UGC MINOR RESEARCH PROJECT

ON

Performance Evaluation and Implementation of Proposed Routing Protocol for WSN

UGC SANCTION NO: 1396-MRP/14-15/KABA057/UGC-SWRO Dated: 04 February 2015.

Submitted to

ज्ञान-विज्ञान विमुक्तये

UNIVERSITY GRANTS COMMISSION
South- Western Regional Office
Bengaluru

Principal Investigator

Jyothi V

Assistant Professor
Vijaya College
R. V. Road, Basavanagudi
Bengaluru – 560 004

Contents

FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction to WSN:

Wireless Sensor Networks (WSNs) play a key role in sensing, computing and communicating the information in most of the fields bringing substantial improvements in a broad spectrum of modern technologies. The various applications of WSN include battlefield surveillance, environmental monitoring, biological detection, smart spaces, disaster search and rescue, industrial diagnostics, sensing a building integrity or structural vibrations during an earthquake, the stress of an airplane’s wings , etc,.

The WSN as shown in figure 1.1 must possess self-organizing capabilities, since the positions of individual nodes are not predetermined.

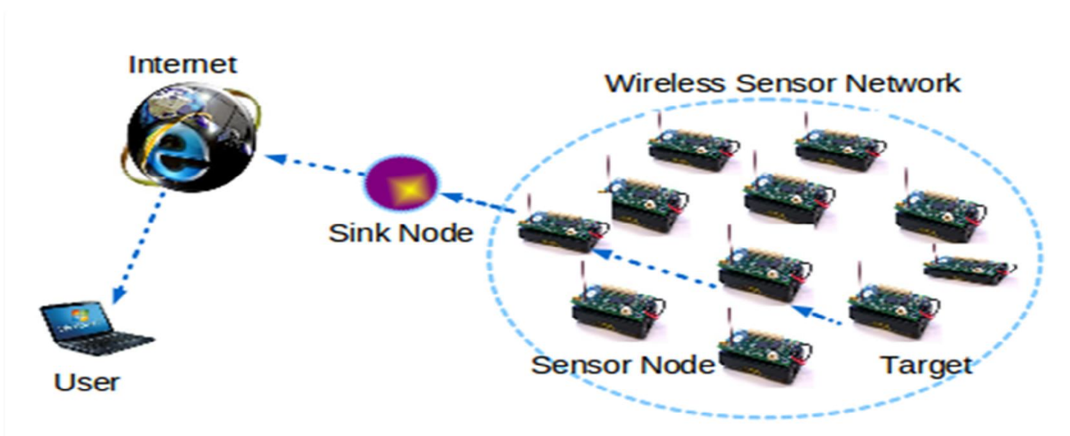

Illustration of Wireless Sensor Networks

Figure 1.1 Wireless Sensor Network

Cooperation among nodes is the dominant feature of the network, where groups of nodes cooperate to disseminate the information gathered in their vicinity to the user. Transmitting raw measured data not only consumes large amount of energy but also increases network traffic which poses high bandwidth demand.

WSNs are composed of large numbers of minimal capacity sensing, computing and communicating devices which are deployed densely in close proximity to the phenomenon to be monitored. These devices operate in complex and real-time environment. Each of the nodes collects data and sends the data to a sink. Most WSNs are multihop networks and rely on communication stacks that include the Medium Access Control (MAC), routing and transport layers.

Routing is the method of selecting an optimal path to transfer the information from the source to the sink (destination) in the network. Routing in WSN depends on many factors including topology, selection of routers and location of request initiator that could serve as an aid in finding the path quickly and efficiently [1, 2]. One of the major requirements in designing a routing protocol is that a node should have the information of its neighbors to reach the destination. An interesting open problem for research is to consider physical layer based routing and broadcasting where nodes may adjust their transmission radii.

The stringent challenges of WSN are mentioned below

- **Fault Tolerance:** The ability to sustain sensor network functionalities without any interruption due to sensor node failures. Sensors nodes may fail due to any of reasons such as lack of power, Physical Damage or environmental interferences.

- **Scalability:** The density can be range from few sensor nodes to few hundred nodes depends on application. The accurate representation in the network is challenging.
- **Production Cost:** Since the sensor network consists of large number of sensor nodes, the cost of single node is very important to justify the overall cost of the networks.
- **Operating Environment:** Nodes are deployed in remote geographic area or unattended place. Designing of nodes under various conditions is a challenging issue.
- **Sensor Network Topology:** Densely deployed nodes require careful handling of topology maintenance. Issues related to topology maintenance in three phases

  - Pre-deployment and deployment phase
  - Post-deployment phase
  - Re-deployment of additional nodes phase

- **Hardware Constraints:** Depending upon application, some other components like, Location Finder (To determine their Position), Mobilizer(To change their location or Configuration i.e. Antenna orientation) (sensing part, microcontroller, transceiver and power unit).
- **Transmission Media:** In a multihop sensor network, communicating nodes are linked by a wireless medium. These links can be formed by radio, infrared or optical media. To enable global operation of these networks, the chosen transmission medium must be available worldwide.

- **Power Consumption:** Based on Architecture , several approaches have been exploited. At a general level classified in to three domains

  - Duty Cycle
  - Data Processing
  - Communication

WSN should satisfy above mentioned stringent challenges which make this field highly potential for active research.

1.2 MANET

MANET has a potential of becoming extremely useful in providing reliable communication services across areas with no pre-existing infrastructure [3]. It guarantees flexibility and convenience by supporting unconstrained mobility. It has the desirable features of future generation networks. However, it does suffer from some of the limitations imposed by its inherent characteristics. Lack of a fixed infrastructure and a dynamic topology has resulted into serious protocol design issues. MANETs are paradigms for wireless communication in which mobile nodes are dynamically and arbitrarily located. Such networks are self-forming and self-organizing as shown in figure 1.2.

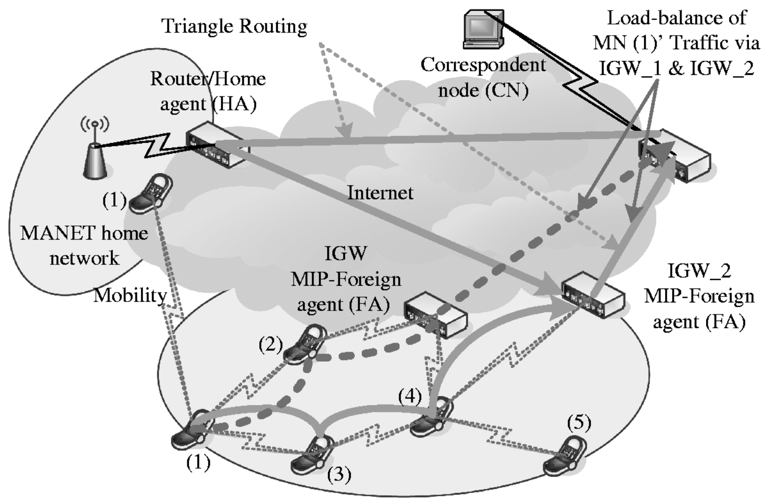

Figure 1.2. Mobile Adhoc Networks

As a technology for dynamic wireless networks, it has been deployed in military since 1970s. Commercial interest in such networks has recently grown due to the advances in wireless communications. A new working group for MANET has been formed within the Internet Engineering Task Force (IETF), aiming to investigate and develop candidate standard Internet routing support for mobile, wireless IP autonomous segments and develop a framework for running IP based protocols in ad hoc networks. The recent IEEE standard 802.11 has increased the research interest in the field.

Research in the area of ad hoc networking is receiving more attention from academia, industry, and government. Since, these networks pose many complex issues, to be addressed with research and significant contributions.

The stringent challenges of MANET are mentioned below

- **Lack of centralized management**: MANET doesn't have a centralized monitor server. The absence of management makes the detection of attacks difficult because it is difficult to monitor the traffic in a highly dynamic and large scale ad-hoc network. Lack of centralized management will impede trust management for nodes.

- **Routing in adhoc topology:** In MANET, the mobility of nodes changes the connectivity between them frequently. Hence, an efficient routing protocol with link stability to be implemented to address this issue to ensure an efficient data packet delivery from source to destination node, thereby improving the throughput.
- **Maintenance of routing:** Regular update of information corresponding to the data packets and the control packets with dynamic links between the nodes in MANET is a major issue to be addressed.
- **Scalability:** In MANETs, the nodes that establish a network are battery powered. Hence, the throughput of the network depends on the parameters namely, input source, processing capability and storage capacity. As the density of nodes increases the battery drain, thereby affecting the lifetime of the network.
- **Energy Efficiency:** The nodes employed in MANETs are battery operated with limited lifetime. The battery of a node is crucial Also, whenever a node gets damaged; recharging or replacing the battery is hard. Further the nodes acts as routers and also as an end device. Hence, a routing protocol to be designed to increase the throughput thereby utilizing minimum energy resource.

With all above mentioned constraints, mobility has been a major limitation to a smooth functioning of a MANET protocol. The increase in node mobility causes increased link distraction and consequently, the performance of the protocol deteriorates as the density of the nodes in the network increases. Further, there exist more control packets generation and the increased control overhead in the network. Furthermore, in a real scenario, mobility of the wireless node is tied to a path fading effect, which further deteriorates the network performance.

1.2.1 Literature survey:

International Status:

Paper [4] deals with head-compression way to simplify the TCP/IP protocol stack, referring to the IPv6 over Low power WPAN (6LoWPAN) technology. Thus, one can make application of TCP/IP in WSN module. The design has been conducted in the Network Simulator 2 (NS2) environment, and stimulation results illustrated that the head-compression mechanism in the WSN module is conducive to improving the message transmission ratio of success, reducing the end to end delaying and decreasing the resource consumption of MAC layer, which satisfies the requirements of WSN.

Authors of [5] have proposed the implementation details of WSN based temperature monitoring application. Its main feature is to continuously monitor the temperature in the 128 node High Performance Computing Cluster for its smooth functioning. The wireless sensor node senses and transmits the current value of temperature to the base station. Also, explains about the various steps involved in the experimental implementation and maintenance of the temperature monitoring network for High Performance Computing cluster at Computer centre, IGCAR.

Authors of [6] have introduced a new design of multi-channel MAC special for cluster-tree WSN, a typical asymmetric network. It has many special features in nodes distribution and data direction. According to these features, phase hopping (control channel and data channel coexist) is used to achieve multi-channel assignment and channel switch and time division multiple address (TDMA) mechanism is utilized to avoid data collision in the period of nodes' communication. The simulation results show that the scheme presented has advantages in providing higher maximum effective data transmission and smaller average packet delay and reducing energy consumption.

In paper [7], authors have implemented an efficient method for finding optimized position of sink node using PSO. The initial constraints in finding optimal Base-Station (BS) locations in two-tiered wireless sensor networks proposed by authors using PSO are relaxed by placing Application Nodes (AN) dynamically based on Euclidean distance and probability of selection. This system is tested by establishing communication between the nodes and sink through the application nodes using query-driven model of WSN.

In paper [8], authors have designed and deployed Wireless sensor network (WSN) in a tea plantation with the purpose of drought monitoring. The sensing nodes were designed, which are composed of an ATmega128L micro-processor, a CC2420 transmit-receive module etc. A WSN gateway layout scheme based on ARM was demonstrated. The networking experiments were conducted in laboratory with Mesh Routing Protocol to examine the network performance. Test results show that the performance of Quality of Service (QoS) declined with network scale extending from 5, 10, 15 to 20 nodes. Simultaneously, the topology formation time, the topology stabilization time and the topology reconstruction time got prolonged. The entrance of a new node did little effects on routing performance of a steady-going WSN. Diamond shaped node deployment was compared and adopted. Tests in tea plantation showed that the system is stable and packet Loss Rate (PLR) goes up when data transmission interval varied from 10s, 20s, 30s to 40s.

Authors of [9] presented a methodology for the optimization of SBST from the energy perspective. The refined methodology presented in this paper is able to be effectively applied in the case that the SBST routines are not initially available and need to be downloaded to the WSN nodes, as well as the case that the SBST routines are available in a flash memory. The methodology is extended to maximize the energy gains for WSN architectures offering clock gating or Dynamic Frequency Scaling features. Simulation results has shown that energy savings at processor level are up to 36.5 percent, which depending on the characteristics of the WSN system, can translate in several weeks of increased lifetime, especially if the routines need to be downloaded to the WSN node.

Jihong Song [10] presented a low energy cost space location system based on WSN technology and random sampling algorithm, such that one can locate a person holding a WSN mote in real time. In this system, by using the representation based on random sampling, authors' location algorithm can represent arbitrary distribution. According to the comparison of the simulation results, authors conclude that the movement trajectory acquired by the system is very close to the real movement trajectory, and the number of location errors does not rely on the number of sampling numbers very obviously.

In paper [11] author presented application programming interface for wireless sensor networks (WSN API) which consists of a client-side API (gateway API) and a sensor-side API (node API). The WSN API conceals the complexities of WSN communication protocols and architectures, and provides a well-defined and easy-to-use way to collect data from sensors. Also, easy expandability for new sensor components and applications is provided. The WSN API is implemented practically in TUTWSN prototype platforms.

In paper [12] authors proposed an Adaptive Weighted Fair Queuing (A-WFQ) by adjusting the weight of a queue according to the queuing delay that relates to the Hurst parameter of self-similar traffic. A-WFQ is found simple to implement in WSN and the simulation results show that proposed algorithm can improve delay and packet loss with service differentiation under self-similar traffic.

In paper[13] author surveyed a set of e-Health proposals for WSN, focusing on the mechanisms used to provide security and privacy to support real-time and multimedia data transmission in WSN.

In [14] authors presented the modeling and simulation of nodes composed by a typical system on chip (SoC) for WSN applications using SystemC/TLM. The SoC contains a MIPS-based processor, a memory, a bus, a timer, a transceiver and a battery. The model is intended to allow power estimation in WSN simulation.

The battery module estimates the energy consumption of the node by computing the amount of electrical charge used by each block as a case study, also performed a star topology WSN simulation, showed the power consumption of each node, and presented a discussion about the computational load. This approach is flexible and can be adapted to simulate more complex systems and topologies.

