ISSN 2277-3061

Simulating Efficient power Wireless Sensor Network over Smart University Campus

Mahmood K. Al Shimmary, Ph.D., Muna M. Al Nayar, Ph.D., Abbas R.Kubba
Control and System Engineering, University of Technology, Baghdad, Iraq.
Computer Engineering, University of Technology, Baghdad, Iraq.

ABSTRACT

Attendance is one of the important factors that determine the students' activity in any educational organizations. Taking attendance manually is considered as a huge task, even if, it was done using traditional methods such as barcode or smart card technologies leading to congestion and waste of time.

The motivation of this papper is to simulate the wireless sensors network of a smart university (UOT) campus using OMNeT++ inorder to enhance the total power consumption of wireless nodes using several nodes scheduiling algorthim . The proposed system has been tested, and the results show that, the processing load of central server is decreased and the total transmission load over the wireless network also enhanced with value of 83%. Different experiments applied over distributed ZigBee wireless nodes in order to enhance network parameters .finally, the network parameters such as total power capacity and number of received packets had been improved using a linear queue node scheduling as proposed method.

Keywords

Wireless Sensor Networks (WSN); ZigBee; OMNet ++; smart university

Council for Innovative Research

Peer Review Research Publishing System

Journal: INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY

Vol. 14, No. 10

www.ijctonline.com, editorijctonline@gmail.com

1. ITRODUCTION

Nowadays, Smart environment became a real idea, especially, after the evolution in different advanced technologies. Smart environment refers to the ongoing progress of the actual ambience at the current time. A smart environment, such as aware system, depends mainly on aggregated information from different sensing devices that located in different sites of genuine world. It collects data about its neighbouring circumstances and also for its interior processing [1].

in the last years, Wireless Sensor Networks (WSN) is a new model of networks has been appeared; its characteristics make it suitable in construction of the smart environment The functionalities of these networks systems not limited to request information for multiple sites , and then forwards them to the administration centers and also contribute with other implementations for Confrontation catastrophes and different ecological topics such as Building automation, Air and Water Quality, Agriculture, Health Monitoring, Traffic Control, Industrial Sensing, Infrastructure Security etc.. [1, 2].

Designing smart home [3] [4], smart university [5] is an evolving issue that enclose ubiquitous/pervasive computing concept of Mark Weiser [6] that favors the seamless integration of technologies in human lives. The stimulus of this paper concentrates on decreasing power consumption of distributed wireless nodes over the university cmpus . The developed prototype shows how evolving technologies of RFID and WSN can add in improving student's attendance method and power conservation .

2. CHOOSING A SIMULATOR

One of the important things that should be taken into consideration before simulates any network, is the selection of appropriate simulator from group of simulators that model the same field.

NS2 is stopped from development since 2007, and it is very old to be used. The subsequent to NS2 is NS3which is a complete-rewrite of NS2, re-implementing the whole NS2 simulator using C++. However, at the time of NS3 developing, it still lacking in supporting WSNs simulation, and so, both of NS2 and NS3 was omitted [7].

MATLAB has a WSN package [9], but its capabilities are limited and do not accomplish the level which is required to simulate in this thesis. In addition to that, MATLAB is used as a general-purpose simulator, instead of a specific one. OPNET simulator is also old, low learning curve, albeit being money expensive, means that it requires a lot of time to understand its logic, designing and implementation. Also, WSN arrangement would require a lot of work and time to do it. However, C++ programming language, network modularity, supporting specific network models, having a well known community and help documentation on the web [7, 8 & 10] .So that, as a conclusion, OMNeT++ was preferred among NS2, N S3, MATLAB, and OPNET**.**

2.1 Omnet++

OMNeT++ or Object Module Networks Testbed using C++ language, it's an expandable, flexible network simulator depended on C++ language's libraries, mainly; it's used to design simulators which appropriate to wireless sensor network. The word network , refers to wide sense such as wired and wireless networks, queuing networks, on-chip networks... etc [11].

