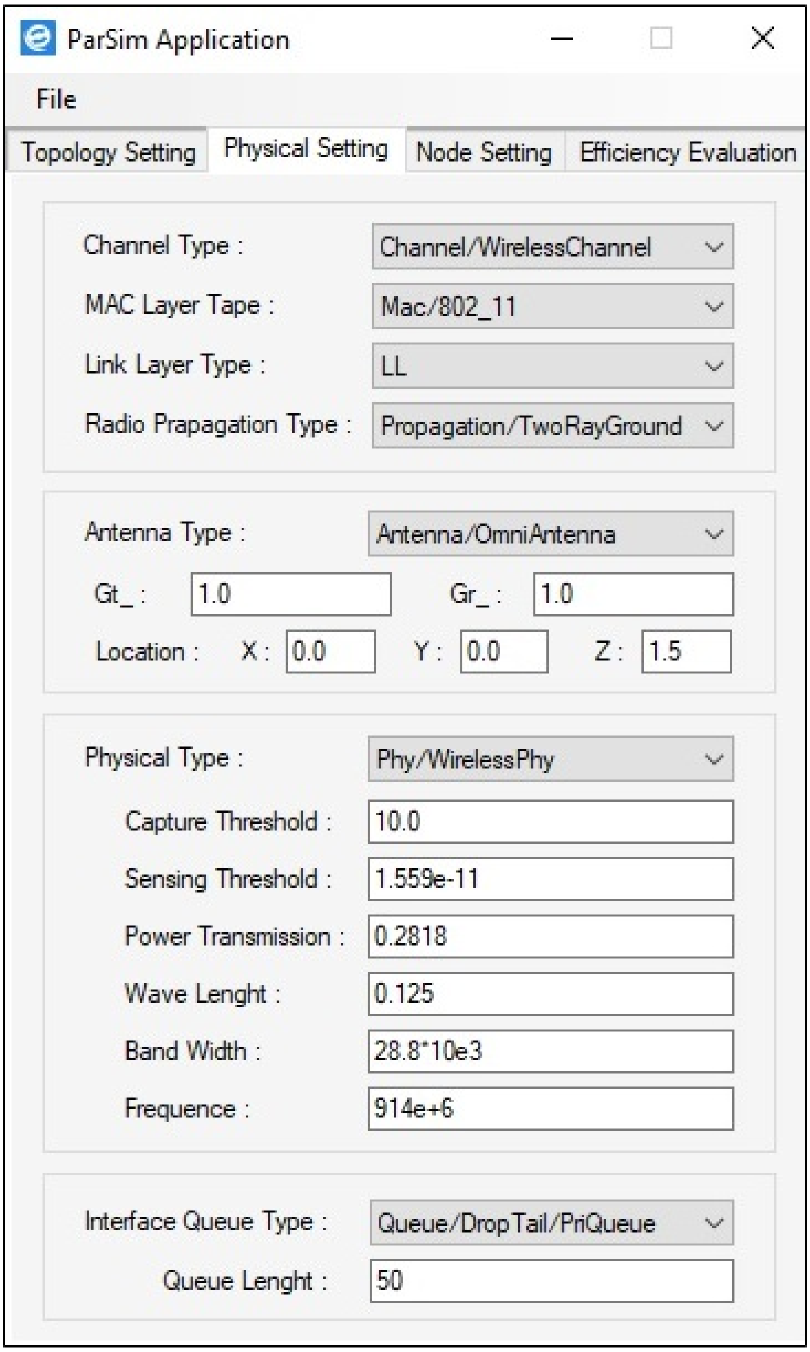

Figure 5: Parameters related to WSN physical layer

section are presented in Table 2. As mentioned in section 3, after creating the scenario file (scenario. Tcl) by the proposed application and executing it in the NS2 simulator, a trice file containing all the events related to the simulation is created. Finally, by executing the performance evaluation file (PerformanceEval.awk), which is also created by the proposed application, on the trice file, the relevant results Are extracted and presented in Table 2.