Paper [15] presents the design and implementation of a practical network monitoring and analysis tool for identifying the causes of misbehavior. The tool consists of a sniffer node that passively captures WSN traffic, and multiple user interfaces that can run e.g. on a PC or mobile phone. Unlike the related proposals, tool neither needs setting up an additional monitoring network alongside the actual WSN nor uses any node resources. Practical testing experiences with multi-hop mesh WSN deployments show that the tool reveals design and implementation defects that are hard to discover with other testing methods such as in-network monitoring systems or debug prints.

Authors of [16] derived an algorithm for energy efficient external and internal intrusion detection. They also analyzed the probability of detecting the intruder for heterogeneous WSN. This paper considers single sensing and multi sensing intruder detection models. It is found that experimental results validate the theoretical results.

In paper [17] authors introduced an enhanced version of SLPv2 (Service Location Protocol version 2), which is one of the service discovery protocols in wired network, and proposed the wireless network based SLP for WSN (wireless sensor network). By transferring wired protocols to wireless domain authors faced critical problems, like resource and power wall, etc. So SLP for the WSN must meet the low power consumption, low rate, and meet the endurable reliability. Thus, suggested the ldquodynamic scope methodrdquo, and the ldquoT_ID check algorithmrdquo for the WSN based SLP. The distance is bounded by the hop count. It not only can reduce the packet exchange dramatically but also can be useful to the WSN in many different ways. The first is the low energy consumption, and second, it can reduce the bottleneck due to the Tree structure of WSN environment. Due to the reduction of bottleneck, it results in the low packet loss, and has a good throughput due to the low probability of packet collision.

In paper [18], it is the introduction of the system's composition structure and Working principle, and detailed explanation about the system's software and hardware design and the routing protocol of WSN based on GIS. Experiments resulted that it is strong practicability and reliability of data transmission, at the same time, compared with the current monitoring system based on GPRS and GPS technology, it is greatly reduced the operating price.

A. National Status:

Authors of [19] proposed partial and full link reversal algorithms to bypass voids during geographic routing over duty-cycled wireless sensor networks. Also, proposed a distributed approach that was oblivious to one-hop neighbor information. Upon termination of the algorithm, the resulting network was guaranteed to be destination-oriented. Further, to reduce the delays incurred under reactive link reversal, they proposed the use of ‘pseudo-events’, a preemptive link reversal strategy that renders the network destination-oriented before the onset of a real event. A simulation study of the effectiveness of pseudo-events was also provided.

Paper [20] shows that nodes mobility can enhance coverage degree for WSN. Analytical model is introduced to describe the coverage degree in mobile nodes wireless sensor network. MATLAB was used to build the simulator. The analytical work is validated by simulated results. Both of analytical model and simulation showing that the coverage degree has been enhanced in Mobile Nodes Wireless Sensor Network (MNWSN) as compared to static network.

Paper [21] introduces a reference implementation of the Institute of Electrical and Electronics Engineers (IEEE) 1451.5-802.11 standard-based wireless sensor network (WSN) developed at the National Institute of Standards and Technology (NIST). The WSN consists of a Network Capable Application Processor (NCAP) and two Wireless Transducer Interface Modules (WTIM). The Network Capable Application Processor (NCAP), a gateway node of the WSN, was developed on a laptop in Java language according to the IEEE 1451.5-802.11 standard. The embedded WTIM, a wireless sensor node, was developed based on the IEEE 1451.5-802.11 standard on a single board computer in Dynamic C language. The wireless communications between the NCAP and WTIMs are based on IEEE 1451.0 messages using Transmission Control Protocol/Internet Protocol (TCP/IP) and User Datagram Protocol/Internet Protocol (UDP/IP) sockets. A few examples are provided to illustrate the functionalities of the WSN.

Authors of [22] described successful attempt to design and deploy a WSN to monitor seabirds on Skomer Island, a UK National Nature Reserve. Summarized the evolution of the system over a period of three years, shared insights on selected design decisions, and discussed both, their experience and the problems they have encountered.

In paper [23], author proposed an in-network mechanism to find frequent sensor patterns in the sensors themselves. So, the sensors send only the frequent senor patterns to the sink, not the sensor activity sets.

Authors of [24] proposed a new cluster head selection method for LEACH clustering routing protocol. It balances the energy consumption of every sensor node in a sensor network. Simulation results shown that the lifetime of sensor network is significantly enhanced compared to the LEACH routing algorithm for the wireless sensor networks.

In paper [25] authors aimed at optimizing the value of R by dynamically increasing or decreasing it according to the distance the border node takes from the border region. Also expected dynamic approach increases the energy efficiency by preventing the nodes inside the virtual clusters going into the active zone unnecessarily but yet at the cost of an increased latency.

In paper [26] authors implemented IEEE 802.15.4 in an unattended scenario to achieve that, cluster head rotation is needed to avoid the problem single point of failure (i.e. when the battery of the cluster head is depleted, all the nodes in the cluster will not be able to forward their data any more). To maintain the features of this protocol while implementing it in an unattended scenario, authors proposed to use a long fixed hyperframe for periodic setup phase. This hyperframe contains smaller variable superframes. In this way, some applications used in an unattended scenario may exploit the characteristics of IEEE 802.15.4 standard.

In paper [27] authors proposed a novel multi-sensor vector prediction history tree (V-PHT) decision algorithm for error correction in a wireless sensor network (WSN). This scheme is based on the recently proposed prediction history tree (PHT) algorithm for model based error correction in WSNs. However, unlike the existing PHT model, which exclusively exploits the temporal correlation inherent in the narrowband sensor data, the proposed V-PHT model for sensor data correction exploits the joint spatial and temporal correlation in sensor data arising out of geographical proximity of the sensor nodes. Towards this end, an optimal multi-sensor spatio-temporal AR model is developed for predictive modeling of the sensor data. Further, the spatio-temporal correlation structure employed amongst the sensors, develop a robust framework for optimal estimation of the multi-sensor AR predictor model. Simulation results obtained employing sensor data models available in literature demonstrate that the proposed spatio-temporal V-PHT model for error correction in a WSN results in a significant reduction in mean-squared error (MSE) compared to the existing PHT scheme which exploits only temporal correlation.

Authors of [28] proposed a centralized scheduling algorithm within 6PANview which non-intrusively analyzes application traffic arrival patterns at the base station, identifies network idle periods and schedules monitoring activities. The proposed algorithm finds those periodic sequences which are likely to have given rise to the pattern of arrivals seen at the base station. Parts of those sequences are then extended to coarsely predict future traffic and find epochs where low traffic is predicted, in order to schedule monitoring traffic or other activities at these times. Simulation results for the proposed prediction, scheduling algorithm and its implementation as a part of 6PANview. As an enhancement, authors briefly talk about using 6PANview's overlay network architecture for distributed scheduling.

In paper [29], an efficient key management scheme based on public key elliptic curves signcryption (ECS) scheme for Wireless Sensor Networks (WSN) has been proposed. The proposed algorithm is compared with the scheme given in. For 250 cluster node, the saving in the total number of operations, key storage requirements, energy consumptions, and communication overhead are 75%, 96%, 23.79 m Joule, and 40% respectively compared with. The proposed protocol is efficient in terms of complexity, number of message exchange, computation, and storage requirements compared with. The proposed key management possesses not only provided confidentiality, authentication, integrity but also unforgeability.

In paper [30], energy efficient Adaptive number of Frames per Round (AFR) protocol is proposed for a WSN where cluster heads are evenly distributed. The proposed AFR protocol combined with the cooperative Multiple Input Single Output (MISO) technique is extended for dual-hop wireless sensor networks to improve the network lifetime. The selection of the cooperative relay then explored so that the unexpected partition in network is avoided. Different cooperative relay positions have been investigated for decreasing and balancing the energy consumption in the network. Comparison is made for AFR protocol with and without dual-hop cooperative MISO scheme as well as with some other existing protocols.

In paper [31], Energy level performances of two packet delivery schemes in random WSN (Wireless Sensor Networks) are evaluated in presence of Rayleigh fading. Two different information delivery mechanisms involving regenerative relays with or without error correcting capability are investigated. Energy consumption for successful delivery of a data packet for each mechanism is evaluated and compared under several conditions of node density, bit rate, transmit power and channel fading. Energy efficiencies of the above schemes in a random WSN are also evaluated. In a random network, an intermediate node in the route selects the nearest node within a sector of angle (called search angle) towards the direction of the destination as the next hop. The impact of search angle on energy consumption, energy efficiency and BER are also investigated. Further an optimal packet length based on energy efficiency is derived. Impact of optimal packet size on total energy expenditure is analyzed for each delivery scheme.

1.2.2 Applications of MANET

- **Military services:** Military services are one of the critical areas of application, where deployment of any fixed infrastructure is difficult in the rival territories. In such situation, the end devices of the network are connected to the soldiers whereby the soldiers act as mobile nodes and establish connectivity within the radio range of communication. Also, coordination between the military object and the workforce in the battle field.
- **Sensor Network:** It is a part of MANET where the nodes are stationary in a defined terrain. The limitation of the network is that the nodes are battery powered and hence, has constrained of energy to be addressed. Here, each sensor nodes are equipped with transceiver, a microcontroller and a battery as source. The sensor

nodes relay the observed data corresponding to physical parameter to the access point which is a monitor through intermediate nodes or directly to the access point if it placed within the radio range of the sensor node. Some of the physical parameter that can be measured is temperature, humidity, pressure etc. Some of the sensors are also attached to the patients in biomedical applications.

- **Personal Area Network:** The communication of data within a distance of 10meters is established through PAN. The devices used to communicate can be a mobile phone or laptops etc. It can also be used for adhoc communication among the devices or for connection to a backbone network.
- **Education:** Sensors are equipped for video conferencing. A network is established to monitor the campus of an educational institution or a university. A virtual class rooms can also be established using sensor nodes.
- **Emergency services:** The natural disasters where entire communication collapses and in that scenario restoring the communication quickly is essential. By adopting adhoc network, one could establish the setup within few days required for wireless communication.

Thus, a robust protocol capable of routing effectively within a highly mobile topology and without compromising its inherent attributes is important for efficient deployment of a MANET. In other words, an improved routing protocol is needed to be designed, in which the protocol keeps information about the speed of the intermediate nodes and adopts this information to establish a more stable routing path with the inclusion of minimal overhead.

1.3 REFERENCES

[1] Perkins C., "Ad-Hoc Networking", Addison Wesley, 2001.

[2] M.Subramanyabhat, D.Swetha and J.T. Devaraju, "A Performance Study of Proactive, Reactive and Hybrid Routing Protocols using Qualnet Simulator", IJCA (0975 - 8887). Vol-28(5), August 2011.

[3] D. B. Johnson and D. A. Maltz, "*Dynamic source routing in ad hoc wireless networks. Mobile Computing*", 153, 1996.

[4] Yu Xiao, Xiaoyan Cui, Hang Li, Teng Xi , "A protocol simplifying mechanism for a WSN Module", International Conference On Electronics and Information Engineering (ICEIE), 2010, Volume: 2, DOI: 10.1109/ICEIE.2010.5559820 , Page(s): V2-474 - V2-477.

[5] D. Baghyalakshmi, J. Ebenezer, S.A.V. Satyamurty, "WSN based temperature monitoring or High Performance Computing cluster", International Conference on Recent Trends in Information Technology (ICRTIT), 2011, DOI: 10.1109/ICRTIT.2011.5972379 , Page(s): 1105 – 1110.

[6] Hong Zhang, Shuyu Li, "A practical design of multi-channel MAC for cluster-tree WSN" , 6th International Forum On Strategic Technology (IFOST), 2011, DOI: 10.1109/IFOST.2011.6021133 Page(s): 761 – 764.

[7] R. Alageswaran, R. Usha, R. G.Gayathridevi, Kiruthika," Design and implementation of dynamic sink node placement using Particle Swarm Optimization for life time maximization of WSN applications", International Conference Advances in Engineering, Science and Management (ICAESM), 2012, Page(s): 552 – 555.

[8] Daozong Sun, Sheng Jiang, Weixing Wang, Jingchi Tang, " WSN Design and Implementation in a Tea Plantation for Drought Monitoring", International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2010, DOI: 10.1109/CyberC.2010.36, Page(s): 156 – 159.

[9] A.Merentitis, N.Kranitis, A.Paschalis, D.Gizopoulos, "Low Energy Online Self-Test of Embedded Processors in Dependable WSN Nodes" Dependable and Secure Computing, IEEE Transactions, D O I: 10.1109/TDSC.2011.17, Page(s): 86 – 100.

[10] Jihong Song, “The application of WSN technology in the space location system”, 4th International Congress on Image and Signal Processing (CISP), 2011, Volume: 2 ,DOI:0.1109/CISP.2011.6100316 , Page(s): 633 – 636.

[11] J.K.Juntunen, M.Kuorilehto, M.Kohvakka, V.A.Kaseva, M.Hannikainen, T.D.Hamalainen, “ WSN API: Application Programming Interface for Wireless Sensor Networks”, 17th International Symposium on Personal, Indoor and Mobile Radio Communications, 2006, IEEE, DOI: 10.1109/PIMRC.2006.254073, Page(s): 1 – 5.

[12] Heng Liu, Fang Ju, “A novel packet scheduling algorithm based on self-similar traffic in WSN” IET-WSN. IET International Conference On Wireless Sensor Network, 2010, DOI: 10.1049/cp.2010.1072, Page(s): 312 – 316.

[13] M.de los Angeles Cosio Leon, J.I.N.Hipolito, J.L.Garcia, “A Security and Privacy Survey for WSN in e-Health Applications” Electronics, Robotics and Automotive Mechanics Conference, 2009. DOI: 10.1109/CERMA.2009.47, Page(s): 125 – 130.

[14] H.M.G.;Madureira, J.E.G.;de Medeiros, J.C.da Costa, G.S. Beserra, “System-level power consumption modeling of a SoC for WSN applications” , 2nd International Conference on Networked Embedded Systems for Enterprise Applications (NESEA), 2011, IEEE ,DOI: 10.1109/NESEA.2011.6144934, Page(s): 1 – 6.

