Simulation Modelling Practice and Theory 69 (2016) 55–79

Contents lists available at ScienceDirect

Simulation Modelling Practice and Theory

journal homepage: www.elsevier.com/locate/simpat

A Graphical-based educational simulation tool for Wireless Sensor Networks

Nabil Sabor[a,b,*], Shigenobu Sasaki[a], Mohammed Abo-Zahhad[b], Sabah M. Ahmed[b]

[a] Department of Electrical and Electronic Engineering, Niigata University, Niigata 950-2181, Japan
[b] Electrical and Electronics Engineering Department, Faculty of Engineering, Assiut University, Assiut 71516, Egypt

ARTICLE INFO

Article history:
Received 19 June 2016
Revised 6 August 2016
Accepted 9 September 2016

Keywords:
Wireless Sensor Networks
Routing protocols
Genetic algorithm
Immune algorithm
Open source software
Command-line
Graphic
Educational software

ABSTRACT

Many routing protocols have been developed to improve the lifetime, bandwidth reusability and scalability of the Wireless Sensor Networks (WSNs). The operation of routing protocols is difficult to understand and some problems may occur while developing these protocols. Simulation is a relatively fast way of estimating these protocols and understating what is happening in the network. Thus, this paper presents an open source Graphical-based educational simulation tool called Gbest-WSN for simulating routing protocols of the static and mobile, homogeneous and heterogeneous WSNs. Gbest-WSN tool has a user-friendly interface that helps the user to select the routing protocol and define the network configuration. It is provided with four routing protocols; namely LEACH, LEACH-Mobile, immune algorithm-based and genetic algorithm-based routing protocols. Also, it allows the user to update the existing routing protocols and add a new routing protocol. Gbest-WSN is provided with radio, coverage and mobility models for modeling the hardware of the sensor node. It shows a detailed 2D and 3D graphical perception for what is happing during the routing process. Also, it has the ability to compare the simulation results of different simulation methods or different network configurations. In addition, it allows the user to save and load simulation scenarios and also exports the graphical results on PDF files and the statistical results on excel or mat files. Moreover, Gbest-WSN is provided with html help documents to help the user how to use it. The illustrative simulation examples clarified that the Gbest-WSN is a helpful tool for the students, teachers and researchers who work in the field of WSNs.

1. Introduction

A Wireless Sensor Network (WSN) consists of up to thousands of small size devices called the sensor nodes. Sensor node is a self-powered device with capabilities of sensing, processing and wirelessly communicating. The technology of WSN was developed for military applications, but now it widely used in many fields such as the medical and healthcare applications, the environment and habitat applications, the industrial and civilian applications, the automation applications and the control application [1–3]. Due to the limited power and non-rechargeable battery options, the energy resources of

* Corresponding author at: Department of Electrical and Electronic Engineering, Niigata University, Niigata 950-2181, Japan. Tel.: +81 08092746344.
E-mail addresses: nabil_sabor@aun.edu.eg (N. Sabor), shinsasaki@ieee.org (S. Sasaki), zahhad@yahoo.com (M. Abo-Zahhad), sabahma@yahoo.com (S.M. Ahmed).

http://dx.doi.org/10.1016/j.simpat.2016.09.004
1569-190X/

the sensor nodes should be handled wisely to extend their lifespan. So, many routing protocols [4–20] have been developed to improve the lifetime, the bandwidth reusability and the scalability of WSNs. These routing protocols operate in rounds manner, where each round consists of a setup phase and a steady-state phase. In the setup phase, the network is partitioned into equal or unequal sized clusters with selecting a best Cluster Head (CH) for each one and concurrently, single-hop or multi-hop routes from CHs to sink are constructed. While in the steady-state phase, data of the sensor nodes are transferred through an adaptive or a fixed number of frames to the sink via the constructed routes.

The operation of routing protocols is difficult to understand and some problems may occur while developing these protocols. These problems may not be identifiable without a way to actually see what is happening in the network. Also, the validation and measuring the performance of routing protocols in the real testbed is costly and time consuming. So, simulation is a relatively fast way of estimating the routing protocols of WSNs before applying them practically. The command-line simulations are complicated and their printouts should be analyzed to follow the process of routing and draw a whole perception of what happened in the network. Therefore, these simulations are non-attractive for naive users who search for the easier simulation tool. Graphics make the routing process easier to understand and teach how WSN operates and how its protocols really work.

