ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 9, March 2016

Delay analysis in sensor networks

N.MARIE CLAUDE[1], RAMESH K [2],KANNAN V [3]
[1,2] Research scholars, [3] Principal
[1] *Department of Computer Science and Engineering*

[2] *Department of Electronics and Communication Engineering*

[1,2] *St.Peter's University, Chennai, India,* [3]*Jeppiaar Institute of Technology, Chennai , India*

Email: [1]nishusishu@gmail.com ,[2]drvkannan@gmail.com

***Abstract-* Network systems are used to of carrying real-time traffic. This field has drawn enormous attention from current researchers because of its flexibility and robustness. However, designing efficient SNS over Mobile Ad Hoc Networks (MANET) is still challenging topic because of its less-implement a control mechanism over a wireless network that is capable predictable aspects, such as inconsistent delay, packet drop probability, and dynamic topology. This paper presents design guidelines of SNS over MANET using the network Simulator version 2, NS2 software. It investigates the impact of delay and packet drop**

Index terms – Mobile Adhoc Networks (MANET), Sensor Networks (WSN).

I.INTRODUCTION

Sensor network is a computer network that is wireless, and it is commonly associated with a telecommunications network whose inter connections between nodes are implemented without the use of wires. Wireless telecommunications networks are generally implemented with some type of remote data transmission system that uses electromagnetic waves, such as radio waves, for the carrier and this implementation usually takes place at the physical level or "layer" of the network. In telecommunications, Wireless communication is the transfer of information without the use of wires. [2] The reasons for using wireless network are cost effectiveness of network deployment, and its applicability to environments where wiring is not possible or it is preferable solution compared with wired networks. Wireless network is described as a network of devices which communicate by using wireless technologies [1]. When designing wireless networks and/or studying their behavior under various conditions, software simulation tools are often used. In this paper the software tool Network Simulator (Version 2), widely known as ns-2, is described and used for the simulation of selected illustrative examples of wireless sensor networks. The main goal of this paper is to learn how to use simulation for designing and studying wireless networks. In order to achieve the defined the task, analyze literature sources related to wireless communication networks. Briefly describe the basic wireless networks categories. Analyze wireless LAN networks and briefly describe their components and technologies. Explain the Wi-Fi technology. Analyze literature sources related to wireless networks simulators. Analyze the Network simulator ns-2 and give its detailed description. Present a brief comparison of ns-2 simulator with other open source network simulators. Specify the configuration for the simple wireless network and create corresponding model by using ns- 2 simulator. Demonstrate selected characteristics of the specified network configuration using the simulation model.

The IEEE 802.15.4 protocol has the ability to support time - sensitive Wireless Sensor Network (WSN) applications due to the Guaranteed Time Slot (GTS) Medium Access Control mechanism. Recently, several analytical and simulation models of the IEEE 802.15.4 protocol have been proposed. The motivation that has driven this work is the validation of the Network Calculus based analytical model of the GTS mechanism that has been previously proposed and to compare the performance evaluation of the protocol as given by the two alternative approaches.

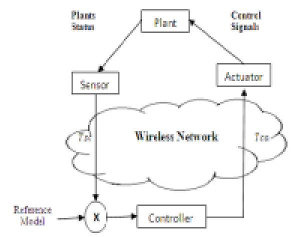

Fig. 1 The general structure of Wireless Sensor Network

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 9, March 2016

The basic distributed structure of the WSN is shown in Fig.1. The state of the plant is sensed by sensors and is sent to the controller over the WSN. The controller compares the states with the reference model and computes control signal accordingly. Then, the control signal is sent to the actuator through the wireless network to be implemented in the plant. This paper attempts to investigate the relation between packet delay and sampling period, as well as the effect of packet delay and packet drop on the performance of a simple control mechanism using the simulation software NS2.

The rest of the paper is organized as follows. Section II explores the related work. Section III explores the design issues of the underlying network for successful implementation of WSN, section IV explains the model used for the simulation, section V implements programming language in NS2 and section VI presents the results. Section VII draws some conclusions.