[15] M.Hanninen, J.Suhonen, T.D.Hamalainen, M.Hannikainen, “Practical monitoring and analysis tool for WSN testing” Design and Architectures for Signal and Image Processing (DASIP), 2011, DOI: 10.1109/DASIP.2011.6136894, Page(s): 1 – 8.

[16] T.M.Mubarak, S.A.Sattar, G.A.Rao, M.Sajitha, ”Intrusion detection: An energy efficient approach in heterogeneous WSN”, International Conference on Emerging Trends in Electrical and Computer Technology (ICETECT), 2011, DOI: 10.1109/ICETECT.2011.5760281, Page(s): 1092 – 1096.

[17] Moon Tae Kwon, Ryu Jae Tek, Kim Ki Hyung, Yoo Seung Wha, “A new SLP based power efficient algorithm for the Wireless Sensor Network (wsn)” 9th International Symposium on Communications and Information Technology, 2009. ISCIT 2009. DOI:0.1109/ISCIT.2009.5341220, Page(s): 402 – 403.

[18] Deng Tianmin, Sha Yao-yao, "Design of the intelligent public transportation monitoring system based on WSN" International Conference on Consumer Electronics, Communications and Networks (CECNet), 2011 DOI: 10.1109/CECNET.2011.5768257, Page(s): 4024 – 4027.

[19] S. Ramachandran, S.V.R.Anand, M.Hegde, A.Kumar, R.Sundaresan, "Neighbor oblivious link reversal over duty-cycled WSNs" National Conference on Communications (NCC), 2010, DOI: 10.1109/NCC.2010.5430167, pp: 1 – 5.

[20] S.Awwad, N.K.Noordin, M.F.A.Rasid, "Coverage Degree in Mobile Nodes Wireless Sensor Network", 6th National Conference on Telecommunication Technologies 2008 and. NCTT-MCP 2008. DOI: 10.1109/NCTT.2008.4814286, Page(s): 273 – 277.

[21] E.Y.Song, K.B.Lee, S.E.Fick, A.M.Donmez, "An IEEE 1451.5–802.11 standard-based wireless sensor network with embedded WTIM" Instrumentation and Measurement Technology Conference (I2MTC), 2011 IEEE, DOI: 10.1109/IMTC.2011.5944344 , Page(s): 1 – 6.

[22] T.Naumowicz, R.Freeman, H.Kirk, B.Dean, M.alsyn, A.Liers, A.Braendle, T.Guilford, J.Schiller,"Wireless Sensor Network for habitat monitoring on Skomer Island", IEEE 35th Conference on Local Computer Networks (LCN), 2010, DOI: 10.1109/LCN.2010.5735827, Page(s): 882 – 889.

[23] Das, Anjan, "A novel association rule mining mechanism in wireless sensor networks ", 3rd National Conference on Emerging Trends and Applications in Computer Science (NCETACS), 2012, DOI: 10.1109/NCETACS.2012.6203273, Page(s): 274 – 276.

[24] R.K.Tripathi, Y.N.Singh, N.K.Verma, "N-LEACH, a balanced cost cluster-heads selection algorithm for Wireless Sensor Network", National Conference on Communications (NCC), 2012, DOI: 10.1109/NCC.2012.6176825, pp:1-5.

[25] N.Mouzehkesh, N.K.Noordin, M.Rasid, "Maximizing the Energy Efficiency in MS-MAC Using a Dynamic Approach", 6th National Conference on Telecommunication Technologies 2008 and 2008 2nd Malaysia Conference on Photonics. NCTT-MCP 2008. DOI: 10.1109/NCTT.2008.4814288, Page(s): 284 – 288.

[26] B.Munajed, M.Drieberg, N.H.B.Hamid, “ IEEE 802.15.4 in an unattended scenario with periodic setup phase for wireless sensor networks” National Postgraduate Conference (NPC), 2011, DOI: 10.1109/NatPC.2011.6136431, Page(s): 1 – 5.

[27] A.Jaiswal, A.K.Jagannatham, “Multi-sensor spatio-temporal vector prediction history tree (V-PHT) model for error correction in Wireless Sensor Networks”, National Conference on Communications (NCC), 2012, DOI: 10.1109/NCC.2012.6176806 , Page(s): 1 – 5.

[28] A.R.Bhadriraju, S.Bhaumik, Y.S.Lohith, M.C.Brinda, A.Svr, M.Hegde, “ 6PANview: Application performance conscious network monitoring for 6LoWPAN based WSNs” National Conference on Communications (NCC), 2012, DOI: 10.1109/NCC.2012.6176753, Page(s): 1 – 5.

[29] E.A.A.A.Hagras, D.El-Saied, H.H.Aly, “Energy efficient key management scheme based on elliptic curve signcryption for Wireless Sensor Networks”, 28th National Radio Science Conference (NRSC), 2011, DOI: 10.1109/NRSC.2011.5873614, Page(s): 1 – 9.

[30] M.Harish, B.Srikanth, R.Bhattacharjee, “An Energy efficient WSN with cooperative relaying technique”, National Conference on Communications (NCC), 2012, DOI: 10.1109/NCC.2012.6176757, Page(s): 1 – 5.

[31] A.Nandi, S.Kundu, “Energy level performances of data services in random WSN with Rayleigh fading”, National Conference on Communications (NCC), 2012, DOI: 10.1109/NCC.2012.6176827, Page(s): 1 – 5.

CHAPTER 2

OVERVIEW OF PERFORMANCE EVALUATION TOOLS

2.1 Introduction

The performance evaluation provides results that are repeatable, allowing for detailed comparisons and careful sensitivity analysis in the design. Performance is a key standard in the design, procurement, use of computer systems and networks. The researchers must specify the requirements needed to study the performance of the design and should be able to compare different substitutes to find the one that efficiently meets the requirements [1].

Performance evaluation is needed at several stages in design phase of a standard. In early stages, when the design is being conceived, performance evaluation is used to make early design tradeoffs. Usually, this is accomplished by simulation models; with the aid of simulation results, several design decisions are made which is followed by the prototype building. Once the design is finalized and is being implemented, simulation is used to evaluate functionality and performance of subsystems of the standard. Later, performance measurement is done after the product is available in order to understand the performance of the actual system to various real world workloads and to identify modifications to incorporate in future designs.

The evaluation requires a thorough knowledge of the system being modelled and selection of the efficient methodology and tools. A performance evaluation is required when a system designer compares the alternative designs and concludes one of the designs as the best for the specified set of applications. Since software applications are enormous, it is difficult to have a standard measure of performance and a standard technique for measurement in general. There are two basic categories of performance measures; the first type is system oriented performance evaluation which is independent of specific applications and the second type is the application oriented methods which may be dependent on system measures. For instance, if file transferring over internet application is considered, application measures include packet delivery ratio, throughput, delay, jitter etc.

2.2 Methods of Performance Evaluation

Desired metrics are obtained from measurements of the system measured in either an operational or a controlled environment. This technique inherently yields the most authentic results. However, measurement based evaluation is a very expensive technique and moreover since it requires an actual system, evaluation is not applicable during system design.

The design and validation of network are categorized into three experimental techniques namely: analytical modelling, simulation and measuring a real system. The selection of the suitable technique depends on the system under consideration, accessibility of model and resources available. In analytical method, the model of the system is designed using mathematical concepts and notations.

In simulation method, several software programs are implemented to adopt a model of the system. The main aim of the performance study is either to compare different alternatives or to find the optimal parameter value.

In the third type, measurements are possible only if standard system is already available similar to the proposed system. If it is an innovative concept, analytical modelling and simulation are the only techniques to be adopted to get a perfect validation of the result [1].

2.2.1 Analytical Modelling

In analytical modeling, a system under study is mathematically described by the application of mathematical tools such as queuing and probability theories and then by numerical methods to gain insight from the developed mathematical model. It is a simple method but the results obtained from it are not reliable many times, as it assumes a simple modeling of the real system. It provides a cost effective method of network dimensioning and helps in understanding the relationship among the system, traffic and performance. The modeling phase is a bit difficult part in this technique, as the system parameters and performance metrics have to be selected properly.

Advantages of analytical modeling

- Cost-effectiveness.
- Though it is a simple abstract of the real system, can provide the qualitative insights into the system [3].
- Recommended for the simple and relatively small systems.

Disadvantages of analytical modeling

- For complex systems, exact modeling is difficult
- A strong background in mathematics and probability theory needed [3].
- Analytical models may not give an accurate representation of the real system.

2.2.2 Simulation Model

Simulation is a technique where a software program models the behavior of a network either by calculating the interaction between the different network entities (hosts/packets, etc.) using mathematical formulae or capturing and playing back observations from a production network. It provides a virtual environment wherein the characteristic of the network is analyzed before deploying in the real scenario or environment. The behavior of the network and the various applications and services it supports can then be observed in a test lab [4].

Using a high performance, high fidelity discrete event simulation engine, extremely accurate *virtual* models of a communications environment can be created, and then analyzed through a series of scenarios to identify where there are weak points or failure modes. This *lab-based risk reduction* methodology is repeatable, verifiable and highly cost effective.

With the help of efficient simulation models, Randomness and Probabilities can be introduced, which makes the simulation model more realistic. It is sometimes possible to design simulation models to find the optimal solution with a high level of assurance.

Advantages of simulation model

- Simulation results are reproducible compared to direct measurement results.
- All performance parameters are monitored.
- It provides a better approach and visualizing effects of performance evaluation.

Disadvantages of simulation model

- A long simulation time is required in order to achieve a desired statistical accuracy for some experiments; in some cases it is impossible to get meaningful results within reasonable times.
- To cover a meaningful fraction of the whole parameter space.
- Model setup may take a long time, also validation and verification time can be significant. Validation makes sure that the assumptions behind the model

suit the real system's behavior and verification confirms that the actual code of the simulation model fits the claimed model assumptions.

2.2.3 Emulation

Network emulation is a high fidelity abstraction model which mimics the functions and behavior of real system so that it appears like the system to be studied, communicates and behaves like a real system. Emulation results are reliable and cost effective technique to reproduce the real system.

In a network emulator, the computers can be attached to the emulator through LAN cable or WiFi network. The computers which are attached to emulator act as end nodes (eg. like user equipments) and will act exactly as they are attached to a real network. Emulation results are reliable and cost effective technique to reproduce the real system. An emulator is similar to a simulator, except that instead of redefining many levels of the open systems interconnection (OSI) model, an emulator only redefines the particular layer, for e.g. if the real time traffic is applied then application layer. Hence all the other features of the node participating in emulation remains same, except the separately modeled layers' features. This solution offers an adequate middle ground between simulators and testbeds [5]. Required feature of the experiment can be emulated in hardware, allowing for better performance and accuracy compared to simulators. At the same time, an emulator eliminates many of the practical problems with testbeds by completely controlling external factors that may influence an experiment [6].

2.3 Simulation and Emulation Tools

The network simulation tools are used to model the network performance in order to verify the distributed functions, debugging the network protocols and to test the reliability of new components [6]. A number of network simulators are available; which are categorized as commercial network simulator tools and open source simulation tools. The list of available simulation tools are mentioned in table 2.1.

Table 2.1 Network simulators/ Emulators

Type	**Network simulators/Emulators**
Commercial	QualNet, EXata, OPNET
Open source	NS2, NCTUns, OMNET++, NS3

Though there are many simulation/emulation tools are available, the choice of simulator should be made based on two key aspects: firstly, the correctness of the simulation models and secondly, the suitability of a particular tool to implement the model. A “correct” model based on solid assumption is mandatory to derive trustful results. The fundamental tradeoff is: precision and necessity of details versus performance and scalability. Considering these two key aspects, the Scalable Networks Technologies (SNT) Inc has developed QualNet simulation platform and EXata Emulation platform [7]. Since QualNet and EXata give reliable results as well as network scalability, these two tools are used for the performance evaluation of proposed algorithms in this research work.

2.3.1 Network Simulator version-2 (NS-2)

NS-2 is a discrete event simulator which provides support for simulation of Transmission Control Protocol (TCP), routing and multicast protocols over wired and wireless (local and satellite) networks. The Network Simulator began as a variant of the REAL network simulator in 1989 and has evolved over the past years. NS-2 has been developed under the Virtual Inter Network Testbed (VINT) project in 1995 and it is a joint effort by people from University of California at Berkeley, University of Southern California's Information Sciences Institute, Lawrence Berkeley National Laboratory and Xerox Palo Alto Research Center. The original NS is an object-oriented, discrete event driven simulator targeted at networking research and it is based on REAL network simulator [11]. The current NS project is sponsored by the Defence Advanced Research Projects Agency (DARPA) and the National Science Foundation. NS-2 is the second version of NS. A Network AniMator (NAM) provides packet-level animation and protocol specific graph for design and debugging of network protocols.

In NS-2, to create the topology of the network for simulation some of topology generators may use Inet Topology Generator, GT-ITM (Georgia Tech Internetwork Topology Models) topology generator or Tiers Topology Generator and convert their outputs to NS-2 format. Generation of topologies by hand is another option. The simulation event scheduler of the simulator, contained in OTcl script interpreter, is either a non-real-time scheduler or a real time scheduler which is mainly used for real-time synchronisation of an emulated network. The user indicates in the event scheduler when network elements should start or stop transmitting packets. In order to visualise a network simulation in NS-2, traffic and movement patterns should be generated and references as inputs into the OTcl code configuring the simulation scenario.

