

# Wireless Sensor Network Simulator

Deepa S, Mamatha C R, Rashmi R  
 UG Student<sup>1</sup>, Assistant Professor<sup>2</sup>, Assistant Professor<sup>3</sup> Dept. computer  
 science and engineering  
 Vemana Institute of Technology  
 Bangalore, India

**Abstract**— Wireless communication has been evolving and growing rapidly during the last few years. In order to meet the demand of this growth researchers as well as industry have been working for new techniques and new standardizations. Wireless sensor networks have been introduced to use in many applications. They provide a bridge between the physical and virtual worlds. The main goal of this paper is to learn how to use simulation for designing and studying wireless networks. Analyzing literature sources related to wireless communication networks and describing the basic wireless networks. Network simulator-2 as a wireless simulator is being demonstrated in this paper. Network simulator ns-2 and its detailed description along with the programming languages used is explained. Demonstration of the throughput, delay and packet dropped by the nodes is compared and the simulation is shown.

**Keywords**— *Wireless, sensors, ns-2, simulation, network component.*

## I. INTRODUCTION

Wireless network is described as a network of

also bring many open issues in the network design. Normally, analyzing methods, computer simulation, and physical measurement are the three main techniques used for analyzing the performance of wired and wireless networks.

Setting up a network to do some real experiments is the best way for studying about communication. However, setting a network is not easy and costly. For this reason, a virtual network provided by network simulator is used for experiment in only one computer. Specially, NS2 which is free and easy to use is the popular all over the world. In our project, NS-2 is chosen as our simulation environment because it is the most widely used network simulator.

NS-2 has several key benefits which suit for

doing the simulation:

- It provides extensive support for simulating TCP/IP, Routing and multicast protocols over wired and wireless network.
- It provides a lot of standard modules to be used in sensor network.
- It is an object oriented design which provides a lot of documents.

devices which communicate by using wireless technologies. Wireless network refers to any type of computer network that is wireless, and is commonly associated with a telecommunications network whose interconnections between nodes are implemented without the use of wires. 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. When designing wireless networks and/or studying their behaviour 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. Optimizing a wireless sensor network

- It uses OTcl to specify the components and OTcl to glue them together.

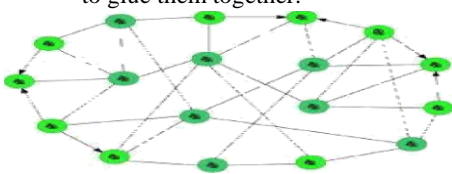


Figure 1: Reality Network

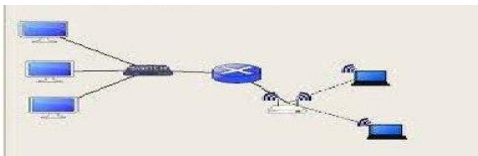


Figure 2: Network for simulation

## II. NETWORK SIMULATOR

The network simulator (NS), which is a discrete event simulator for networks, is a simulated program developed by VINT (Virtual InterNetwork Testbed) project group (A Collaboration among USC/ISI, Xerox PARC, LBNL, and UCB). It supports simulations of TCP and UDP.

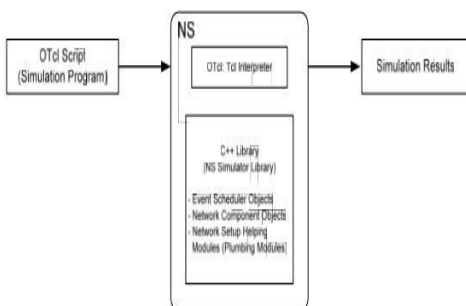


Figure 3: Basic Structure of Network simulator

To setup and run a simulation, a user writes an OTcl script, which is a simulation program to initiate an event scheduler, set up the network topology using the network objects and plumbing functions in the library, and to tell

traffic sources when to start and stop transmitting packets through the event scheduler. When NS-2 which works as OTcl interpreter receives the OTcl script, it will set environment parameters following the received script. If a user wants to make a new network object, it will be easy to make a compound object from the object library, and plumb the data path through the object rather than write a new one. When the simulation is finished, the simulation results are produced in one or more text-based output files that contain detailed simulation data, which can be used to analyze directly or can be used in the graphical user interface “Network Animator (NAM)”.

## III. PROGRAMMING LANGUAGES IN NS-2

The reason for having two programming languages stems from the aim to have an easy to use, yet fast and powerful simulator. Object-oriented C++ forms an efficient class hierarchy core of ns-2 that takes care of handling packets, headers and algorithms. Object Tcl, or OTcl, is also an object oriented programming language utilized in ns-2 for network scenario creation, allowing fast modifications to scenario scripts. Simulation scenario contains network nodes, applications, topology and

connections between the nodes. OTcl and C++ interact with each other through Tcl/C++ interface called TclCI.

Tcl/OTcl is a language with very simple syntaxes that allows easy integration with other languages. Tcl was created by John Ousterhout. The characteristics of these languages are:

- It allows a fast development.
- It provides a graphic interface.
- It is compatible with many platforms.
- It is flexible for integration.
- It is a scripting language.