III. RELATED WORK

Design of wireless Network uses NS2, as a base on Security evaluation, and describes the proposed model of the system and complete description of the Simulations and software program needed for implementing the Network. Ns-2 is a widely used tool to simulate of networks. Network simulator is a part of software that predicates the performance of a network without a real network being there.NS2 is a vital simulation tool for networks. It supports a number of algorithms for routing and queuing. NS2 is very helpful because it is very costly to verify viability of new algorithms, test architectures, check topologies, check data transmission etc. Network simulators are names for series of discrete event network simulators and are heavily used in ad-hoc networking res. and support popular network protocols, offering simulation results for wireless networks [3]. We particularly focus on the performance evaluation of the GTS mechanism, comparing the obtained simulation results with the ones that were previously obtained [4] using an analytical model based on Network Calculus. The behaviors of both models are roughly identical in terms of the GTS data throughput and the delay bound; the analytical results upper bound the simulation results.

In order to ease the process of extracting data for performance study, the NS-2 Trace Analyzer is proposed. This software is a tool for extracting and presenting trace files for the network simulation environment of NS-2. The NS-2 Trace Analyzer software consists of three layers. The first layer is the source layer which consists of the trace file data. The second layer is the processing layer. This layer processes the data obtain from the source and convert it to meaningful format for the third layer. The third layer is the presentation layer. This layer presents meaningful data in the form of graph, table and report for network performance study, i.e. throughput, end-to-end delay, packet loss and jitter. Through the NS-2 Trace Analyzer the user would be able to do performance study of a network scenario through interactive GUI [5].

Yunbo Wang Mehmet C. Vuran Steve Goddard [1] have proposed to improve the event detection reliability, usually timely delivery of a certain number of packets is required. Traditional timing analysis of WSNs are, however, either focused on individual packets or traffic flows from individual nodes a spatio- temporal fluid model is developed to capture the delay characteristics of event detection in large-scale SNs. Mean delay and soft delay bounds are analyzed for different network parameters. The resulting framework can be utilized to analyze the effects of network and protocol parameters on event detection delay to realize real-time operation in WSNs. but fail to give single hop delay distribution [6].

III. SN DESIGN ISSUES

Control networks generally produce small, compared to data networks, but time-critical packets [6]. The WSN must the following constraints. 1. Frequent small sized packets.2.Bounded packet delay, however, it can be noted that the smaller the delay the better the performance.3. Quality packet delivery. 4. Energy consumption. Delay and packet drop can degrade the performance of NCS significantly and even destabilize the entire NCS. Therefore, in this research paper, these two network issues are the main focus.

A. Packet Delay

The total closed loop delay τ_{total} can be formulated as $\tau_{total} = \tau_{sc} + \tau_c + \tau_{ca}$, where τ_{sc} is sensor to controller, τ_c is controller computation and τ_{ca} controller – to- actuator delays, respectively. However, for simplicity, the controller delay τ_c can be ignored as it is negligible compared to the delays τ_{sc} and τ_{ca}. Therefore the total delay is approximately $\tau_{total} \approx \tau_{sc} + \tau_{ca}$ .If no delay compensation technique is implemented ,the controller must receive the *j*-th sample from the plant before computation of the *j*-th control signal to ensure stable operation of the wireless network system. Therefore, total delay must satisfy the relation $\tau_{total} \leq T$, where T is the sampling period.

B. Packet drop rate

SN carries real-time traffic, therefore, it might be beneficial to drop a packet that cannot be transmitted immediately in order to avoid retransmissions. Thus, the tolerable packet drop rate must be analyzed to maintain the system stability.

C. Energy –Aware sensor Networks

Nodes in a WSN are usually highly energy-constrained and expected to operate for long periods from limited on-board energy reserves. To permit this, nodes – and the embedded software that they execute – must have energy-aware operation. Energy efficiency has been of significant importance since WSNs were first conceived but, as certain applications have emerged and evolved [7], a real need for ultra-miniaturized long-life devices has reemerged as a dominant requirement. Because of this, continued developments in energy-efficient operation are paramount, requiring major advances to be made in energy hardware, power management circuitry and energy aware algorithms and protocols.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 9, March 2016

The energy components of a typical wireless sensor node are shown in Fig. 2. Energy is provided to the node from an energy source, whether this is a form of energy harvesting from sources such as solar, vibration or wind, or a resource such as the mains supply or the manual provision and replacement of primary batteries. Energy obtained from the energy source is buffered in an energy store this is usually a battery or super capacitor. Finally, energy is used by the node's energy consumers these are hardware components such as the microcontroller, radio transceiver, sensors and peripherals. With the increased usage of energy sources in nodes [8, 9], the need for energy stores other than batteries (many of which suffer from only offering a limited number of charging cycles) is increased. This can be seen by the body of research that is now utilizing super capacitors (devices that are similar to standard electrolytic capacitors, but with capacities of many Farads) to store the node's energy [9, 10].