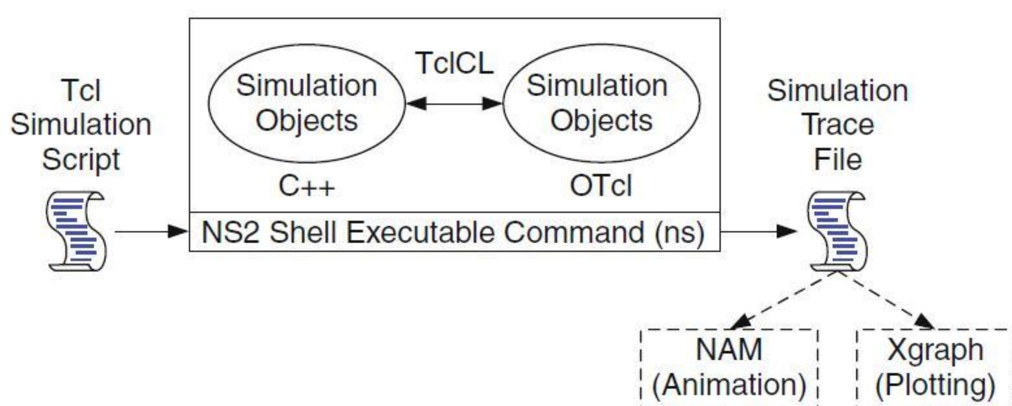

Figure 2.1. Architectural view of NS2

Features of NS-2

- NS-2 supports deterministic or probabilistic packet loss in queues attached to network nodes as well as it supports deterministic and stochastic modelling of traffic distribution.
- NS-2 can be connected to a real network and capture live packets just like a common node. It can also inject packets into the live network.
- The simulator can generate personalised trace files by allowing users to select parameters to be traced, therefore saves CPU resource.
- NS-2 offers a comprehensive documentation and a regularly updated manual as well as an Application Programming Interface (API) for C++ and OTcl classes.
- Other features of the simulator include models for different network architectures including Wireless Local Area Network (LAN), Mobile Adhoc Networks (MANET) and satellite, built-in traffic models with support for development of new ones, plugging of new pseudo random number generators and network state estimation.

2.3.2 Network Simulator version-3 (NS-3)

Some of the main goals included are building better support for network emulation and reuse of implementation code, to integrate the tool with testbed-based research. In the process of developing NS3, it was decided to abandon backward compatibility with NS2, mainly due to the high maintenance overhead that would have resulted.

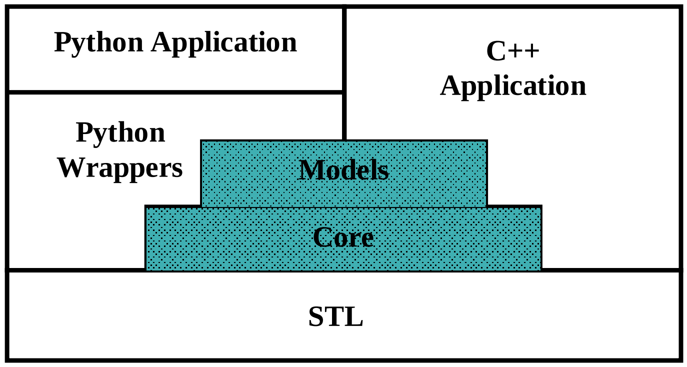

Figure 2.2. Architecture of NS3

The new simulator shown in figure 2.2 is built using C++ and Python and scripting is available with both languages. These automatically generated C++ files are finally compiled into the NS-3 python module to allow users to interact with the C++ NS-3 models and core through python scripts. NS-3 is licensed under the GNU GPLv2 license and is available for research and development. The first release, NS-3.1 was made in June 2008 and afterwards the project continued making quarterly software releases and more recently has moved to three releases per year. NS-3 made its fifteenth release (NS-3.15) in the third quarter of 2012. NS-3 is often criticized for its lack of support for protocols which were supported in NS-2, as well as for the lack of backward compatibility with NS-2. As with NS-2, NS-3 is also cumbersome to learn and use compared to Graphical User Interface (GUI) based simulators [12]. NS-3 redesigns a lot of mechanisms based on the successful and unsuccessful experiences of NS-2.

2.3.3 Java Simulator (J-Sim)

J-Sim is a network simulator written in Java and is built according to the component based software paradigm, which is called Autonomous Component Architecture (ACA). In J-Sim, each component can be atomic or composed of other components interconnected through ports. J-Sim ports support one-to-one,

one-to-many and many-to-many connections. J-Sim protocol architecture comprises of two layers, the lower layer called Core Service Layer (CSL) consists of network to physical layers and the higher layer consists the remaining of Open System Interconnection (OSI) layers.

Although J-Sim does not have tool for network visualization, it allows generating trace files which conform to NS2's NAM format. J-Sim offers good introductory material with overviews and examples for small scenarios. However, it lacks a comprehensive manual. J-Sim uses Tcl for configuration of simulation scenarios which requires a certain learning overhead. Figure 2.3 depicts Schematic of a wireless node in J-Sim.

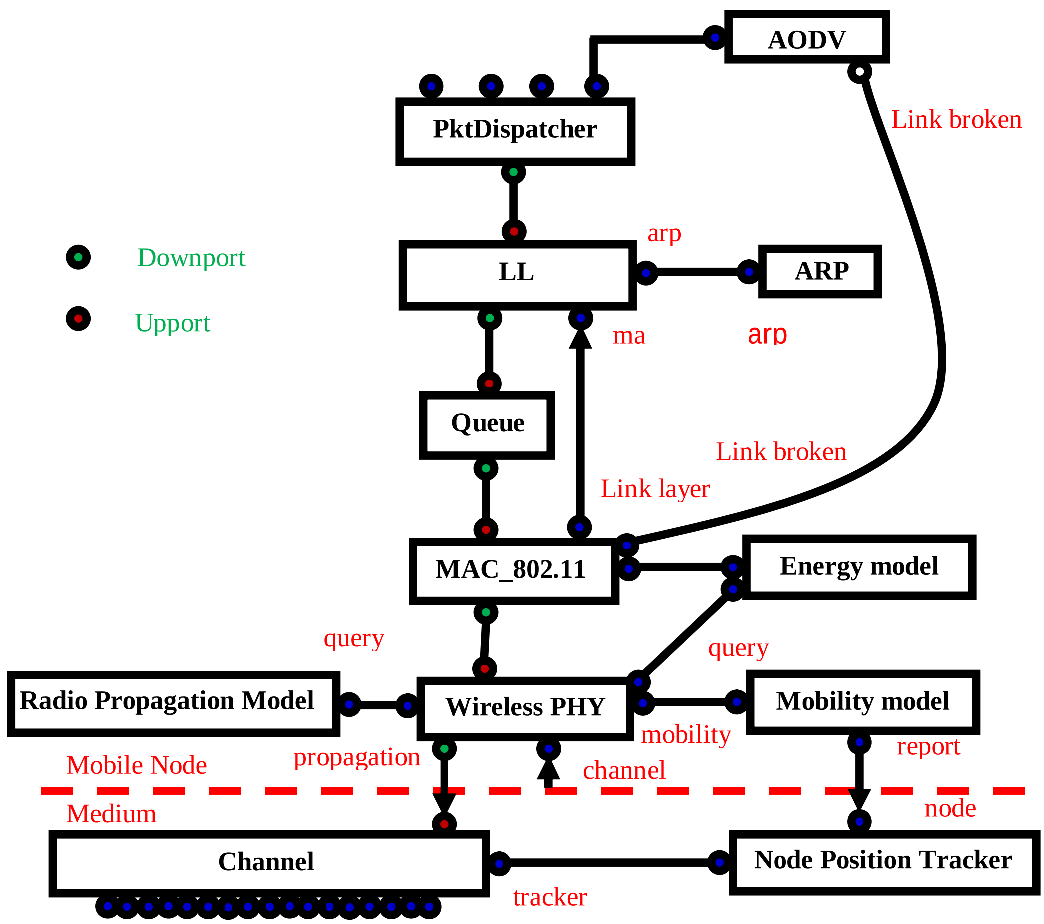

Figure 2.3. Schematic of a wireless node in J-Sim

2.3.4 Optical Micro-Networks (OMNeT++)

OMNeT++ is an extensible, modular, component-based C++ simulation library and framework, primarily for building network simulators. OMNeT++ provides component architectures for models, which are programmed in C++ and then assembled into larger components/models.OMNeT++ is not a simulator in itself but rather a simulation framework. Instead of containing explicit and hardwired support for computer networks or other areas, it provides the infrastructure for writing such simulations. Specific application areas are catered by various simulation models and frameworks, most of them are open source. These models are developed completely independently of OMNeT++ and follow their own release cycles.

OMNeT++ has a well-written fairly large user manual and Application Programming Interface (API) documentation. OMNeT++ is the only simulator with online visualization allowing users to pause the simulation and inspect or even directly change values in the models. It is also possible to change a node's appearance (color, size, shape, etc.) to reflect an inner state which the user wants to visualize. Statistics can be written to a trace file and displayed with external but commonly available tools like *prove*. OMNeT++ is very complex, thus careful consultation of the available documents is needed.

2.3.5 Global Mobile information system Simulator (GloMoSim)

GloMoSim is a scalable simulation environment for wireless network systems. It is being designed using the parallel discrete-event simulation capability provided by PARSEC. Most network systems are currently built using a layered approach that is similar to the OSI seven layers network architecture. Standard Application Programming Interfaces (APIs) are used between the different simulation layers.

This will allow the rapid integration of models developed at different layers by different people. The goal is to build a library of parallelised models that can be used for the evaluation of a variety of wireless network protocols. The proposed protocol stack will include models for the channel, radio, MAC, network, transport and higher layers. GloMoSim stopped releasing updates in 2000. Instead, it is now updated as a commercial product called QualNet [3].

Features

- GloMoSim allows the simulation scalability to simulate networks with a hundreds and thousands of nodes.
- GloMoSim supports protocol for the wireless networks.
- GloMoSim provides the Random Waypoint mobility model, which may not be suitable for all types of simulations.
- The Bonn Motion software provides a generator for other kinds of mobility models.
- GloMoSim is designed to be extensible, with all protocols implemented as modules in the GloMoSim library.

2.3.6 REAL

REAL is a network simulator used for studying the dynamic behaviour of flow and congestion control schemes in packet switched data. It provides users a way of specifying such networks and to simulate their behaviour. It provides around 30 modules that exactly emulate the actions of several well known flow control protocols and five research scheduling disciplines. The modular design of the system allows new modules to be added to the system with little effort. Source code is provided so that interested users can modify the simulator to their own purposes. Online documentation and source code are part of the distribution [7].

2.3.7 Optimised Network Engineering Tools (OPNET)

Optimized Network Engineering Tools (OPNET) is one of the popular commercial network simulators comprising a suite of protocols and technologies [16]. OPNET provides a sophisticated development environment which is specialized for network research and development. It can be flexibly used to study communication networks, devices, protocols and applications. OPNET offers relatively much powerful visual or graphical support for the users. The graphical editor interface can be used to build network topology and entities from the application layer to the physical layer. Object-oriented programming technique is used to create the mapping from the graphical design to the implementation of the real systems; an example of the graphical user interface (GUI) of OPNET can be seen in fig.2.4.

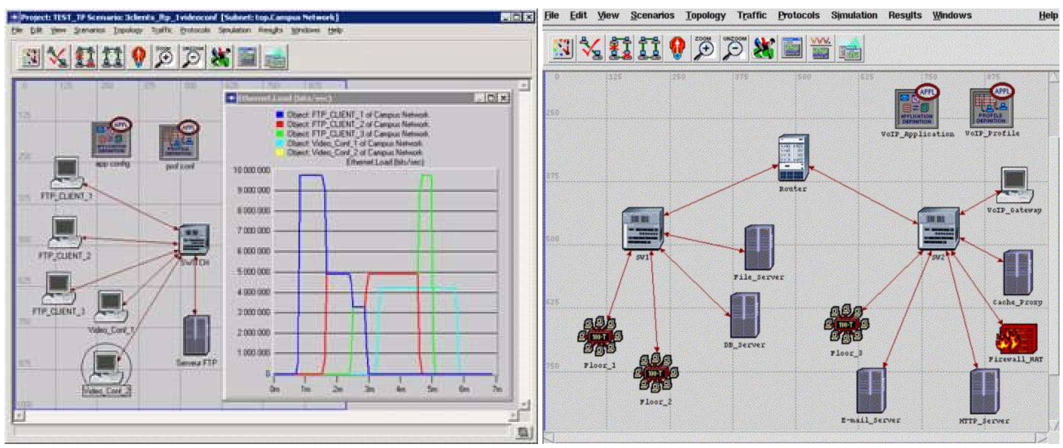

Figure 2.4. GUI of OPNET

In OPNET all the topology configuration and simulation results can be presented very intuitively and visually. The parameters can also be adjusted and the simulations can be repeated easily through GUI. OPNET is based on a mechanism called discrete event system which means that the system behavior can simulate by modeling the events in the system according to the order of scenarios set up by user. Hierarchical structure is used to organize the networks. As other network simulators, OPNET also provides programming tools for users to define the packet format of the protocol. The programming tools are also required to accomplish the tasks of defining the state transition machine, defining network model and the process module. As of all, OPNET is one of a popular simulators used in industry and academia for network research and development.

The main features of OPNET are listed below,

- Fast discrete event simulation engine
- Lot of component library with source code
- Object-oriented modeling
- Hierarchical modeling environment
- Scalable wireless simulations support
- 32-bit and 64-bit GUI
- Customizable wireless modeling
- Discrete event, hybrid and analytical simulation
- 32-bit and 64-bit parallel simulation kernel
- Grid computing support
- Integrated, GUI-based debugging and analysis
- Open interface for integrating external component libraries

2.3.8 QualNet

The QualNet communications simulation is a planning, testing and training tool that "mimics" the behaviour of a real communications network. QualNet provides a comprehensive environment for designing protocols, creating and animating network scenarios and analysing their performance to improve their design, operation and management [8]. QualNet is categorized into QualNet Runtime and QualNet Developer. QualNet Runtime provides tools for design, simulation, visualisation and analysis of networks. It is a binary-only (executable) product. Users can create scenarios, modify protocol parameters and simulate the network's performance. QualNet Developer includes all the features of QualNet Runtime. In addition, it includes the C/C++ source code of the protocol models available in QualNet. Customers can add new protocol models or modify the existing models. Also, additional model libraries are available for QualNet Developer. QualNet Runtime and QualNet Developer can both be used in the same way to create scenarios and run simulations.