OTcl in ns-2 enables full control over simulation setup, configuration, and occasional actions (e.g. creating new TCP flows). It is a language that compromise between speed and abstraction level offered to the user. OTcl enables simulation scenarios of slightly varying parameters or configuration, where the main goal is to quickly Xplore a number of scenarios (change the model and re-run). Iteration time is an important aspect in OTcl [11]. C++ object oriented language is used for byte manipulation, packet processing and algorithm implementation. It is important to notice that detailed protocol simulations require the use of systems programming language. With C++ fast execution is enabled, so the Run time speed is important aspects in ns-2. Modularity of ns-2 is observed trough the Class hierarchy core of ns-2 written in C++. Modularity of simulation elements results in a compact simulation model, ns2 does not waste resources of the platform in vain. Compact design of elements opens up new space for the scalability of the elements of a simulation scenario.

#### IV. IMPLEMENTATION OF NS-2 BASED WIRELESS SENSOR NETWORK SIMULATOR

##### A. Basic structure of wireless networks

Wireless networks consist of numerous mobile nodes which communicate with each other via wireless channels, while in wireless sensor networks, these mobile nodes are attached with sensors to sense the physical target and processors to generate data packets from sensing data and transmit them to their neighbours. Wireless network is described as a network of devices which communicate by using wireless technologies wireless is a concept of “relating to data communications using radio waves; operating by means of transmitted electromagnetic waves”.

Wireless communication is used as a term for transmission of information from one place to another without using cables.

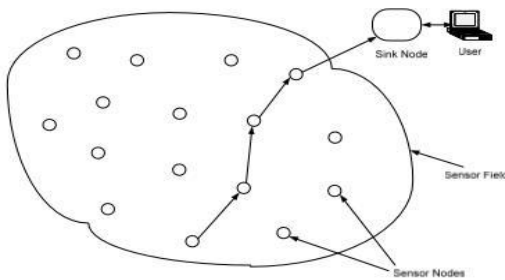


Figure 4: Basic Structure of wireless sensor network

##### B. Simulation model of wireless sensor network

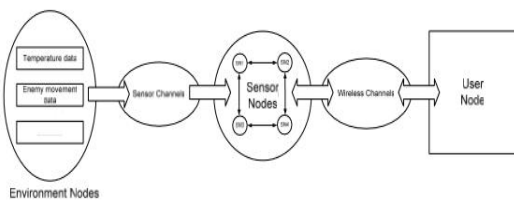


Figure 5: simulation model of sensor network

##### C. Network inside NS-2

The node (terminal, router) in real network have 4 layer using TCP/IP model . Actually, forwarding router have only 2 bottom layer. In the otherwise, 4 layers of TCP/IP model in NS2 are shown in. Two bottom layers are ready automatically when a node setting up. TCP or UDP in transport layer are shown as Agent. FTP, CBR are in application layers. Actually, the sender will set the application for creating data while the receive will not because it is no need to simulate the receiving data in application layer. Realistic network use cable for link between 2 node. One cable is available for biconnection. In NS2, a link is use for one connection.

Two lines are needed for biconnection. Each link has a queue which similar to buffer in realistic network. Packet, which is sent from node should be queuing in queue. When queue is empty, it will be send to other node via link.

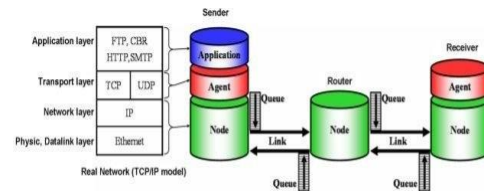


Figure 6: Network inside NS-2

#### V. SIMULATION EXPERIMENT DESIGN

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. Determining the throughput, packet drop rate and average packets end to end delay

The nodes in our scenario use IEEE 802.11 standards (CSMA/CA) to communicate with each other.

We are going to use the simulation for: Determining the throughput for each node; ns-2 should calculate the bytes received by each node.

Determining the packet losses; ns-2 should calculate the bytes that are transmitted and not receive by any node. Determining the average packets end to end delay; ns-2 should calculate the difference time of the last packet received and the number of all packets received.

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.

##### A. The network throughput

In this section we are showing the simulation results illustrating the network throughput. We will see the throughput of nodes to see how nodes affect each other as a result of different emitting times of the node. Figure 7 show the relation between throughput and transmission rate. Here, the X axis represents the time of simulation and Y axis represents the throughput



Figure 7: Throughput for two nodes (Node\_1 is red line and Node\_2 is green line)