OMNeT++ provides component architecture features for models. These components programmed in C++ are nested hierarchically and simple components can be accumulated to compound components and models using a high-level language—Network Description (NED) language.

NED allows the declaration of simple modules, link and collect them into larger one called compound module. The compound modules can be labeled and considered as networks, also, the compound model is considered as self-contained simulation models. The communication between channels can be explained as different unit type, those templates can be utilized in various compound modules. NED language provides many characteristics which makes the network to be scalable. As a result, it can be used to model large communication topologies [11, 12].

2.2 Mixim

MiXiM is the wireless framework modeling application developed to work with an OMNeT++ simulator, MiXiM produced to provide the desired requirements for various types of network, static and dynamic wireless networks especially, designed to work wireless sensor networks. MiXiM focuses on the physical and medium access control layers of the OSI protocols; it provides comprehensive models to determine the network performance such as wave propagation, amounts of interference, average power consumption for the transmitting nodes, in addition to several protocols that it's accommodating with MAC layer [13].

So that, many modules can be implemented using different protocol easily. Also, each sensor node in OMNeT++ can support the following ZigBee protocol stack as shown in Figure 1, in addition to that, different modules can be provided using MiXiM such as mobility module, battery module, the ARP module and utility module [12].

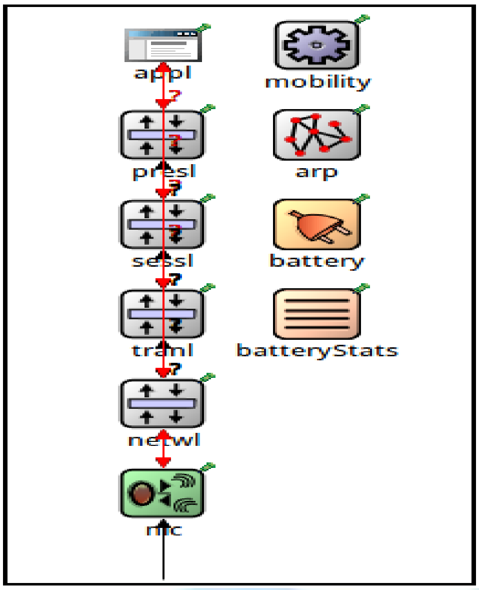

Figure 1: ZB Protocol Stack used in OMNeT++

The motions of the wireless nodes and any mobile matter are one of mobile's module responsibilities. Power and energy issues can be utilized using the battery module. For the sensing wireless nodes, the usage of the node's battery can be emulated. Another important module is called ARP module, it carries out the issues of Address Resolution Protocol (ARP), as example, the conversion and interpretation of the addresses of network and MAC layers. The utility module is the most important module which provides two major advantages: Firstly, it supports global interfacing abilities to collect numerical information of a simulated environment. Secondly, the utility module manages several factors that should be entered by different module for the same node simultaneously [14].

3. SENSORS DISTRIBUTION OVER THE UNIVERSITY

In order to simulate the WSN in the university, the sensor nodes have to be planted across the campus, with each department contains its determined number of nodes. Also, the sensing nodes distributed over the main gates of the university. Those nodes as mentioned in previous chapter use ZigBee device (Xbee series 2 pro) to send the data retceived from the reader node and control circuit after student attendance completion in the specific location to the central server that aggregate and save all of the received data from all the sensors as shown in Figure 2. The university that is chosen as a map campus in the simulation is the University of Technology area that is shown in Figure 3. This map has sensors distributed across its departments with various locations in the simulator as shown in Figure 4. The total number of sensor nodes used is 70 nodes. This number is assumed for the simulation purpose only, and can be extended to any number when it need to be applied in real life. Each reader node will detect 100 users near the RFID field area, and will send the detection (presence of the student) to the central server to act upon those data.