The figure 2.5 depicts the architecture of QualNet which is composed of simulation kernel, extensive model libraries such as advanced wireless library for Sensor networks, WiMAX, Universal Mobile Telecommunication Systems (UMTS), satellite communication etc. with a well built GUI.

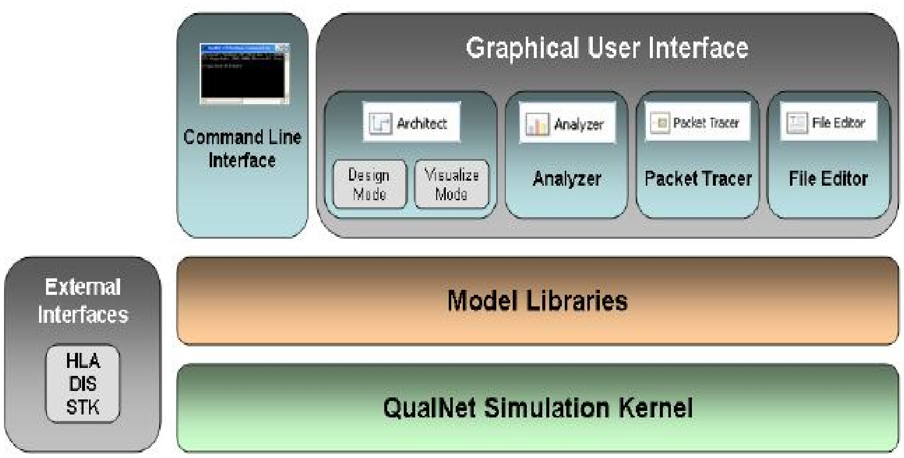

Figure 2.5. Architecture of the QualNet

The tools of the qualnet are as discussed below:

QualNet architect - A graphical scenario design and visualisation tool. In Design mode, one can set up terrain, network connections, subnets, mobility patterns of wireless users and other functional parameters of network nodes. Also customize the protocol stack of any of the nodes, specify the application layer traffic and services that run on the network. In Visualise mode, one can perform in-depth visualisation and analysis of a network scenario designed in Design mode [8].

QualNet analyser - A statistical graphing tool that displays hundreds of metrics collected during simulation of a network scenario. It is used to plot the obtained results by running the simulation.

QualNet packet tracer - A graphical tool that provides a visual representation of packet trace files generated during the simulation of a network scenario. Trace files are text files in XML format that contain information about packets as they move up and down the protocol stack.

QualNet file editor - A text editing tool.

QualNet command line interface - Command line access to the simulator.

QualNet enables users to design new protocol models and to optimise new and existing models. Users are allowed to design large wired and wireless networks using pre-configured or user-designed models.

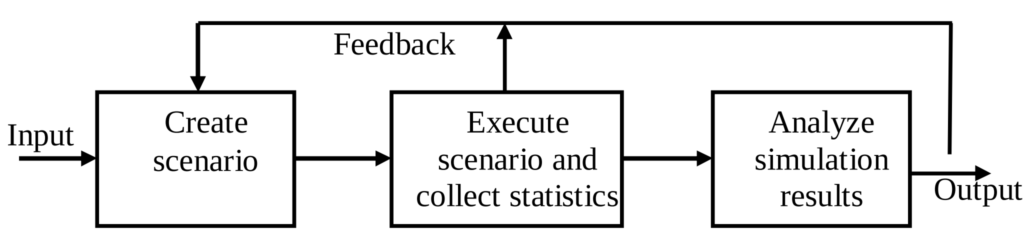

Figure 2.6. General procedure of simulation in QualNet

The performance measures are analysed, to optimise them. The general procedure of simulation in QualNet is shown in figure 2.6. The strong features of the QualNet, which make it one of the popular tools among network simulators for research community, are as discussed below.

Speed: QualNet can support real time speed to enable software-in-the-loop, network emulation and hardware-in-the-loop modelling. Faster speed enables model developers and network designers to run multiple *what-if* analyses by varying model, network and traffic parameters in a short time.

Scalability: QualNet can model thousands of nodes by taking advantage of the latest hardware and parallel computing techniques. QualNet can run on cluster, multi-core and multi-processor systems to model large networks with high fidelity.

Model fidelity : QualNet uses highly detailed standard-based implementation of protocol models. It also includes advanced models for the wireless environment to enable more accurate modelling of real world networks.

Portability: QualNet and its library of models run on a vast array of platforms including Windows XP, Mac OS X, Linux operating systems, distributed and cluster parallel architectures for both 32-bit and 64-bit computing platforms. Users can develop a protocol model or design a network in QualNet on Windows XP computer and then transfer it to a powerful multi-processor Linux server to run capacity, performance and scalability analyses.

Extensibility*:* QualNet can connect to other hardware and software applications, real networks and third party visualisation software to greatly enhancing the value of the network model [9].

2.3.9 EXata

Emulation is a technique used to bridge the gap between simulation experiments and real-world testing. The intention of emulation is to reproduce in real time and in a controlled manner the essential functionality of a system, so that it can interact with other real systems that can thus be evaluated. The network emulation methodology applies the technique of emulation to the field of networks both for network equipment, whose behaviour is reproduced and for the communication conditions between devices, which are modelled and reproduced in a controlled way in the emulated network, thus providing flexibility and repeatability. These emulated components are used in a setup together with the real network equipment and applications under test. Therefore, the experimental results are close to reality and the observations are directly applicable to real situations. Hence, network emulation is a powerful tool for evaluating network equipment, protocols and applications, for research and education purposes, as well as for pre-deployment assessments. To facilitate its understanding and promote its usage, we shall attempt to thoroughly describe in this book the technique of network emulation and compare it with the other experimental approaches: the scholarly analytical modelling, the popular network simulation and the demanding real-world testing.

To emphasise the practical aspects related to emulation, one shall give large number examples of network emulators on the market, as well as provide an in-depth analysis of a case study, the wireless network emulation testbed called Quality Observation and Mobility Experiment Tools (QOMB). One key feature is to discuss not only the emulation of wired networks, which is perhaps an easier task, but also that of wireless networks (WLANs, active RFID tags, IEEE 802.15.4). Given that wireless environments are more exposed to external influences, the technique of network emulation is even more useful in such cases.

Hence, more people will be willing to adopt it once it becomes sufficiently usable, reliable and accurate. An evidence in this sense is the fact that currently all widely used network simulators, such as Ns-2, QualNet Developer, or OPNET Modeller, do offer the possibility of emulation as an optional feature.

Early stage simulations are not always accurate, whereas late stage testbeds are very expensive. There is a real need for better, cheaper testing in the middle and later stages of product development. Apart from aforesaid two types, another type of evaluation platform is network emulation, which is a high fidelity abstraction model. It mimics the functions and behaviour of real system so that it appears like the system to be studied, communicates and behaves like a real system [16].

Typically a network emulator means that end-systems such as computers can be attached to the emulator and will act exactly as they are attached to a real network. The major task of the network emulator is to emulate the network which connects end-hosts, but not the end-hosts themselves. Emulation results are reliable and cost effective technique to reproduce the real system. An emulator is similar to a simulator, except that instead of redefining many levels of the OSI model, an emulator only redefines the particular layer. Hence all the other features of the node participating in emulation remains same, except the separately modelled layers' features. This solution offers an adequate middle ground between simulators and testbeds. Required feature of the experiment can be emulated in hardware, allowing for better performance and accuracy compared to simulators. At the same time, an emulator eliminates many of the practical problems with testbeds by completely controlling external factors that may influence an experiment. Network emulation blends the flexibility and low cost of simulation with the highly accurate, highly detailed results of a testbed. It is valuable throughout the life-cycle of the network-centric system and can be employed in increasing level of detail and realism as the system demands [17].

EXata is a wireless emulator that lets one to evaluate on-the-move communication networks faster and with more realism than any other emulator. EXata creates a digital network replica that interfaces with real networks in real time, using real applications. Software, hardware, human and Internet-in-the-loop connections enable ultra-realistic communication over all layers of the network.

EXata 2.0.1 [9] is a network emulator and simulator provided by SNT. EXata is a superset of QualNet with the hands-on capacity of emulation; it is a comprehensive suite of tools for emulating large wired and wireless networks. It uses simulation and emulation to predict the behaviour and performance of networks to improve their design, operation and management. EXata enables users to handle the network emulation and simulation using the extensive features of its model suite.

EXata enables users to

- Develop emulation or simulation models for new networking technologies.
- Design new network technologies.
- Design new communications protocol models using the OSI-style architecture of EXata protocol stack.
- Design wireless networks of real world size (EXata can be run on multi-core processor computers to evaluate large wireless networks of 100s and 1000s of devices).
- Perform what-if analyses: Analyse the performance of networks and perform "what-if" analyses to optimise them. You can design a network and then run batch experiments to test network performance when parameters, such as routing protocols, timers and transmission power, are varied.

- Connect real networks, applications and devices with EXata emulated network.
- See real applications run on emulated networks: EXata can run real applications, such as VoIP, Internet browsers and streaming video, on emulated networks as if they are running on real networks.
- Train with the network before it is ready to be deployed: EXata makes it possible to conduct training and operations with next-generation tactical networks and devices that are still being designed.
- Analyse and manage EXata virtual networks with popular, industry-standard, tools.
- Snoop on packets: EXata has a packet sniffer interface that enables third-party tools like Wire shark and Microsoft Network Monitor to snoop / capture traffic from any device in EXata and analyse it. This lets you debug and troubleshoot network problems.
- Manage an emulated network: EXata comes with a SNMP Agent, which enables one to use standard SNMP managers to view, monitor and control emulated networks in EXata just like managing real networks.

Figure 2.7 depicts the general architecture in EXata. Figure 2.8 shows the general procedure of simulation and emulation of EXata tool.

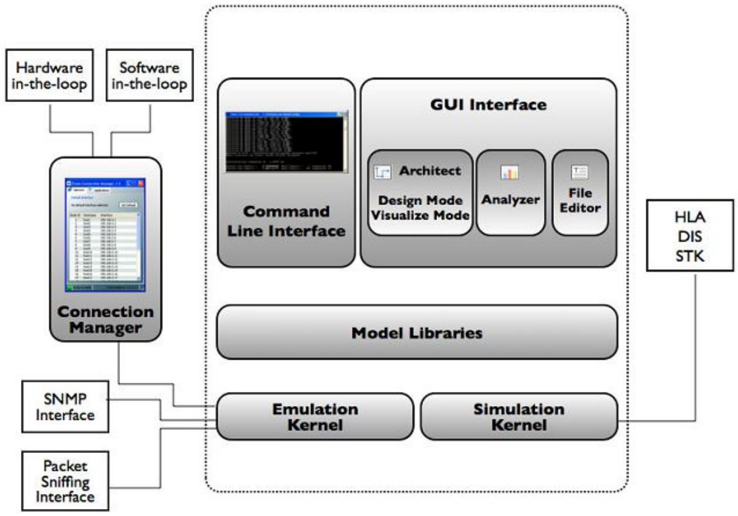

Fig 2.7. Architecture of EXata

EXata is very similar in its architecture to QualNet, but some extra blocks which make EXata different from QualNet are discussed below.

The kernel consists of both simulation and emulation kernels.

EXata connection manager: EXata connection manager is the companion module of the main EXata emulation engine. The EXata emulation engine creates a digital replica of the target network and EXata connection manager is used to run applications on the emulated network. The connection manager makes EXata advanced emulation technology easy and simple to use. Applications need no modification or customization to use the realistic emulated network in EXata. Connection manager supports a large variety of applications such as: internet browsers, tactical communications, situational awareness information, sensor data, instant messaging

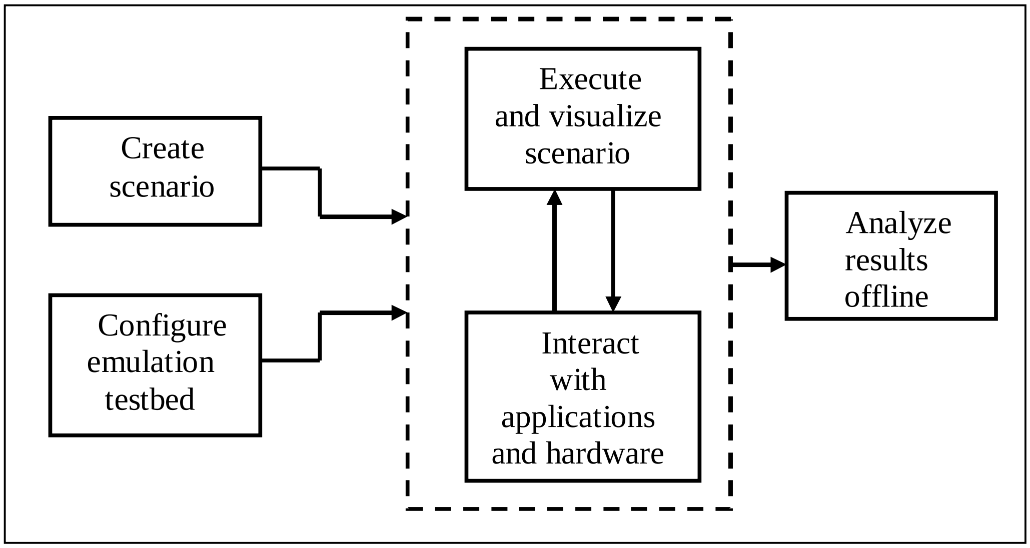

Fig 2.8. General procedure of simulation and emulation in EXata

EXata is very similar in its architecture to QualNet, but some extra blocks which make EXata different from QualNet are discussed below.

The kernel consists of both simulation and emulation kernels.

EXata connection manager: EXata connection manager is the companion module of the main EXata emulation engine. The EXata emulation engine creates a digital replica of the target network and EXata connection manager is used to run applications on the emulated network. The connection manager makes EXata advanced emulation technology easy and simple to use. Applications need no modification or customization to use the realistic emulated network in EXata. Connection manager supports a large variety of applications such as: internet browsers, tactical communications, situational awareness information, sensor data, instant messaging, VoIP, streaming video and multiplayer games.