Thus, this paper proposes a Graphical-based educational simulation tool (Gbest-WSN) for monitoring the operation of routing protocols and comparing between them. Gbest-WSN tool is built based on Matlab environment. MATLAB [21], developed by MathWorks Inc., is a software package for high-performance numerical computation and visualization. The combination of analysis capabilities, flexibility, reliability, and powerful graphics make MATLAB the premier software package for scientific researchers. Gbest-WSN tool has a friendly graphical user interface that helps the user to define the network configuration, choose the routing protocol and select the radio, coverage and mobility models of the sensor nodes. The Gbest-WSN tool is provided with four routing protocols; namely are LEACH [4], LEACH-Mobile [5], genetic algorithm-based [10–12,22,23] and immune algorithm-based [13–15,24] routing protocols. Gbset-WSN tool is an open source software which allows the user to update the existing routing protocols and add a new routing protocol. Also, it has the ability to compare between the simulated results of different simulation methods or different network configurations. It shows a 2D and 3D graphical perception for what is happing in the network during the routing and clustering processes. In addition, it exports the obtained statistical results on excel or mat files and the obtained graphical results on PDF files. Moreover, it is provided with html help documents that help users how to use it. Therefore, Gbest-WSN tool could be useful for the students, teachers and researchers in their work. The rest of the paper is organized as follows. Section 2 is a literature survey about the various developed simulation tools. A brief discussion about the considered routing protocols in our simulation tool is presented in Section 3. Section 4 describes the architecture, functionalities and requirements of the proposed Gbest-WSN simulation tool and its operation flow diagram. The performance of Gbest-WSN is validated through some of illustrative simulations in Section 5. Finally, Section 6 offers some conclusions with future directions for Gbest-WSN tool.

2. Related works

Many simulation tools have been developed to simulate WSNs [25–38], but most of them forces users to have knowledge on programming languages that make them non-attractive for the naïve users. NS-2 [25] is a command-line general purpose network simulator which originally developed for simulating the routing protocols of wide-area networks. Simulations in NS-2 are written by C++ and OTcl (Object-oriented Tcl) languages. Although NS-2 is an object-oriented architecture, it suffers from the scalability problem in terms of the memory usage and the simulation time. Also, it cannot simulate the bandwidth and the power consumption problems of WSNs. SensorSim [26] is an extension version of NS-2 that is provided with additional features for modeling the WSNs. These features include sensor channel models, battery and radio models, lightweight protocol stacks for wireless micro-sensors, scenario generation and hybrid simulation. SensorSim suffers from the scalability problem because it is built up on NS-2.

A component-based compositional simulation environment called Java Simulator (J-Sim) [27,28] is developed entirely in Java. Unlike to NS-2, the concept of J-Sim is based on the Autonomous Component Architecture (ACA). ACA consists from three-level components; target node, sensor node and sink node. These nodes communicate with each other by sending and receiving data using their ports. J-Sim is an object-oriented and provides definitions of sensor node, target node, sink node, communication channel, mobility model and power model. However, it is not user-friendly and supports only 802.11 as MAC Layer and AODV as routing protocol. Neves et al. [29] enhanced the J-Sim functionality by developing a user interface that increases the user-friendliness of J-Sim. The developed simulator was called G-JSim. G-JSim allows the user to define the number of sensor and target nodes, network topology method, power, sensor and mobility models. Although G-JSim increases the user-friendliness in definition of the network parameters, it does not provide a visualization perception for what is happening in the network.

SENSE (Sensor Network Simulator and Emulator) [30,31] is a command-line simulator that was developed for WSNs based on a component-oriented simulation technology. The operation of SENSE depends on command line parameters, C++ compiler and make files to run. SENSE suffers from the user-friendliness problem because it has not a graphical interface and displays the simulation results on the terminal directly. Moreover, it forces the users to have knowledge on C++ language. Rosa et al. [32,33] solved the interface problem of SENSE tool by developing a G-Sense simulator. G-Sense tool is provided with a graphical user interface for creating the simulation scenario and entering the simulation parameters. Moreover, G-Sense allows the user to save and load simulation scenarios and also exports the simulation results on PDF files. G-Sense