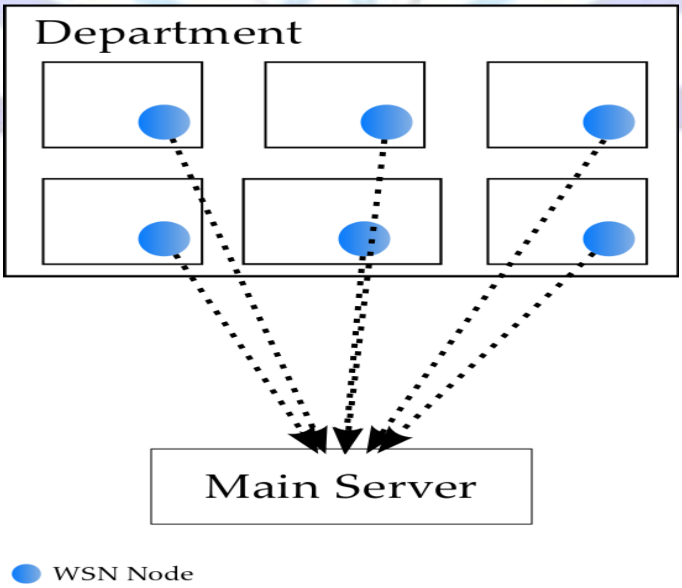

Figure 2: Basic description of how sensors send data to the main server.

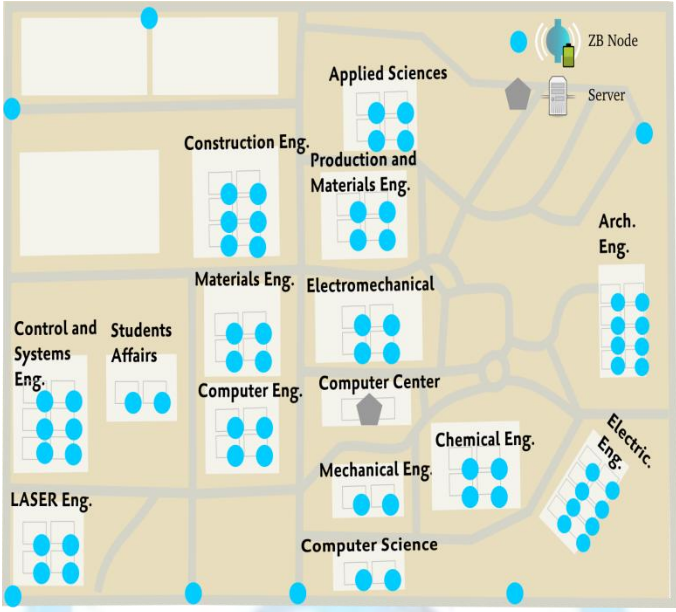

Figure 3: Sensors Distribution over the University Campus

4. WSN PARAMETERS

Each ZigBee node in OMNeT++ adjusted with parameters similar to the XBee Series 2 Pro parameters that implemented in the Hardware prototype as shown in chapter three , as example, the XBee Series 2 Pro has a transmission power of 63 mW, hence the sensor node in the simulator also having a 63 mW of transmission power. The specifications of those simulated sensing nodes are listed in Table 1.

Table1: Sensing Node parameters

Parameter name	***Value***
Max data rate	250 kbps
Communication range	90 -1600 m
Maximum transmission power	63 mW
Sensitivity	-100 dBm
Operating power	3.3 V -295mA
Frequency of the Carrier	2.4 GHz
MAC Protocol	CSMA
Type of Modulation	MSK
Default state of the radio	Transmit
Using of the thermal noise	TRUE

The working environment parameters by which the sensors operate, or the medium which the sensor used to carry out their messages through as shown in the channel parameters information, which are listed in Table 2.