5 Conclusion and future work

In this article, considering the wide applications of sensor networks and in order to reduce costs, we have proposed an effective application that allows simulation of sensor networks with high ease. In this application, a complete list of parameters for setting the assumed WSN is provided to the user, which allows the user to simulate the desired

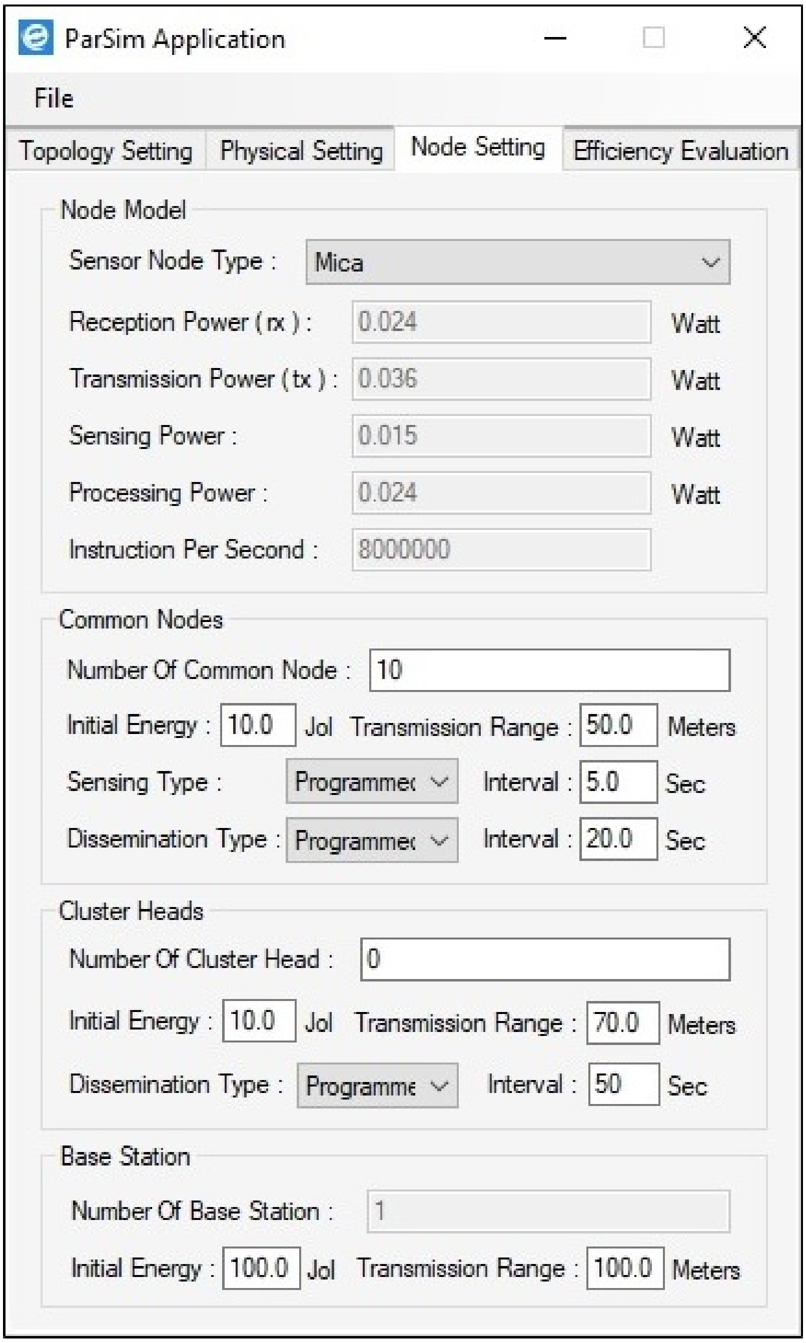

Figure 6: Parameters related to WSN nodes settings

WSN without the need for specialized knowledge. The proposed application also includes a component to evaluate the performance of the simulated WSN, based on which users can easily improve their WSN. Finally, in order to evaluate the proposed application, we simulated scenarios with it and presented its results. In the future, by upgrading the proposed application, we intend to add a component to study the behavior of various security attacks in WSNs. Our main purpose in doing this is to analyze the behavior of security attacks and provide solutions to improve security in WSNs based on.