EXata external interface: EXata supports a packet sniffer interface to enable capture and analysis of network traffic using standard packet sniffer/analyser tools like Wire shark or Microsoft network monitor. Additionally, EXata can be managed using standard SNMP network managers like HP Open view, IBM Tivoli or Solar Winds Orion.

An example of EXata testbed is shown in figure 2.10, consisting of an emulation server networked with two real computers through router switch. One of the hosts is mapped as video server and the other host as video client.

2.4 Simulation using QualNet

QualNet 6.1 consists of well defined comprehensive Wireless sensor networks library, advanced wireless library module to support WiMAX and Long Term Evolution (LTE) model libraries. Figure 2.9 is an example scenario screenshot captured from QualNet tool GUI. This tool facilitates to study the performance of the different network parameters such as throughput, average end-to-end delay, average jitter.

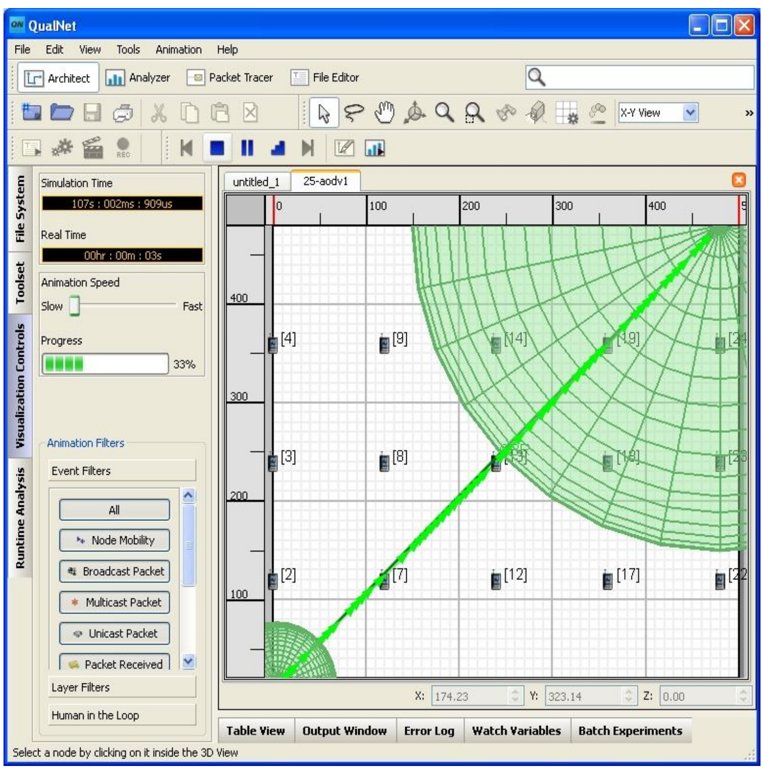

Fig 2.9. Snap shot of QualNet simulator for 20 node scenario

The performance comparison of routing protocols AODV, LAR and ZRP are studied using Qualnet 5.0.2 simulator considering IEEE 802.15.4 standard. In this simulation study, the CBR connections are varied from 2 to 6 connections in steps of 2 for different node densities of 20 to 100 nodes in steps of 20 nodes. The simulation parameters configured for the performance evaluation are shown in the table 1. The performance of the routing protocols are studied and compared by considering throughput, total bytes received, end-to-end delay and average jitter as performance metrics.

Table 2.1. Simulation parameters

Parameter	**Value**
Radio type	802.15.4
Simulation time	300 sec
Routing Protocols	AODV, LAR & ZRP
No. of Channels	One
Channel frequency	2.4 GHz
Path loss model	Two Ray
Energy model	Mica Motes
Shadowing model	Constant
Battery model	Linear model
Number of nodes	20, 40, 60 80 and 100
Traffic types	2 , 4 and 6 CBR connections
Mobility of nodes	None
Node Placement	Grid
Packet size	50 bytes

The figure 2.10, 2.11 and 2.12 show the throughput, average end-to-end delay and average jitter performances respectively.

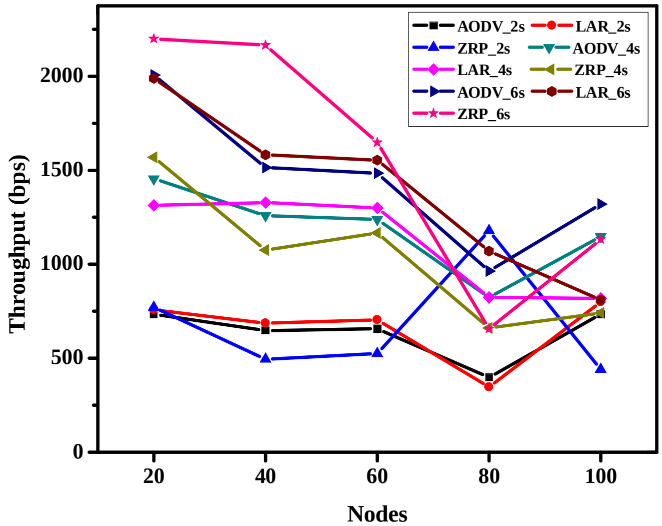

Fig 2.10. Throughput at different node density

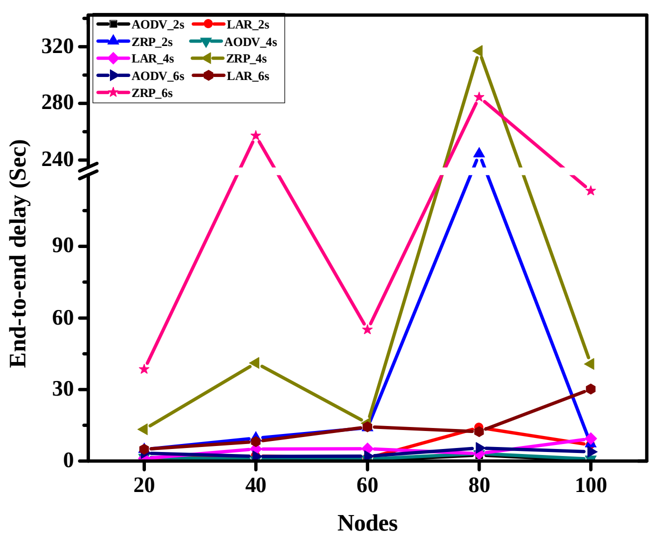

Fig 2.11. Average end-to-end delay at different node density

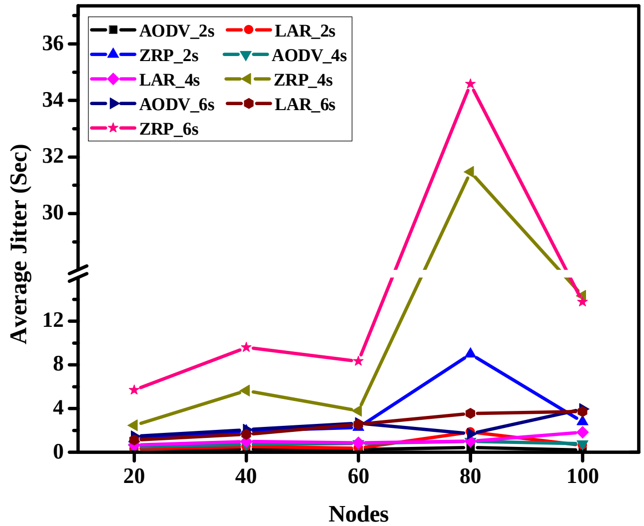

Fig 2.12. Average jitter at different node density

2.5 Simulation/Emulation using EXata

EXata 2.0.1 consists of well defined Wireless Sensor network module to support the WSN simulation/emulation. Figure 2.13 is an example scenario screenshot captured from EXata tool.

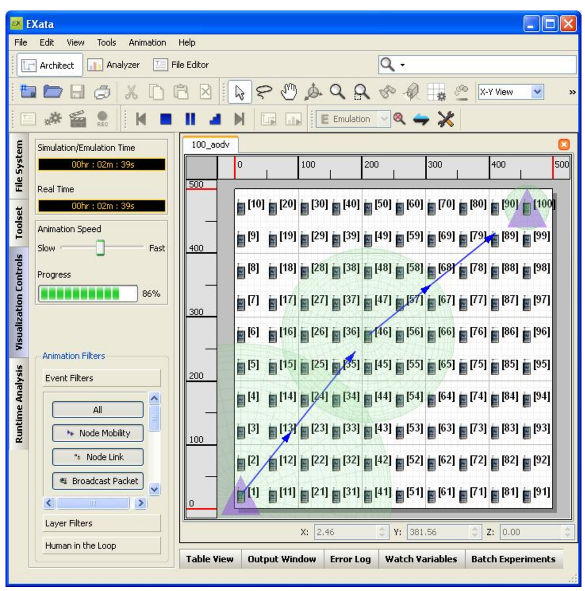

Fig 2.13. AODV routing for video streaming in EXata

2.6 REFERENCES:

[1] Raj Jain, “Art of Computer Systems Performance Analysis Techniques for Experimental Design Measurements Simulation and Modelling”, Wiley Computer Publishing, John Wiley & Sons, Inc. ISBN: 0471503363,1991.

[2] Jean-Yves and Le Boudec, “Performance Evaluation of Computer and Communication Systems”, EPFL Press, Lausanne, Switzerland, ISBN: 978-1-4398-4992-7, 2010.

[3] Aleksi Penttinen, “Algorithms and Performance Evaluation Methods for Wireless Networks”, Doctoral thesis, Helsinki University of Technology Networking Laboratory, 2006.

[4] Johannes Lessmann, Peter Janacik, Lazar Lachev and Dalimir Orfanus, “Comparative Study of Wireless Network Simulators”, Seventh International on Networking ICN, IEEE Computer Society,978-0-7695-3106-9/2008.

[5] Razvan Beuran, “Introduction to Network Emulation”, PanStanford Publishing Pte. Ltd, ISBN: 978-981-4364-09-6, 2013.

[6] http://www.webnms.com/simulator/network-simulator-ds.html

[7] http://www.cs.cornell.edu/skeshav/real/overview.html

[8] Thontadharya H.J., Shwetha D., Subramanya Bhat M. and Devaraju J.T., “Simulation and Emulation Approach for the Performance Evaluation of Adaptive Modulation and Coding Scheme in Mobile WiMAX Network”, International Journal of Computer Applications (0975 – 8887) Vol. 44– No16, April 2012.

[9] Documentation of QualNet User’s Guide, http://www.scalablenetworks.com\qualnet\documentation.

[10] Documentation of EXata 2.0.1 User’s Guide, http://www.scalablenetworks.com \EXata.

[11] http://www.isi.edu/nsnam/ns

[12] http://en.wikipedia.org/wiki/Ns-3

[13] T. J.-S. developers. J-sim.http://www.j-sim.org.

[14] http://www.omnetpp.org/ documentation

[15] http://www.opnet.com/documentation

[16] Sam Burnett, “Combining Wireless Network Simulation and Emulation”, Undergraduate thesis, Carnegie Mellon University, May 2008.

[17] White Paper, “Test & Evaluation of Network-Centric Systems”, www.QualNet.com

CHAPTER 3

MULTIPATH ADHOC ON DEMAND DISTANCE VECTOR ROUTING PROTOCOL (MP-AODV)

3.1 Introduction

In this chapter, a routing protocol Multipath Adhoc On demand Vector Routing protocol (MP-AODV) is developed by considering AODV as a base protocol. The main objective of the MP-AODV protocol is to increase the throughput of the data between the source and destination node and to reduce the end-to-end delay and average jitter compared to AODV protocol. Hence, additional modifications are made at different control messages in the base protocol to design the proposed routing protocol. The rest of this chapter is organized as follows. Section 3.2 describes the AODV protocol. Section 3.3 outlines the related work in the literature. The Multipath Adhoc On demand Vector Routing protocol (MP-AODV) is presented in Section 3.4. Section 3.5 discusses the simulation results followed by conclusion in Section 3.6.

3.2 Adhoc On demand Distance Vector protocol (AODV)

Ad-hoc On Demand Distance Vector (AODV) protocol is suitable for Unicast and Multicast routing. It is a reactive routing protocol [1] and is basically a combination of DSDV and DSR. It incorporates the basic on-demand mechanism of route discovery and route maintenance from DSR, the use of hop-by-hop routing, sequence numbers and periodic beacons from DSDV. This protocol performs route discovery using control messages such as route request (RREQ) and route reply (RREP) whenever a node wishes to send packets to destination. The forward path sets up an intermediate node in its route table with a lifetime association RREP. The neighborhood information is obtained from broadcast Hello packet. When the source node receives the route error (RERR) message, it can reinitiate route if it is still needed. AODV is a flat routing protocol which does not need any central administrative system to handle the routing process. AODV tends to reduce the control traffic messages overhead at the cost of increased latency in finding new routes. The RREQ and RREP messages which are responsible for the route discovery do not increase significantly the overhead from these control messages. AODV reacts relatively quickly to the topological changes in the network. It updates the hosts that may be affected by the change, using RERR message. The Hello messages are responsible for the route maintenance and are limited so that they do not create unnecessary overhead in the network. The AODV protocol is a loop free and uses sequence numbers to avoid the infinity counting problem which are typical to the classical distance vector routing protocols.

3.3 Related work

A mobile ad hoc network is a mobile, multi-hop wireless network that does not rely on any preexisting infrastructure. Mobile ad hoc networks are characterized by dynamic topologies due to uncontrolled node mobility, limited and variable shared wireless channel bandwidth, and wireless devices constrained by battery power. One of the key challenges in such networks is to design dynamic routing protocols that are efficient, that is, consume less overhead [1]. The authors in [1] have proposed an on-demand multipath protocol called AOMDV that extends the single path AODV protocol to compute multiple paths. AOMDV ensures that theset of multiple paths are loop-free and the alternate paths at every node are disjoint.