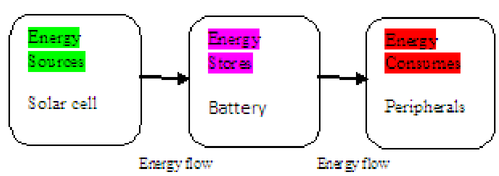

Fig. 2 Energy components of a typical wireless sensor node

IV. SIMULATION MODEL

A. Plant and Control Model

A simple water tank model is chosen as the plant, shown in Fig.3, so that the controller delay τ_c can be ignored. The current water height is sent to the controller sends the controlled input flow (signal) to the tank, which is initially empty. The input flow is tuned according to the difference between the tank height and the desired height, REF.

The continuous time water model is

$$\tau y + y = Ku \qquad (1)$$

Where the time constant τ = 63.2456, the gain K =15.8144, *y* is the height and *u* is the input flow of the water tank. Using the Newton approximation method, the discrete model of (1) is

$$\left(\frac{\tau}{T} + 1\right) y(j) = Ku(j) + \frac{\tau}{T}\, y(j-1) \qquad (2)$$

Where T is sampling period, *y(j)* and *u(j)* are the output and input at the *j*-th sample, respectively.

Here we use simple propotional controller with propotional constant K*p* = 5 and the control law is

$$u(j) = Kp\,(REF - y(j)) \qquad (3)$$

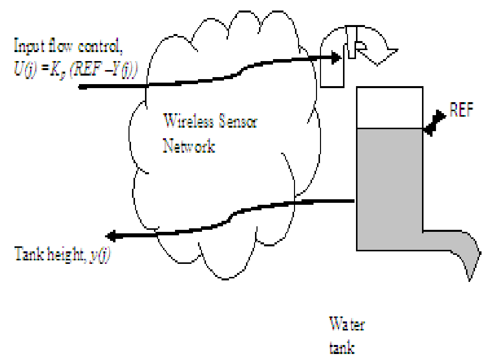

Fig. 3 Plant model used to analyze packet delay and drop rate

B. Network simulator

Network Simulator (Version 2), widely known as ns-2, is simply a discrete event driven network simulation tool for studying the dynamic nature of communication networks. It is an open source solution implemented in C++ and Otcl programming languages. Ns-2 provides a highly modular platform for wired and wireless simulations supporting different network element, protocol (e.g., routing algorithms, TCP, UDP, and FTP), traffic, and routing types. In general, ns-2 provides users with a way of specifying network protocols and simulating their corresponding behaviors. Result of the simulation is provided within a trace file that contains all occurred events. NS is a REAL simulator developed by UCB (University of Carolina Berkley) in 1989. An earlier version of simulator ns- 1 (version 1) was developed by Floyd and McCanne. Ns-2 (version 2) was developed at LBNL (Lawrence Berkeley National Laboratory) under VINT project (Virtual Internet Test bed) by LBL [8], PARC [9], UCB [9], and USC/ISI (University of Southern Carolina/ Information Sciences Institute) [10]. It is currently maintained at USC/ISI, with input from K. Fall, S. Floyd et al. NS-3, a newest version of ns written in C++ and Python came as a replacement for ns-2 in 2006. Ns-3 is not an extension of ns-2, but a new simulator that does not support the ns-2 APIs. Also, ns-3 does not support all NS-2 functionality; some models are still being ported from ns-2.Totally changed API and missing functionality in ns-3 are only some of aspects that favor the use of ns-2. The real advantage of ns-2 over ns-3 is its popularity. Its user base is greater than 1000 institutions and 10000 users. NS-2 size in terms of line of codes is above 200k written in C++ and TCL and 350 page manual. Ns-2 is portable tool that works on most UNIX and Windows like operative systems. Ii it supported on Linux (Ubuntu, Fedora etc.), FreeBSD, SunOS/Solaris, HP/SGI, and Windows 95/98/NT/ME/2000/XP.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 9, March 2016