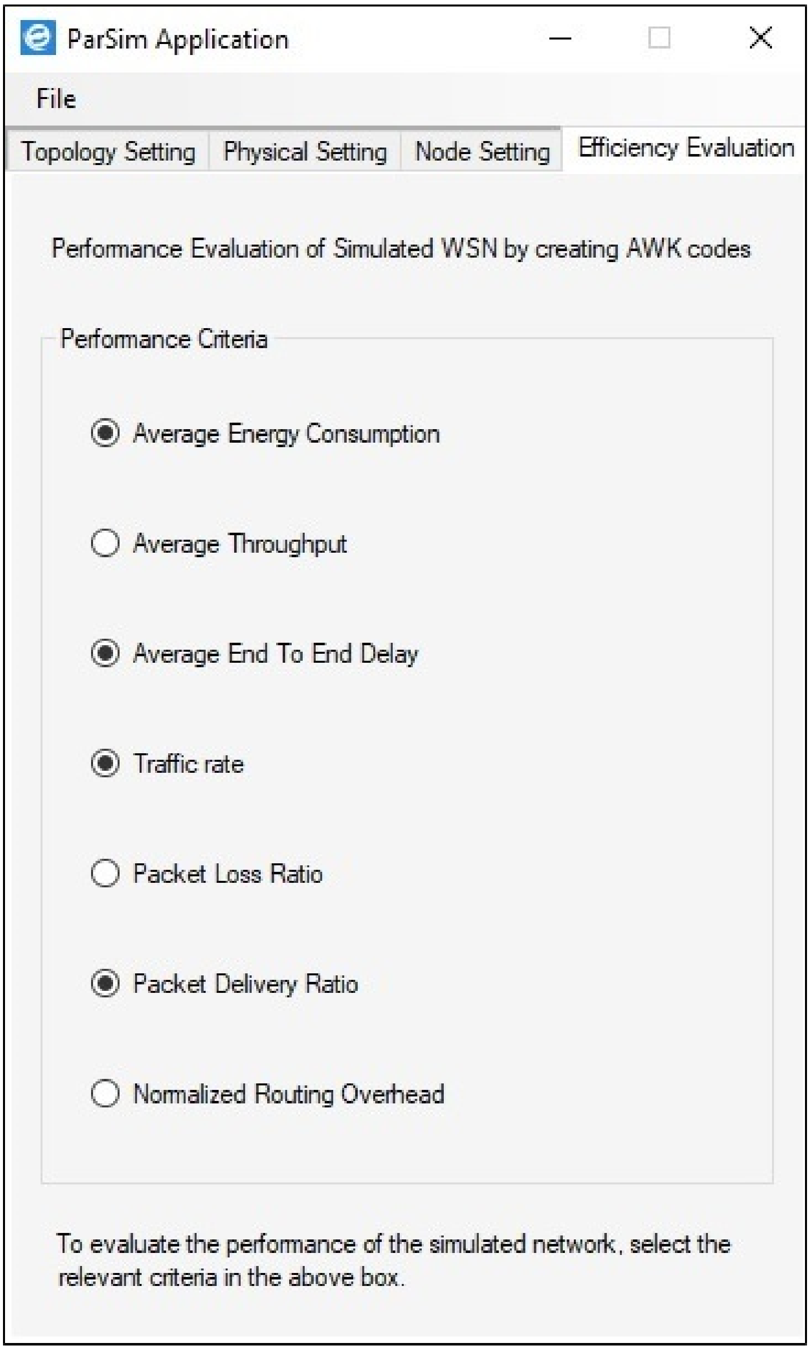

Figure 7: Parameters related to performance evaluation criteria

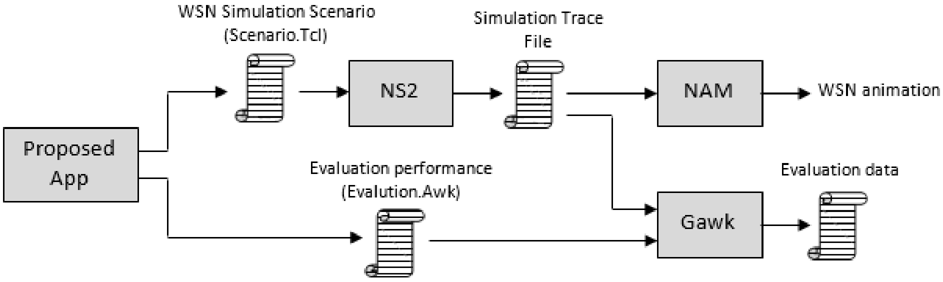

Figure 8: WSN simulation process with proposed application.