The authors in [2], evaluated the performance of AODV, AOMDV and DSR using ns-2. Comparison was based on the packet delivery fraction, throughput and end-to-end delay. They concluded that in the static network (pause time 50 sec), AOMDV gives better performance as compared to AODV and DSR in terms of packet delivery fraction and throughput but worst in terms of end-to end delay. We have also seen that DSR routing protocol is best in terms of end-to-delay in both Static and dynamic network for each set of maximum connections.

In paper [3], the performance characteristics of AODV and AOMDV were analyzed at varying workload and the energy consumption at nodal level was closely monitored. From the NS-2 simulation results, the following conclusions were drawn by the authors. The performance AOMDV is better than AODV at all workloads with respect to packet loss. At medium and low workloads, the throughput was higher for AOMDV. But at high workload, the throughput was lower. An increasing workload is detrimental to the performance of AOMDV. The end-to-end delay is another important issue to be addressed when using AOMDV. The use of longer multi hop paths could lead to higher end-to-end delay in most instances. An indicative energy consumption pattern of a particular node shows notable variation in energy level. At higher workload, the energy level of the observed node is minimum compared to medium and lower workload. This is suggesting that at higher data traffic, load will get balanced in more paths resulting in energy preservation of nodes in the network. This could lead to a positive effect by increasing the network life.

In this paper [4] authors discussed some important issues related to the load-balanced routing protocols for mobile ad hoc networks Load balanced routing protocols have different load metric as route selection criteria to better use MANET recourses and improves MANET performance. Many areas of research in this field which deserve further investigation include robustness, security, energy efficiency, low overhead, reliability and scalability. Effective and efficient solutions to these issues require the design and development of new multipath routing protocols in MANETs.

The authors in [5], proposed AOMDV with queue length estimation technique showing reduced congestion by choosing non congested routes to send RREQ and data packets and to transfer the load to higher hop count alternate paths if the nodes or route turn out to be congested. In other words, among paths that exist in the network, those paths that have longer lifetime and nodes along them have smaller queue lengths are selected for routing data packets along them. The authors presented a AOMDV routing protocol for identification of possibly multiple node-disjoint paths between a given source and a destination such that the paths identified satisfy performance constraints. The performance of the network like packet delivery ratio, throughput and minimize the end-to-end delay of the network were analyzed.

In future, the authors planned to apply proposed scheme with any location aware protocol like DREAM or LAR and analyze the effect of location aware protocol on energy consumption and also apply the energy efficient routing scheme in WIMAX technology to find the proper energy consumption on it.

The authors in [6], evaluated the performances of AODV and AOMDV using ns-2. Comparison was based on of packet delivery fraction, routing overhead incurred, average end-to-end delay and number of packets dropped, they conclude that AOMDV is better than AODV. AOMDV outperforms AODV due its ability to search for alternate routes when a current link breaks down. Though AOMDV incurs more routing overheads while flooding the network and packet delays due its alternate route discovery mechanism, it is much more efficient when it comes to packet delivery for the same reason. Hence, in conclusion they examined that when network load tolerance is of no consequence, AOMDV is a better on-demand routing protocol than AODV since it provides better statistics for packet delivery and number of packets dropped. But if routing overhead is a concern, then AODV is preferred over AOMDV.

3.4 Multipath Adhoc On demand Distance Vector protocol (MP-AODV)

In this work, MP-AODV protocol is developed by extending AODV protocol by adding extra fields in route request packet such as next hop and last hop information. The protocol is designed to discover multiple paths between the source and the destination in every route discovery.

In MP-AODV, RREQ propagation from the source towards the destination establishes multiple reverse paths both at intermediate nodes as well as the destination. The core of the routing protocol lies in ensuring that multiple paths discovered are loop-free and disjoint, and efficiently finding such paths using a flood-based route discovery.

MP-AODV relies as much as possible on the routing information already available in the underlying AODV protocol, thereby limiting the overhead incurred in discovering multiple paths. In particular, it does not employ any special control packets. In fact, extra RREPs and RERRs for multipath discovery and maintenance along with a few extra fields in routing control packets (i.e., RREQs, RREPs, and RERRs) constitute the only additional overhead in MP-AODV relative to AODV.

Besides maintaining multiple loop-free paths, the protocol seeks to find disjoint alternate paths. We consider two types of disjoint paths: link disjoint and node disjoint. Link disjoint set of paths between a pair of nodes have no common links, whereas node-disjointness additionally eliminates common intermediate nodes.

In this work, a node disjoint path is established to ensure a common node does not exist between two alternate paths established between source node and destination node. The modifications are made in four components of AODV to extend it to a MP-AODV routing protocol

- Routing table structure
- Route discovery
- Route maintenance
- Data packet forwarding

3.4.1 Routing table structure

In MP-AODV, a route list is used to store additional information for each alternate path including: next hop, last hop, hop count, and expiration timeout. The last hop information is useful in checking the disjointness of alternate paths. The difference in the route table structure between MP-AODV and AODV routing protocol are shown in figure 3.1 and figure 3.2 respectively.

Destination	Sequence number	Route list			
		nexthop1	lasthop1	hopcount1	Timeout1
		Nexthop2	lasthop2	hopcount2	Timeout2…….

Fig 3.0(a) MP-AODV

Destination	Sequence number	Hop count	Next hop	timeout

Fig 3.0(b) AODV

3.4.2 Route discovery

As in AODV, when a traffic source needs a route to a destination, the source initiates a route discovery. In AODV, only the first copy of the RREQ is used to form reverse paths; the duplicate copies that arrive later are simply discarded.

In MP-AODV, these duplicate copies can be gainfully used to form alternate reverse paths. Thus, all duplicate copies are examined for potential alternate reverse paths, but reverse paths are formed only using those copies that preserve loop-freedom and disjointness among the resulting set of paths to the source.

3.4.3 Route maintenance

Route maintenance in MP-AODV is a simple extension to AODV route maintenance. Like AODV, AOMDV also uses RERR packets. A node generates or forwards a RERR for a destination when the last path to the destination breaks.

MP-AODV also includes an optimization to salvage packets forwarded over failed links by re-forwarding them over alternate paths.

3.4.4 Data packet forwarding

For data packet forwarding at a node having multiple paths to a destination, we adopt a simple approach of using a path until it fails and then switch to an alternate path. We use paths in the order of their creation. In particular, we need a more restricted notion of disjointness than node or link disjointness that additionally accounts for interference among alternate paths.

3.5 Simulation results

The performance of MP-AODV and AODV protocols are evaluated using Qualnet 6.1 simulator considering the wireless radio 802.15.4 and 802.11. The procedure and the result of the simulation are as discussed below.

3.5.1 Scenario 1

Initially, the simulation is carried out for MP-AODV routing protocol by placing 20 stationary nodes in grid format with a CBR connection of 4.096Kbps data rate established between source and destination. During simulation, at a regular interval of 50 seconds the active routes are broken by deactivating the one of the node in the path, which is used to forward the data packets from source to destination to study performance of multi reserve path characteristics of MP-AODV as compared to single path characteristics of AODV. The performance metrics such as throughput, average end-to-end delay and average jitter are evaluated. The simulation studies are repeated by increasing the node density from 20 to 80 in steps of 20 nodes. Similarly, the simulation studies are repeated by considering AODV routing protocol. The performance metric considered are throughput, average jitter and end-t-end delay. The simulation parameters configured for the performance evaluation are shown in the table.3.1.

Table 3.1. Simulation parameters

Parameters	Values
Radio type	802.15.4 Radio
Simulation time	300s
Link Break time	50s
Simulation area	1500m X 1500m
Data Rate	4.096kb/s
Routing protocols	AODV, MP-AODV
Channel frequency	2.4 GHz
Fading Model	None
Path Loss model	Two Ray
Shadowing mean	4.0dB
Energy model	Mica motes
Battery model	Linear model
Traffic type	Constant Bit Rate
Packet size	512 bits
Node placement	Grid

Figure 3.1 illustrates the variation of throughput performance of AODV and MP-AODV routing protocols for various node densities with the CBR data rates of 4.096 kbps. It is observed that, the throughput corresponding to MP-AODV is higher compared to AODV. This is because MP-AODV discovers multiple reserve routes from source to destination in each route request and stored it in routing table. After each break of active route, MP-AODV retrieves route information to switch to the previously discovered alternate routes from the routing table to send the data packets to destination instead of initializing the re-route discovery process. As a result, the amount of flooding packets involved in route discovery process is been considerably reduced after each break of active route. This process will continue until all routes in the routing table get exhausted and route discovery process is re-initiated to discover the fresh routes, whereas in AODV route discovery is re-initiated. Thus, unnecessary network congestion due to broadcasting of RREQ is been reduced which has led to improvement in throughput.

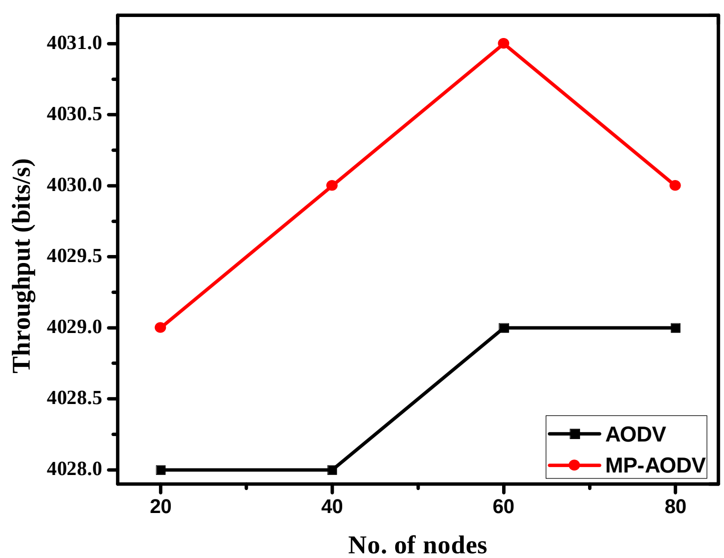

Figure 3.1: Throughput for various node densities

Figure 3.2 illustrates the variation of average jitter for various node densities by using AODV and MP-AODV routing protocols. It is observed that in each route discovery MP-AODV discovers and stores multiple routes from source to destination. After each break of active route in MP-AODV, the time overhead involved for switching to the alternative path is considerably less as compared with route re-discovery overhead. Thus, jitter between the data packets in AOMDV is decremented by 50.3 % as compared with AODV data packets.

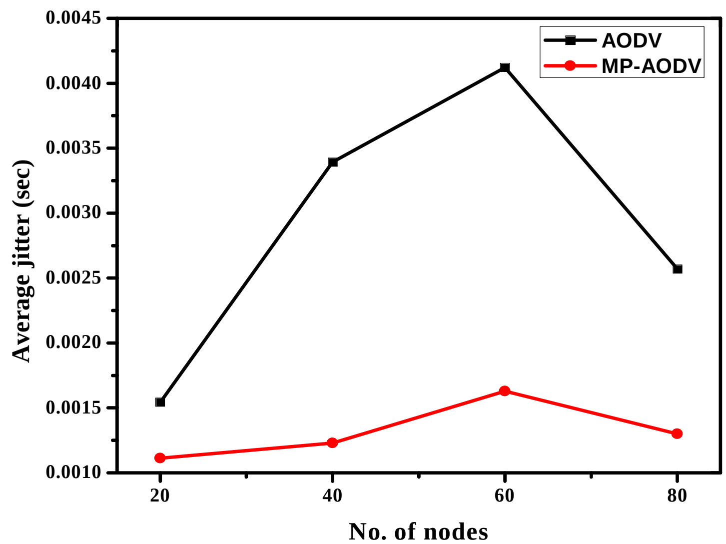

Figure 3.2:Average Jitter for various node densities

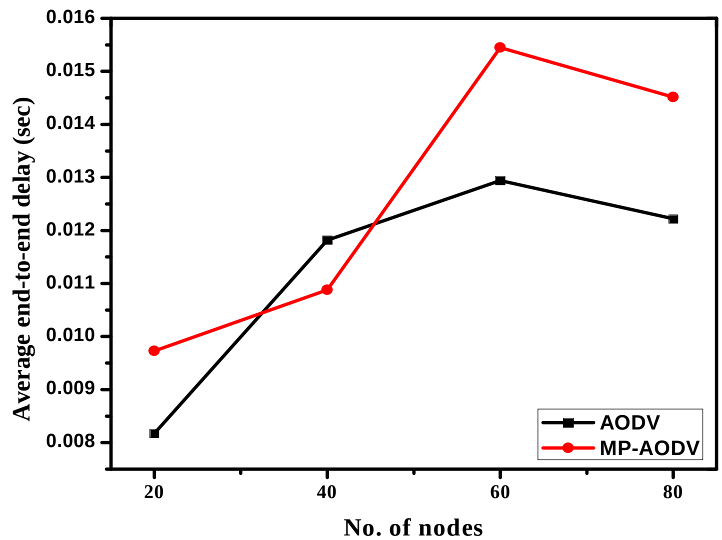

Figure 3.3: Average End -to -end delay for various node densities

Figure 3.3 illustrates the variation of end-to-end delay for various node densities at 4.096kbps data rates using AODV and MP-AODV. In MP-AODV, the route discovery process inherently contributes slight increase in end-to-end delay as compared with AODV because the number of processing steps involved in finding disjoint routes is more during the broadcast of RREQ . In addition to this, some of the alternative paths discovered in MP-AODV may not be optimal, since they contain higher hop count metric. Routing the data packets through non-optimal path may lead to increase in end-to-end delay, which can be reduces to some extent by selecting the optimal path consisting of lower hop metric.