V. PROGRAMMING LANGUAGE IN NS2

A. Sensor Networks

The network consists of 4 nodes (n0, n1, n2, n3). The duplex links between n0 and n2, and n1 and n2 have 2 Mbps of bandwidth and 10 ms of delay. The duplex link between n2 and n3 has 1.7 Mbps of bandwidth and 20 ms of delay. Each node uses a DropTail queue, of which the maximum size is 10. A "tcp" agent is attached to n0, and a connection is established to a tcp "sink" agent attached to n3. As default, the maximum size of a packet that a "tcp" agent can generate is 1KByte. A tcp "sink" agent generates and sends ACK packets to the sender (tcp agent) and frees the received packets. A "udp" agent that is attached to n1 is connected to a "null" agent attached to n3. A "null" agent just frees the packets received. A "ftp" and a "cbr" traffic generator are attached to "tcp" and "udp" agents respectively, and the "cbr" is configured to generate 1 Kbyte packets at the rate of 1 Mbps. The "cbr" is set to start at 0.1 sec and stop at 4.5 sec, and "ftp" is set to start at 1.0 sec and stop at 4.0 sec.

B. TCL- Algorithm

Step 1: Create a simulator object
set ns [new Simulator]
Define different colors for data flows
ns color 1 Blue ,ns color 2 Red
Step 2: Open the NAM trace file
set nf [open out.nam w]
ns nam -trace-all $nf
Step 3: Define a 'finish' procedure
proc finish { } { global ns nf
Close the NAM trace file
execute NAM on the trace file }
Step 4: Create nodes and links
Create links between the nodes
Set Queue Size of link (n2-n3)
Give node position (for NAM)
Monitor the queue for link
Setup a TCP connection
Setup a FTP over TCP
Setup a UDP connection
Setup a CBR over UDP connection
Schedule events for the CBR and FTP agents
Step 5: Call the finish procedure after 5 seconds of simulation time
Print CBR packet size and interval puts "CBR packet size = [$cbr set packet_size_]" puts "CBR interval = [$cbr set interval_]"
Run the simulation

VI. SIMULATION EXPERIMENT RESULTS

A. Simulation Scenario

Wireless network performance depends mainly on the end to end throughput and average delay. Different applications place different requirements on the network. Real time applications such as voice over IP are highly sensitive to delay but function satisfactorily with little bandwidth. At the other hand data transfer applications like FTP are insensitive to delay but require as much bandwidth as possible. In this section we are going to present simulation scenario aimed at stimulating the network performance through network throughput, packet drop rate and average packets end to end delay. Within the scenario we have stated clearly the layout and configuration of our network and the simulation experiment setup.

B. Results

As already mentioned, wireless simulators provide full control to researchers in investigating traffic flow behavior, but do not always reflect real-world scenarios. Therefore, in this chapter we will present results of simulation obtained by using ns-2 simulator where the experiment setup is aimed primarily at demonstration of ns-2 features and illustrations of some basic performance for simulated network. The results of our simulation from ns-2 trace files are shown in subsequent sections.

C. Nam- output

The Nam class outputs at runtime in our simulation setup (Figures 4, 5,6and7) show two networks consisting of two wireless access points (Access Point_1 corresponds to node 0 and Access Point_2 to node 3) and five nodes (node_1, node_2, node_4) that are connected to these access points and the traffic of packets between nodes.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 9, March 2016