*B. Packet drop rate*

Packet drop occurs when one or more packets of data travelling across a computer network fail to reach their destination. Here, the X axis represents the time of simulation and Y axis

represents the number of packets dropped

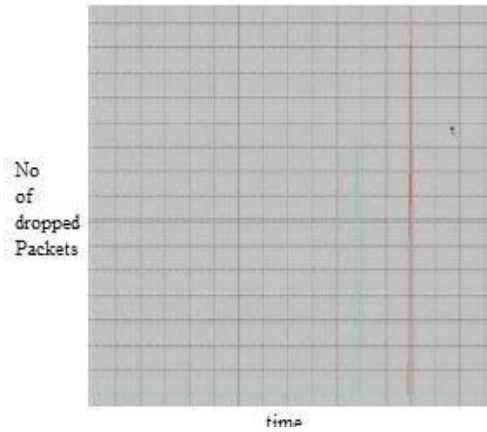


Figure 8: Packet drop for two nodes (Node\_1 is a red line and Node\_2 is a green line)

*C. Average packet 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. Here, the X axis represents the time of simulation and Y axis represents the average delay packets

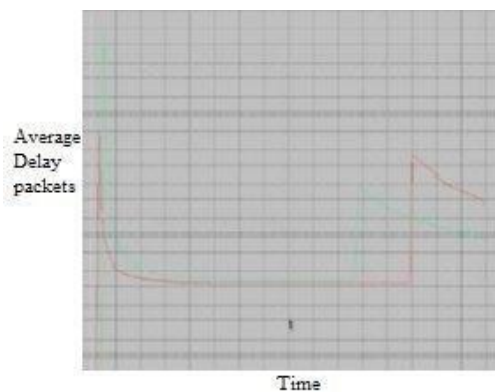


Figure 9: Delay for two nodes (Node\_1 is a red line and Node\_2 is a green line)

**VI . CONCLUSION**

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 behaviour. The results of the simulation are stored in a trace file that records data about all events that occurred during the simulation process. In the master thesis we have used ns-2 to simulate the end user performance of the wireless network consisting of two APs and five nodes for variable throughput and transmission rate for the nodes. The simulation results in following conclusions about network behavior:

- For throughput and transmission rate for the nodes, the performance of network remains constant and there is high fluctuation for single node.
- Another comparative study between packet drop rate and transmission rate for the nodes shows that the performances of the observed networks differ and there is high fluctuation in the node .
- Third important feature of performance study is average packets end to end delay and transmission rate for the nodes is that 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 Wireless Communication, link <http://www.atis.org/>, Archived from the original on 2008-01-02.
- [2] Andrea Goldsmith, Wireless Communications, Cambridge University Press, September 2005, ISBN13: 9780521837163.
- [3] Jim Geier, Wireless Networks first-step, Cisco Press August 03, 2004.
- [4] H. Labiod, H. Afifi and C. Desantis, WI-FI, Bluetooth, Zig Bee and WIMAX, Springer 2007, ISBN 978-1-4020-5397-9.
- [5] IEEE-SA, IEEE 802.11n-2009—Amendment 5: Enhancements for Higher Throughput, 29 October 2009.
- [6] Lawrence Berkeley National Laboratory, link: <http://ee.lbl.gov/>, December 2010.
- [7] Berkeley University, link: <http://www.cs.berkeley.edu/>, December 2010.
- [8] University of Southern Carolina, link: <http://www3.isi.edu/home>, December 2010.
- [9] Eitan Altman and Tania Jimenez, Network Simulator for beginners, December 4, 2003.
- [10] Kevin Fall and Kannan Varadhan, The ns Manual, May 9, 2010.
- [11] Teerawat Issariyakul, Ekram Hossain, Introduction to Network Simulator ns-2, 2009.
- [12] NS-2, link: <http://www.isi.edu/nsnam/ns/tutorial/>, December 2010.
- [13] Estrin, R. Govindan, J. Heidemann, and S. Kumar, "Next century challenges: Scalable coordination in sensor networks" in Proc. 5th Annu ACM/IEEE Int. Conf. Mobile Comput. Netw., 1999, pp. 263–270.
- [14] Gartner Group, Understanding Hype Cycles[Online]. Available: <http://www.gartner.com/pages/story.php.id.8795.s.8.jsp> Estrin,
- [15] D. Culler, K. Pister, and G. Sukhatme, Instrumenting the physical world with pervasive networks, IEEE Pervasive Comput., vol. 1, no. 1, pp. 59–69, Jan.–Mar. 2002.
- [16] W. J. Kaiser, K. Bult, A. Burstein, D. Chang, *et al.*, "Wireless integrated microsensors," in *Technical Digest of the 1996 Solid State Sensor and Actuator Workshop*, 06 1996.
- [17] R.T. Lacoss, "Distributed mixed sensor aircraft tracking," in *American Control Conference, 1987*, pp. 1827–1830, IEEE, 1987.
- [18] J. Pottie, "Wireless integrated network sensors (WINS): the web gets physical," in *Frontiers of Engineering: Reports on Leading-Edge Engineering from the 2001 NAE Symposium on Frontiers of Engineering*, p. 78, National Academies Press, 2002.
- [19] J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," *Communications of the ACM*, vol. 43, no. 5, pp. 51–58, 2000.
- [20] S. Vardhan, M. Wilczynski, G. Portie, and J. Kaiser, "Wireless integrated network sensors (WINS): distributed in situ sensing for mission and flight systems,"