Table 2: Channel parameters

Name of the Parameters	***Amount***
transmitting power of network as Max	2 mW
attenuation threshold of the signal as Max	-100 dBm
path loss as Min	2.5

Those sensor nodes are battery-powered, other meaning, they are battery power dependent, and Table 3 explains the qualifications of batteries for the simulated nodes.

Table 3: Qualifications of Batteries Nodes

Name of the Parameters	***Amount***
Power capacity	99999 mAh
Operating Voltage	3.3 Volt

5. THE PROPOSED METHODES AND SIMULATION RESULTS

The payload size of the XBee Series 2 Pro is 72 bytes, after setting this parameter in the simulator; the detected tags have been assumed and its overhead data (to separate each tag from other to split them in the server) can reach to three tags per payload. That is, each transmission, with a 72 bytes payload, it will transmit only three tags. In this case study and the follows, each node will try to detect 90 tags and send them to the central server and a total of 30 packets from each node will be transmitted.

5.1 Case Study 1: Adjusting Initial Settings

Initial settings, it means, the default nodes configurations shipped with OMNeT++. In another word, all nodes try to send at the same time using Carrier Sense Multiple Access/Collision Avoidance(CACSMA/CA) protocol implemented over the data link layer that provided by 802.15.4 standard.

The proposed system simulated ZigBee stack protocol which is dealing with all layers of OSI model .So that, different factors can be AQWtransmissions between two packets. In the following states, all the nodes will transmit at the same initial time, with different values of PPT 0.01, 0.1 and 1s between packets respectively. After setting-up the environment in the simulation, distributing sensors across the university, and running the simulation, the data about sensor nodes, or network information pertaining to each one, were generated. The total power capacity of the nodes batteries (mW-s) has been presented in Figures 4, 5 and 6. The horizontal line represents the average of the plotted values. Every node transmits 30 packets. The x-axis represents the nodes IDs, as example, all of the nodes transmit 30 packets. Table 4-4 contains the information presented in the mentioned figures. The average column contains data from all of the sensor nodes, which are 70 nodes. That is to say, all of the battery capacities are summed, and then divided by 70