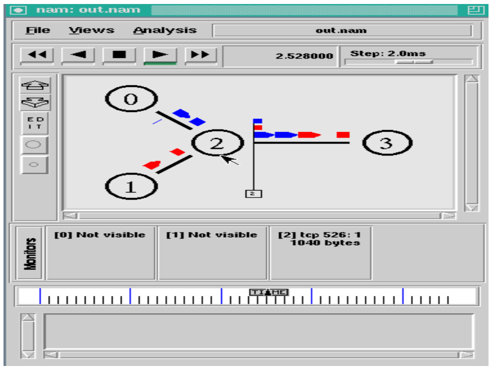

Fig. 4 Nam-output showing nodes of Wireless sensor Networks

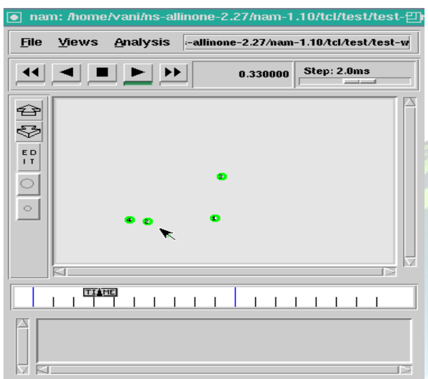

Fig. 5 End-to-End node display

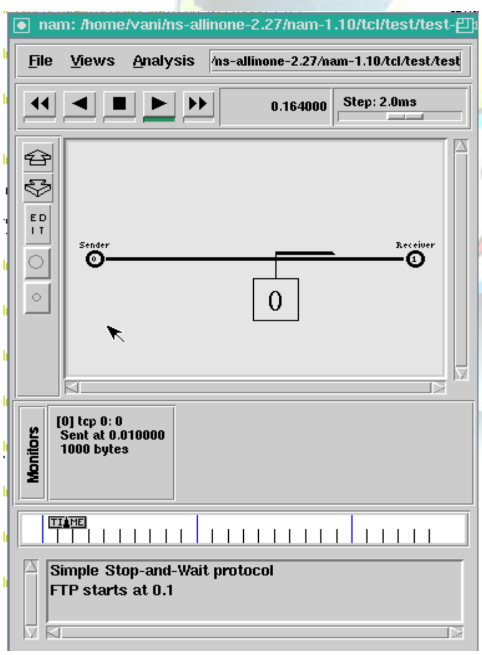

Fig. 6 Simulation snapshot_1

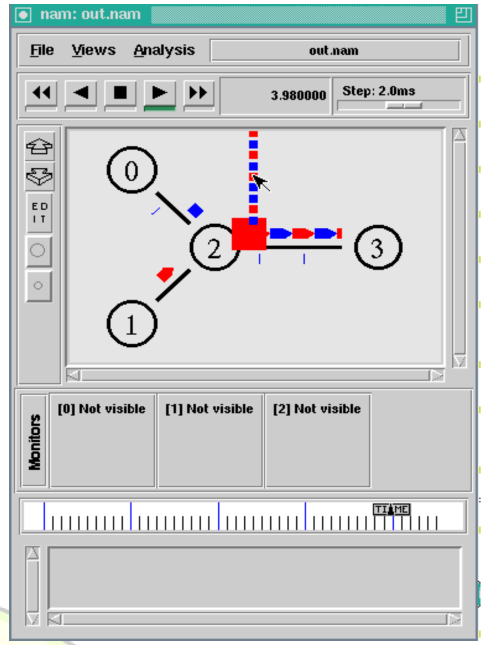

Fig. 7 Simulation snapshot_2

D. Packet Drop Rate

Packet drop occurs when one or more packets of data travelling across a computer network fail to reach their destination. Fig.8 shows the relation between drop packets and transmission rate. Here *x* axis represents the simulation time and *y* axis represents to number of dropped packets.

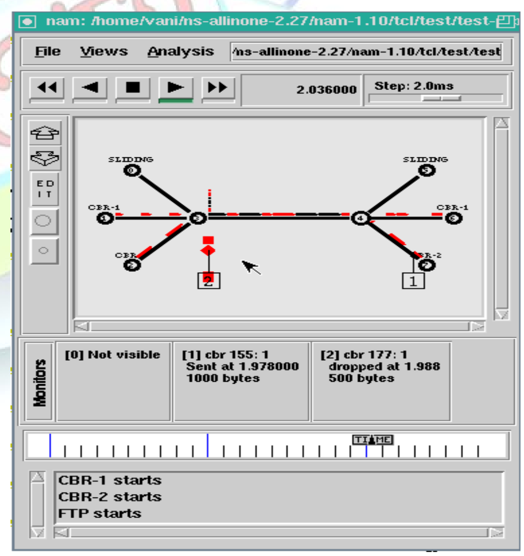

Fig. 8 Packet drop for two nodes

E. Average Packets END to END Delay

Packet delay is the difference in end-to-end delay between selected packets in a flow with any lost packets being ignored. Fig.8 and 9 shows the relation between delay of the packets and transmission rate. *x* axis represents the simulation time and *y* axis represents to average delay packets.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 9, March 2016