Table 1: Wireless sensor network simulation parameters

No	Parameters	Values
1	Number of nodes	60/100
2	Size of network	100 * 100 m^2
3	Routing protocol	AODV
4	MAC protocol	802.11
5	Link Layer protocol	LL
6	Type of traffic	CBR
7	Packet size	70 byte
8	Clustering method	Static / Dynamic (LEACH)
9	Number of Cluster	2 / 3 / 4 / 5
10	Antenna Model	Omni Antenna
11	Interface Queue Type	Drop Tail
12	Queue Length	50
13	Simulation Time	100 sec
14	Type of nodes	Mica2
15	Sensing Power	0.015 w
16	Processing Power	0.024 w
17	Sleep Power	0.0001 w
18	RX Power	0.024 w
19	TX Power	0.036 w
20	Energy Model	Battery
21	Initial Energy of nodes	1 Joule
22	Channel Type	Wireless Channel
23	Radio Propagation Model	Two Ray Ground

Table 2: WSN simulation results based on the performance evaluation criteria

Scenario	Nodes	Clusters	Packets Sent	Packets Received	Packets Dropped	End to End Delay	Packet Delivery Ratio	Throughput (Kbps)	Traffic Rate (Kbps)	Routing Overhead	Average Energy Consume
Scenario 1	60	4	7396	7371	25	0.026	99.66	41.29	41.43	0.365	0.756
Scenario 2	100	5	11197	11075	121	0.032	98.91	62.04	62.72	0.712	0.738

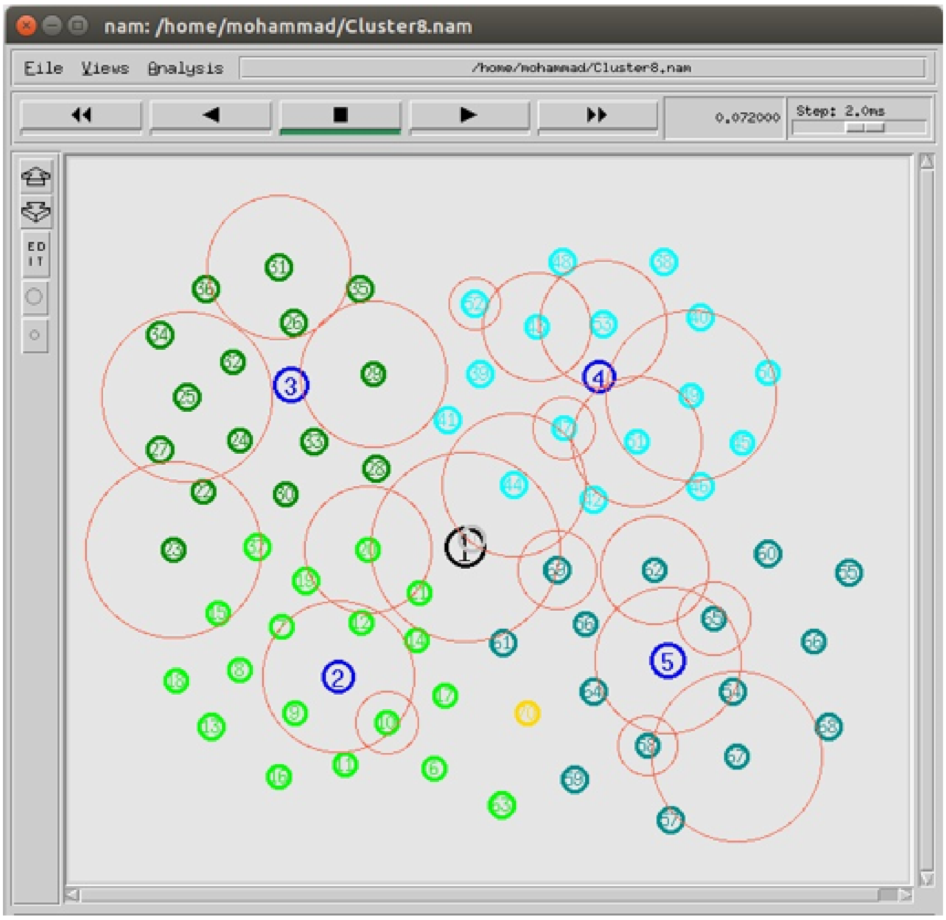

Figure 9: WSN animated display from the NAM tool

References

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, *A survey on sensor networks*, IEEE Commun. Magazine **40** (2002), no. 8, 102–114.

[2] B.I. Bakare and J.D. Enoch, *A review of simulation techniques for some wireless communication system*, Int. J. Electron. Commun. Comput. Engin. **10** (2019), no. 2, 60–70.

[3] M. Bakni, L. Manuel, M. Chacón, Y. Cardinale and G. Terrasson, et al., *WSN Simulators Evaluation: An Approach Focusing on Energy awareness*, Int. J. Wireless Mobile Networks (IJWMN). **11** (2020), no. 6, 1–20.