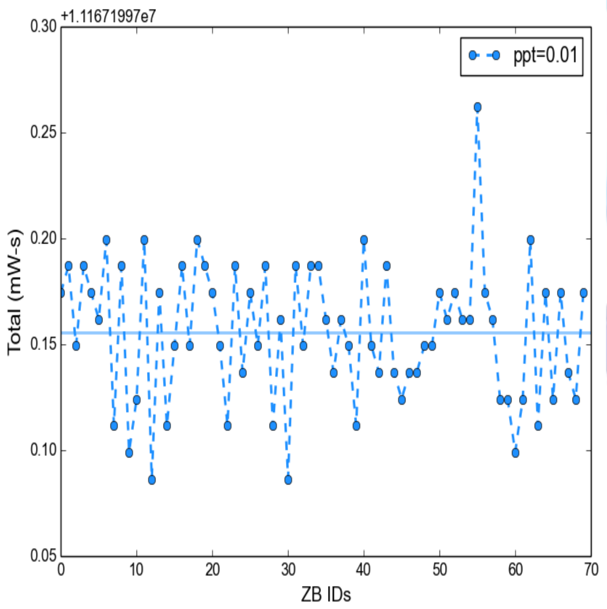

Figure 4: Sensors Power Consumption with 0.01of PPT

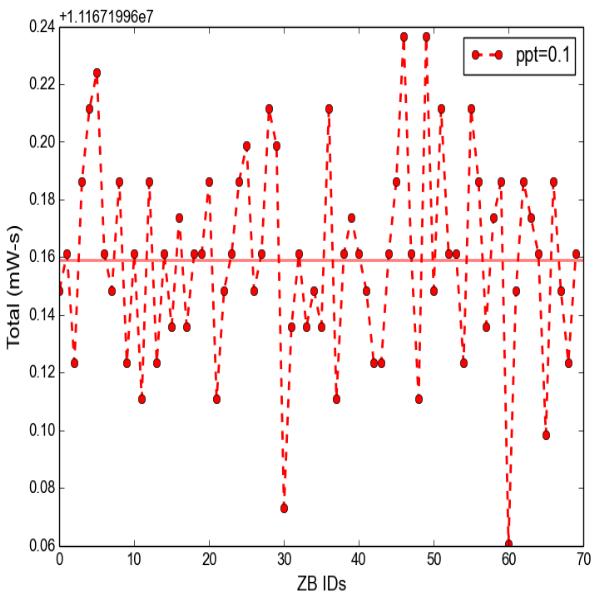

Figure 5: Sensors Power Consumption with 0.1of PPT

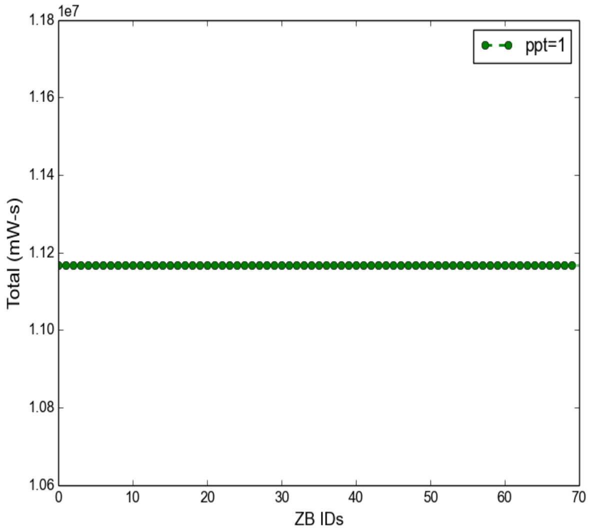

Figure 6: Sensors Power Consumption with 1 of PPT

The obtained results from Table 4 shows the number of received packets at central server is directly proportional with values of PPT at the expense of consumed battery power for each node, the maximum number of packets can be received in this case is 2050 with value 1 off PPT, while the number of desired packets is 2100. Different values of PPT more than 1sec was tried but doesn't obtain better results because all nodes try to send at the same time .This leads to try another case in hope to enhance the results.

Table 4: Case Study 1 Parameters Information

PPT	Average Total (m W-s)	Rx packets at server
0.01	11167199.85533178	58/2100
0.1	11167199.759109126	587/2100
1	11167199.622853007	2050/2100

5.2 Case Study 2: The Proposed Method

Since the recording of the students presence had been stored in distributed RF-ZB nodes, so that, it can be delayed or transmitted to the main server later, and the final day attendance is not very urgent or critical data to be processed instantly, the linear scheduling algorithm used for allowing only a single sensor node at a time to send its data to the server. This approach mitigates the contention between sensor nodes to check the transmission medium simultaneously in order to save delivery of the sensed data. Hence, linear queue node scheduling is programmed in OMNeT++ among the sensor nodes to gain better results.

The simple queuing scheduling example is shown in Figure 7. Each tag in each node has its unique time to be allowed to transmit the packet. The following equation was used:

scheduleTime = currentSimulationTime + ((myIndex * nbPackets) +(sentPackets)) *α

Where current Simulation Time is the current simulation time (in seconds), myIndex is the node index, that is, node with ID 0, has index of 0, node 1 has index of 1, and so on. nbPackets is the total number of packets to be transmitted (30 in these case studies), and sentPackets is a counter of the currently sent packets.