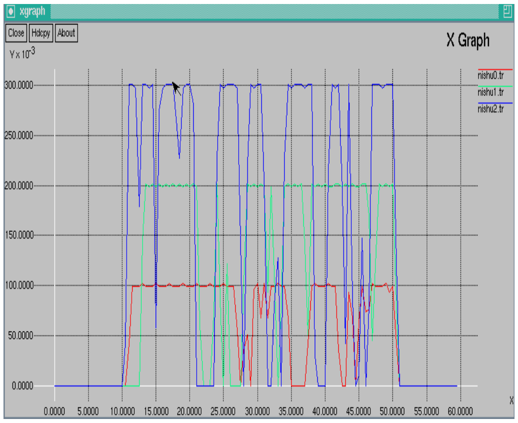

Fig. 9 Delay for three nodes (Node_1 is a red line and Node_2 is a green line)

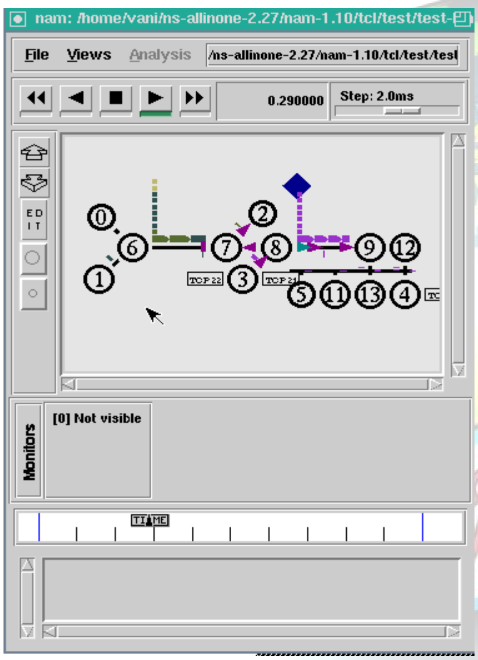

Fig. 10 Delay analyses for nodes

VII. CONCLUSION

Wireless network is a computer network that is wireless, and it is commonly associated with a telecommunications network whose interconnections between nodes are implemented without the use of wires. Wireless telecommunications networks are generally implemented with some type of remote data transmission system that uses electromagnetic waves, such as radio waves, for the carrier and this implementation usually takes place at the physical level or "*layer*" of the network.

The reasons for using wireless network are cost effectiveness of network deployment and applicability to environments where wiring is not possible or it is preferable solution compared with wired networks. When designing wireless networks and/or studying their behavior under various conditions, software simulation tools are often used. In this master thesis the software tool Network Simulator (Version 2), widely known as ns-2, is described and used for the simulation of selected illustrative examples of wireless networks.

Network Simulator (Version 2) is a discrete event driven network simulation tool for studying the dynamic nature of communication networks**.** Ns-2 provides a highly modular platform for wired and wireless simulations supporting different network elements, protocols, traffic and routing types. In general, ns-2 provides users with a way of specifying network protocols and simulating their behavior.

The simulation results in following conclusions about network behavior, comparative study between packet drop rate and transmission rate for the nodes of Access point_1 and Access point_2 shows that the performances of the observed networks differ and there is high fluctuation for single node in 3 node networks. Another important feature of performance study is average packets end to end delay and transmission rate for the nodes of Access point_1 and Access point_2. The performance of the whole network is finding to be transient initially, but it comes to a stable state after a certain amount of time.

REFERENCES

[1] William Stallings, Wireless communications and networking, William Stallings books on computer and data communications technology,Publisher Prentice Hall, 2002, ISBN10 0130408646, ISBN13 9780130408648, Length 584 pages.

[2] Wireless Communication, link http://www.atis.org/ , Archived from the original on 2008-01-02.

[3] Andrea Goldsmith, *Wireless Communications*, Cambridge University Press, September 2005, ISBN13: 9780521837163.