[4] P. Baldwin, S. Kohli, E.A. Lee, X. Liu and Y. Zhao, *Modeling of sensor nets in ptolemy II*, Proc. Inf. Process. Sensor Networks (IPSN), Berkeley, 2004, pp. 359–368.

[5] M.G. Ball, B. Qela and S. Wesolkowski, *A review of the use of computational intelligence in the design of military surveillance networks*, Recent Adv. Comput. Intell. Defense Secur. **621** (2015), 663–693.

[6] R. Barr, Z.J. Haas and R. Van Renesse, *JiST: embedding simulation time into a virtual machine*, Proc. 5th EUROSIM Cong. Model. Simul., Paris, France, Sep. 2004.

[7] I.T. Downard, *Simulating sensor networks in NS-2*, Technical Report NRL/FR/5522-04-10073, Naval Research Lab, Washington DC, U.S.A., May 2004.

[8] E. Egea-Lopez, J. Vales-Alonso, A.S. Martinez-Sala, P. Pavon-Marino and J. Garcia-Haro, *Simulation tools for wireless sensor networks*, Procedia Summer Simulation Multiconference - SPECTS. 2005.

[9] L. Girod, T. Stathopoulos, N. Ramanathan, J. Elson, D. Estrin, E. Osterweil and T. Schoellhammer, *A system for simulation, emulation and deployment of heterogeneous sensor networks*, Proc. 2nd ACM Int. Conf. Embedded Networked Sensor Syst.(SenSys), Baltimore, MD, 2004, pp. 201–213.

[10] *GloMoSim*, Available: http://pcl.cs.ucla.edu/projects/glomosim.

[11] D. He, N. Kumar, J. Chen, C.C. Lee and N. Chilamkurti, *Robust anonymous authentication protocol for health-care applications using wireless medical sensor networks*, Multimedia Syst. **21** (2015), no. 1, 49–60.

[12] H. Jiang, P. Wang and H. Liu, *Research on oPNET simulation model in wireless sensor networks*, Jisuanji Gongcheng/ Comput. Engin. **33** (2007), no. 4.

[13] A. Jiang and L. Zheng, *An effective hybrid routing algorithm in WSN: ant colony optimization in combination with Hop count minimization*, Sensors **18**, no. 4, 1020–1020.

[14] *J-Sim*, Available: http://www.j-sim.org.

[15] A. Kr¨oller, D. Pfisterer, C. Buschmann, S.P. Fekete and S. Fischer, *Shawn: A new approach to simulating wireless sensor networks*, Proc. DASD'05, Design, Anal. Simul. Distributed Syst., (San Diego, USA), Apr. 2005.

[16] M. Korkalainen, M. Sallinen, N. Kärkkäinen and P. Tukeva, *Survey of wireless sensor networks simulation tools for demanding applications*, Fifth Int. Conf. Network. Serv., 2009, pp. 102–106.

[17] P. Levis, N. Lee, M. Welsg and D. Culler, *TOSSIM: accurate and scalable simulation of entire tinyOS applications*, Proc. 1st ACM Int. Conf. Embedded Networked Sensor Syst. (SenSys), Los Angeles, CA, 2003, pp. 126–137.

[18] M. Li and H.J. Lin, *Design and implementation of smart home control systems based on wireless sensor networks and power line communications*, IEEE Trans. Ind. Electron. **62** (2015), no. 7, 4430–4442.

[19] A. Marchiori, L. Guo, J. Thomas and Q. Han, *Realistic performance analysis of WSN protocols through trace based simulation*, Seventh ACM Perform. Eval. Wireless Ad Hoc, Sensor, and Ubiquitous Networks Conference (PE-WASUN), (Bodrum, Turkey), Oct, 2010.

[20] F.G. Marmol and G.M. Perez, *TRMSim-WSN, trust and reputation models simulator for wireless sensor networks*, Proc. IEEE ICC 2009, IEEE Int. Conf. Commun. (Dresden, Germany), Jun, 2009.

[21] M.A. Moulavi, J. Nasiri, B. Bahmani, H. Parvar, M. Sadeghizadeh and M. Naghibzadeh, *DHA-KD: Dynamic hierarchical agent based key distribution in group communication*, Ninth ACIS Int. Conf. Software Engin. Artif. Intell. Network. Parallel/Distributed Comput., 2008, pp. 301-306,

[22] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne and T. Voigt, *Cross-level sensor network simulation with COOJA*,

Proc. SenseApp, First IEEE Int. Workshop Practical Issues in Build. Sensor Network Appl., (Tampa, USA), Nov. 2006.