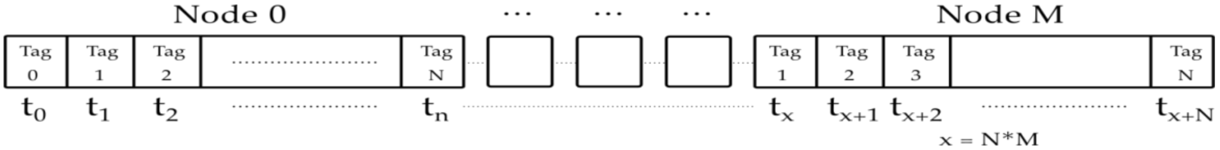

Figure 7: A Simple Linear Node Scheduling Table 4-5: Case Study 2 Parameters Information

Finally, α is a factor that is used to widening or squeezing the time between successive transmissions. Analogically, it is like the principle of PPT. the total power capacity of the batteries (mW-s) of nodes has been explained in Figures 8, 9 and 10. The horizontal line represents the average of the plotted values. Table 5 contains the results had been obtained from different values of α which tried in this method.

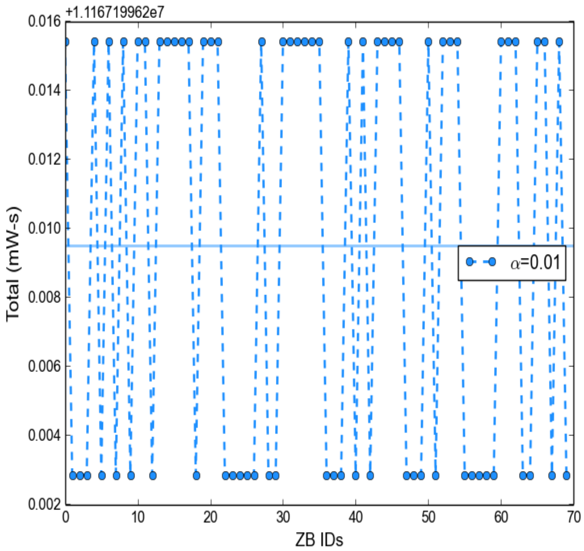

Figure 8: Sensors Power consumption with 0.01 of α

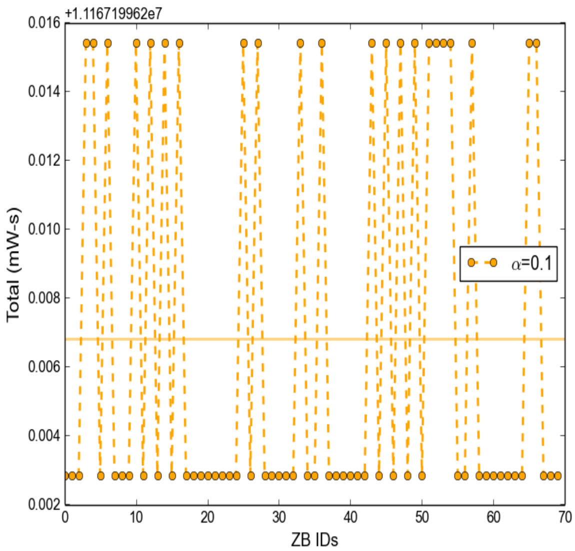

Figure 9: Sensors Power consumption with 0.1 of α

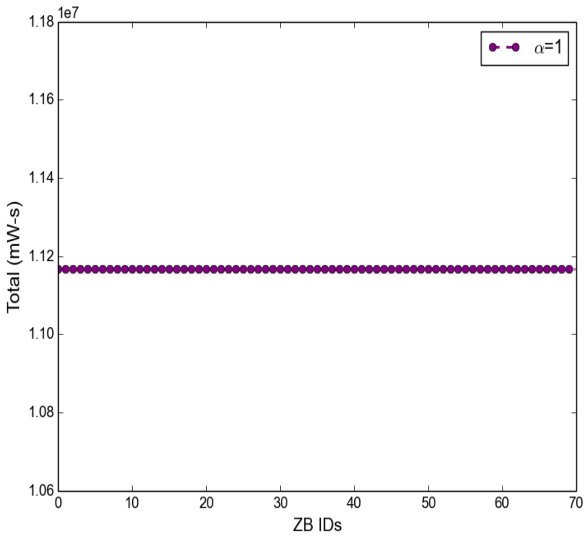

Figure 10: Sensors Power consumption with 1 of α

As shown in Table 5, the proposed method satisfies the desired results. When compared the number of received packets with the same values of PPT and α, as example, with value of 0.1 in case study1,it was 587 packets while in case study 2 it is 2058 with approximate values of total power capacity. The main goal being achieved, the number of received packets is the same as when it has been transmitted, 2100 packets with the same total power capacity of nodes batteries that has been consumed in case study1.