[4] Jim Geier, *Wireless Networks first-step*, Cisco Press August 03, 2004,1-58720-111-9.

[5] H. Labiod, H. Afifi and C. Desantis, *WI-FI, Blutooth, Zig Bee and WIMAX*, Springer 2007, ISBN 978-1-4020- 5397-9.

[6] IEEE-SA, *IEEE 802.11n-2009—Amendment 5: Enhancements for Higher Throughput*, 29 October 2009.

[7] Lawrence Berkeley National Laboratory, link: http://ee.lbl.gov /, December 2010.

[8] Berkeley University, link: http://www.cs.berkeley.edu/ , December 2010.

[9] University of Southern Carolina, link: http://www3.isi.edu/home, December 2010.

[10] Eitan Altman and Tania Jimenez, *Network Simulator for beginners*, December 4, 2003.

[11] Kevin Fall and Kannan Varadhan, *The ns Manual*, May 9, 2010.

[12] Teerawat Issariyakul, Ekram Hossain, *Introduction to Network Simulator ns-2*, 2009.

[13] IEEE 802.15.4 Standard-2003, “Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (LR WPANs)”, IEEE SA Standards Board, 2003.

[14] A. Koubaa, M. Alves, and E. Tovar, “Time-Sensitive IEEE 802.15.4 Protocol,” chapter of the book “Sensor Networks and Configurations: Fundamentals, Techniques, Platforms, and Experiments,” Springer-

Verlag, Germany, Jan. 2007

[15] A. Koubaa, M. Alves, and E.Tovar, "A Comprehensive Simulation Study of Slotted CSMA/CA for IEEE 802.15.4 Wireless Sensor Networks," Workshop on Factory Communication Systems (WFCS'06), Torino (Italy), Jun. 2006.

[16] I. F. Akyildizn et al, "Wireless sensor networks: a survey," Computer Networks, vol. 38, no. 4, pp. 393–422, 2002.

[17] Wei Ye, et al, "Medium Access Control with Coordinated, Adaptive Sleeping for Wireless Sensor Networks" USC/ISI TECHNICAL REPORT ISI-TR-567, JANUARY 2003.

[18] H.D.Trung et al. "Performance evaluation and comparison of different ad hoc routing protocols", Department of Electrical Engineering, Chulalongkorn University, Bangkok, Thailand, May 2007.

[19] Abdul Hadi Abd Rahman et al, *"Performance Comparison of AODV, DSDV and I-DSDV Routing Protocols in Mobile Ad Hoc Networks"* European Journal of Scientific Research ISSN 1450-216X Vol.31.

AUTHORS PROFILE

Mr.N.Marie Claude was born at Pondicherry on 1st May 1978.He received his Bachelors of Computer Science and Engineering from Madras University,Chennai, Masters Degree from sathyabama University,Chennai.He is currently pursuing his Ph.D in St.Peters University and working as Associate Professor in IFET Engineering College, Villupuram, TamilNadu. His Research interest pertains to Networking, Cloud Computing and Image processing .

Mr. K. Ramesh was born in Odugathur, Tamil nadu, India on 10th June 1966. He received his Bachelors of Electrical and Electronics Engineering from Anna University in the year 2007.Masters Degree in VLSI Design from Sathyabama University in the year 2011. He is currently a Ph.D., student of St Peter's University, Chennai. Majoring in Electronics and Communication Engineering. He has presently working as Asst. Professor. His current research focuses on wireless sensor networks and VLSI design. He has seven Research publications in National/International Journals and Conferences to his credit.

V.Kannan was born in Ariyalore, Tamil nadu, India in 1970. He received his Bachelor Degree in Electronics and Communication Engineering from Madurai Kamarajar University in the year1991,Masters Degree in Electronics and control from BITS, Pilani in the year 1996 and Ph.D., from Sathyabama University, Chennai, in the year 2006. His interested areas of Research are Optoelectronic Devices ,VLSI Design, Nano Electronics, Digital Signal Processing and Image Processing. He has more than 150Research publications in National / International Journals / Conferences to his credit. He has 22 years of experience in teaching and presently working as Principal in Jeppiaar Institutue of Technology, Sriperumpudur, Chennai, India, He is a life member of ISTE.