[23] C. Pang, G. Xu, G. Shan and Y. Zhang, *A new energy efficient management approach for wireless sensor networks in target tracking*, Defence Technol. **17** (2021), no. 3, 932–947.

[24] E. Perla, A.O. Cathain, R.S. Carbajo, M. Huggard and C. Mc Goldrick, *PowerTOSSIM z: realistic energy modelling for wireless sensor network environments*, Proc. 3nd ACM Workshop Perform. Monitor. Measure. Heterogen. Wireless Wired Networks, ACM, 2008, pp. 35–42.

[25] J. Polley, D. Blazakis, J. McGee, D. Rusk, J.S. Baras and M. Karir, *ATEMU: A fine-grained sensor network simulator*, Proc. 1st IEEE Int. Conf. Sensor Adhoc Communication Networks (SECON'04), Santa Clara, CA, Oct. 2004.

[26] A. Rodríguez, C. Del-Valle-Soto and R. Velázquez, *Energy-efficient clustering routing protocol for wireless sensor networks based on yellow saddle goatfish algorithm*, Math. **8** (2020), no. 9, 1515–1515.

[27] M. Sadeghizadeh and O.R. Marouzi, *A lightweight intrusion detection system based on two-level trust for wireless sensor networks*, Iran. J. Electric. Comput. Engin. **17** (2019), no. 1, 47–56.

[28] M. Sadeghizadeh and O.R. Marouzi, *A lightweight intrusion detection system based on specifications to improve security in wireless sensor networks*, J. Commun. Engin. **7** (2018), no. 2, 29–60.

[29] M. Sadeghizadeh and O.R. Marouzi, *Securing cluster-heads in wireless sensor networks by a hybrid intrusion detection system based on data mining*, J. Commun. Engin. **8** (2019), no. 1, 1–19.

[30] M. Sadeghizadeh, *A lightweight intrusion detection system based on RSSI for sybil attack detection in wireless sensor networks*, Int. J. Nonlinear Anal. Appl. **13** (2022), no. 1, 305–320.

[31] A. Sarkar and T. Senthil Murugan, *Routing protocols for wireless sensor networks: What the literature says?*, Alexandria Engin. J. **55** (2016), no. 4, 3173–3183.

[32] S. Silmi, Z. Doukha, R. Kemcha and S. Moussaoui, *Wireless sensor networks simulators and testbeds*, Comput. Sci. Inf. Technolo. **10** (2020), no. 9, 141–159.

[33] G. Simon, P. Volgyesi, M. Maroti and A. Ledeczi, *Simulation-based optimization of communication protocols for large-scale wireless sensor networks*, Proc. IEEE Int. Aerospace Conf. **3** (2003), pp. 1339–1346.

[34] L. Shu, C. Wu, and M. Hauswirth, *NetTopo: Beyond simulator and visualizor for wireless sensor networks*, Tech. Nep., Digital Enterprise Research Institute (DERI), Jul. 2008.

[35] A. Sobeih, J.C. Hou, L.C. Kung, N. Li, H. Zhang, W.P. Chen, H.Y. Tyan and H. Lim, *J-Sim: a simulation and emulation environment for wireless sensor networks*, IEEE Wireless Commun. **13** (2006), no. 4, 104–119.

[36] *SQualnet*, Available: htpp://nesl.ee.ucla.edu/projects/squalnet.

[37] S. Sundresh, W. Kim and G. Agha, *SENS: A sensor, environment and network simulator*, Proc. 37th ACM Ann. Symp. Simul., Washington, DC, 2004, pp. 221.

[38] B.K. Szymanski and G.G. Chen, *Sensor network component based simulator*, Handbook of Dynamic System Modeling (P. Fishwick, ed.), 2007.

[39] A.S. Tonneau, N. Mitton and J. Vandaele, *How to choose an experimentation platform for wireless sensor networks? A survey on static and mobile wireless sensor network experimentation facilities*, Ad Hoc Networks **30** (2015), 115–127.

[40] *The network simulator - ns2*, [Online]. Available: www.isi.edu/nsnam/ns, 2009.

[41] *OPNET technologies*, Available: http://www.opnet.com/, 2012.

[42] PARSEC: *Parallel simulation environment for complex systems*, Available: http://pcl.cs.ucla.edu/projects/parsec.