Table 5: Case Study 2 Parameters Information

α	*Average Total (m W-s)*	*Rx at server*
0.01	11167199.629495559	2029/2100
0.1	11167199.626802633	2058/2100
1	11167199.622853007	2100/2100

6.CONCLUSION

This work demonstrates the deployment of wireless sensor network with RFID technology to evaluate the possibility of smart university with considering the most common issues such as attendance, electrical power monitoring and control in a smart manner with minimum cost and efforts. The main goal of this work has been achieved; the most important aspects are enhanced and concluded as follow:

- Selecting low power, appropriate data rate wireless communication protocol (ZigBee) which is fitting with simple size data such that been used in a smart university.
- Decreasing the load on central server and wireless network to 83 % due to the use of distributed wireless RF-ZB nodes.
- Enhancing the total power consumption of the wireless sensor network by applying proposed adjustable linear queue node scheduling method.

REFERENCES

1. Pathan, A. K., Choong Seon Hong, and Hyung-Woo Lee. "Smartening the environment using wireless sensor networks in a developing country", Advanced Communication Technology, the 8th International Conference. ICACT, Vol. 1, PP: 5, IEEE, 2006.
2. Augusto, Juan Carlos, and Paul McCullagh. "Ambient intelligence: Concepts and applications." Computer Science and Information Systems Vol.4, No.1, PP: 1-27, 2007.
3. M. Shanmugasundaram, G. Muthuselvi, S. Sundar, "Implementation of PIC16F877A Based Intelligent Smart Home System", Volume 5 International Journal of Engineering and Technology (IJET), Apr-May 2013.
4. Usha Sharma, S.R.N. Reddy, " Implementation of a WSN based Home/Office Automation (HOA)", Volume 3 International Journal of Engineering and Advanced Technology (IJEAT) ,Issue 3, February 2014.
5. Mahmood K. Al Shimmary, Muna M. Al Nayar, and Abbas R. Kubba. "Designing Smart University using RFID and WSN." International Journal of Computer Applications 112.15 (2015).
6. Mark Weiser, "The computer of the 21st century", Scientific American, Volume 265(3), pp.66--75, September 1991.
7. Mishra, Vinita, and Smita Jangale. "Analysis and comparison of different network simulators"
8. Varga, András, and Rudolf Hornig "An overview of the OMNeT++ simulation environment." Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.
9. Wislab, Wireless sensor network simulation - tutorial for matlab, http://wislab.cz/our- work/wireless- sensor-network- simulation tutorial- for-matlab, [Online; accessed July 12, 2015
10. Sundani, Harsh, et al. "Wireless sensor network simulators a survey and comparisons." International Journal of Computer Networks Vol. 2, No.5, PP: 249-265. 2011.
11. OMNeT++ User Manual, Online accessed on April 19.2015 http://www.omnetpp.org/doc/omnetpp41/manual/usman.html
12. Lei, C. "Cooperative Adaptive Cruise Control model study based on traffic & network simulation." to be found via (URL visited in May 2011): http://www. utwente. Nl/ewi/dacs/assignments/completed/bachelor/reports/2011_lei_chenxi . Pdf (2011), Online accessed on April 19.2015
13. OMNeT++ official website, http://www.omnetpp.org, Online accessed on April 19.2015
14. Kopek, Andreas, et al. "Simulating wireless and mobile networks in OMNeT++ the MiXiM vision." Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), pages: 71, 2008.