3.5.2 Scenario 2

Initially, the simulation is carried out for MP-AODV routing protocol by placing 25 stationary nodes in grid format with a CBR connection of 1second CBR rate established between source and destination. During simulation, at a regular interval of 30 seconds the active routes are broken by deactivating the last hop node in the path, which is used to forward the data packets from source to destination. To study performance of multi reserve path characteristics of MP-AODV as compared to single path characteristics of AODV, the routes were broken till the end of simulation time 300s at a regular interval of 30s. The performance metrics such as throughput, average end-to-end delay and average jitter, total bytes received, total energy consumed, total packet drops, number of RREQ initiated, total number of active routes between source and destination and number of links broken during entire simulation time are evaluated. The simulation studies are repeated by increasing the node density from 25 to 225 nodes in steps of n^2 nodes, where n=5, 7, 9, 11, 13 and 15. Similarly, the simulation studies are repeated by considering AODV routing protocol. The simulation parameters configured for the performance evaluation are shown in the table.3.2.

Table3.2. Simulation parameters

Parameters	Values
Radio type	802.11b
Simulation time	300s
Link Break time	30s
Simulation area	500m X 500m
CBR Rate	1sec and 0.1 sec
Routing protocols	AODV, MP-AODV
Channel frequency	2.4 GHz
Fading Model	None
Path Loss model	Two Ray
Number of nodes	25,49,81,121,169,225
Energy model	Mica motes
Battery model	Linear model
Traffic type	Constant Bit Rate
Packet size	512 bytes
Node placement	Grid

Figure 3.4 and 3.13 illustrates the variation of throughput performance of AODV and MP-AODV routing protocols for various node densities with the CBR

data rates of 1s and 0.1 s respectively. It is observed that, the throughput corresponding to MP-AODV is better compared to AODV. This is because MP-AODV discovers multiple reserve routes from source to destination in each route request and stored it in seen table. After each break of active route, MP-AODV retrieves route information to switch to the previously discovered alternate routes from the routing table to send the data packets to destination instead of initializing the re-route discovery process. As a result, the amount of flooding packets involved in route discovery process is been considerably reduced after each break of active route. This process will continue until all routes in the routing table get exhausted and route discovery process is re-initiated to discover the fresh routes, whereas in AODV route discovery is re-initiated at every route break. Thus, unnecessary network congestion due to broadcasting of RREQ is been reduced resulting in improvement of throughput by approximately 6% in 1s CBR rate. It is also observed in figure 10 that due to increase in data rate by 10 times, there exist considerable improvement in the throughput with respect to MP-AODV compared to AODV. Similar result is observed with respect to total bytes received i.e., in figure 3.7 and 3.16.

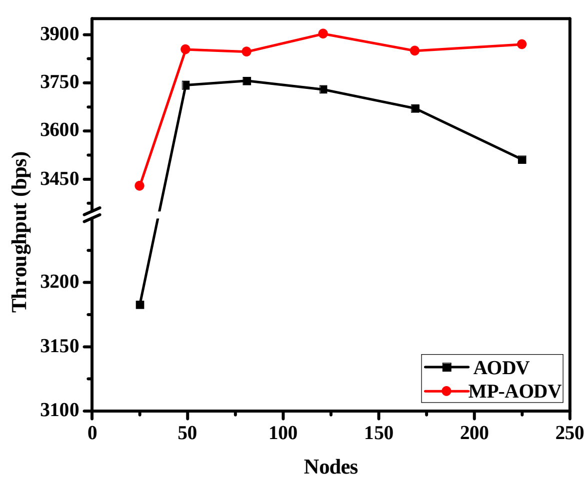

Figure 3.4: Throughput for various node densities

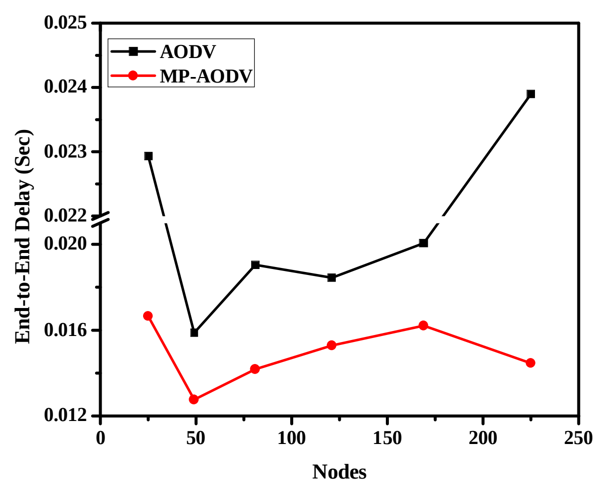

Figure 3.5: End-to-end delay for various node densities

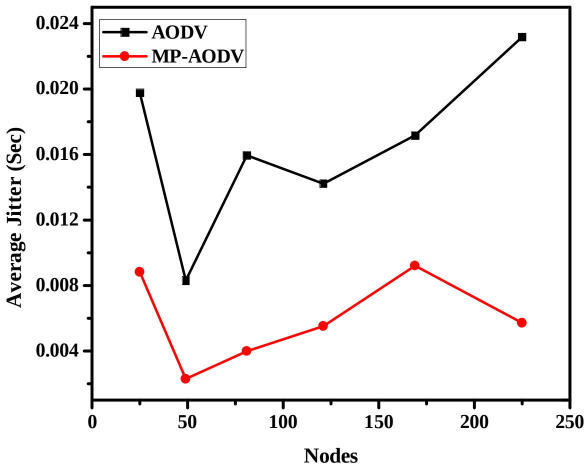

Figure 3.6: Average jitter for various node densities

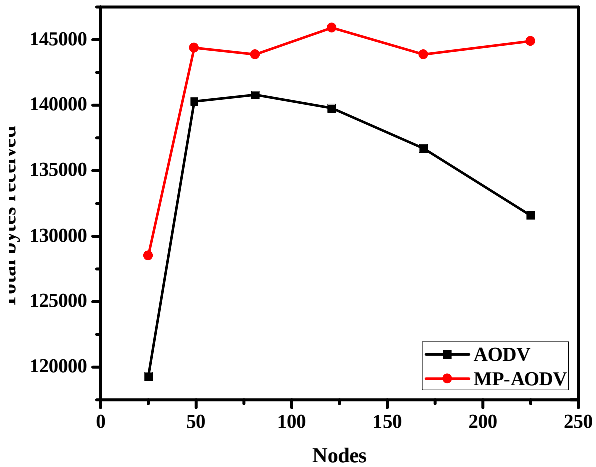

Figure 3.7: Total bytes received for various node densities

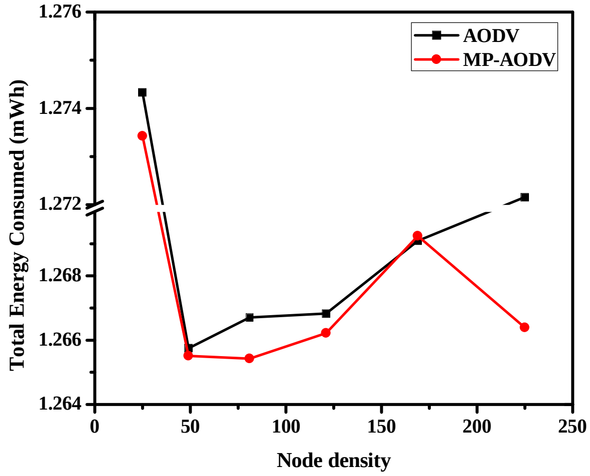

Figure 3.8: Total Evergy consumed for various node densities

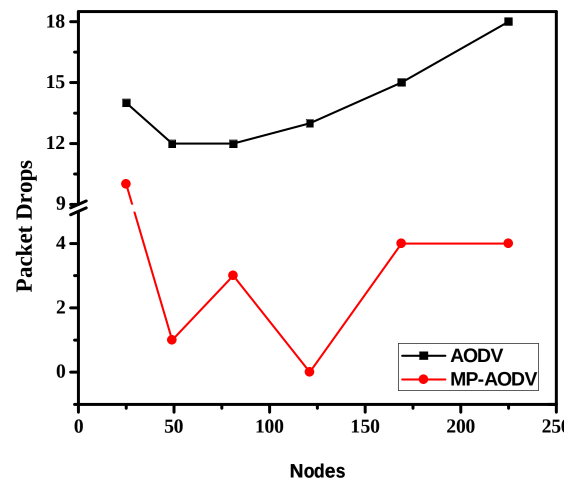

Figure 3.9: Total packet drop for various node densities

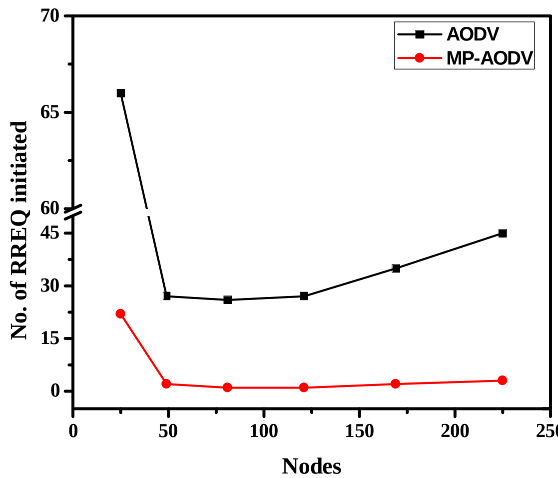

Figure 3.10: Total RREQ initiated for various node densities

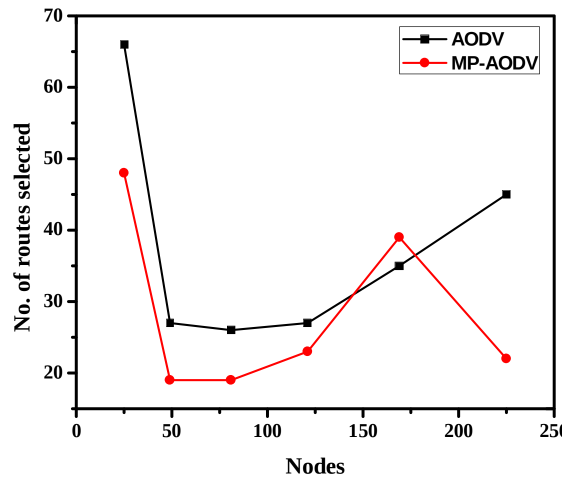

Figure 3.11: Total no. of routes for various node densities

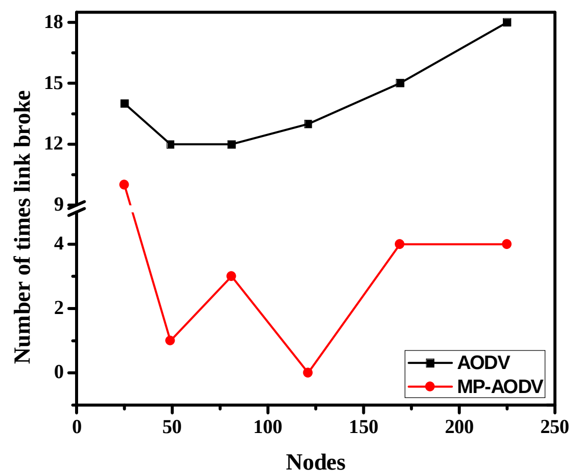

Figure 3.12: Total no. of links broken for various node densities

Figure 3.5 and figure 3.14 illustrate the variation of End-to-end delay for various node densities by using AODV and MP-AODV routing protocols. It is observed that in each route discovery MP-AODV discovers and stores multiple routes from source to destination. It is observed that the delay with respect to MP-AODV on an average reduced by 24.7% compared to the delay with respect to AODV routing protocol at 1s CBR rate where as there exist a considerable decrease in delay at 0.1s CBR rate. At higher data rate, the number of processing steps involved in finding disjoint routes is much more during the broadcast of RREQ. In addition to this, some of the alternative paths discovered in MP-AODV may not be optimal, since they contain higher hop count metric. Routing the data packets through non-optimal path may lead to increase in end-to-end delay, which can be reduces to some extent by selecting the optimal path consisting of lower hop metric.

Figure 3.6 and figure 3.15 illustrate the variation of Average jitter for various node densities by using AODV and MP-AODV routing protocols. After each break of active route in MP-AODV, the time overhead involved for switching to the alternative path is considerably less as compared with route re-discovery overhead.

Thus, jitter between the data packets in MP-AODV is decremented by 64.28 % as compared with AODV data packets at 1s CBR rate and decremented by 50% as compared with AODV data packets at 0.1 s CBR rate.

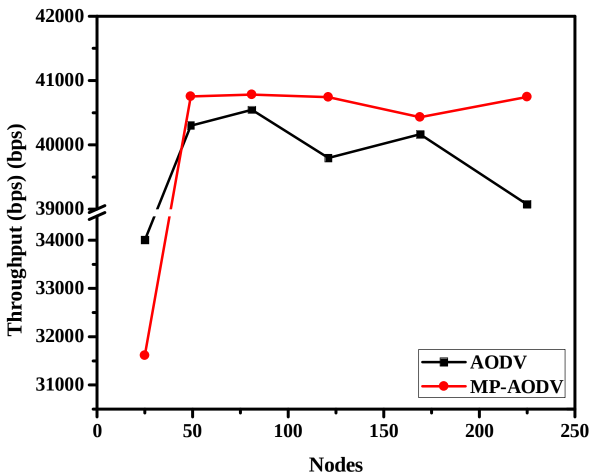

Figure 3.13: Throughput for various node densities

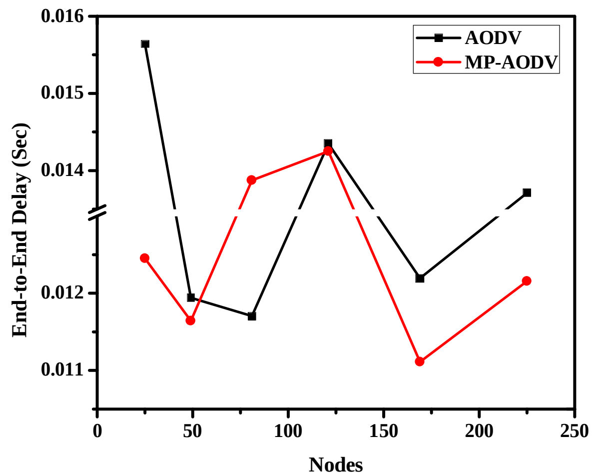

Figure 3.14: End-to-end delay for various node densities

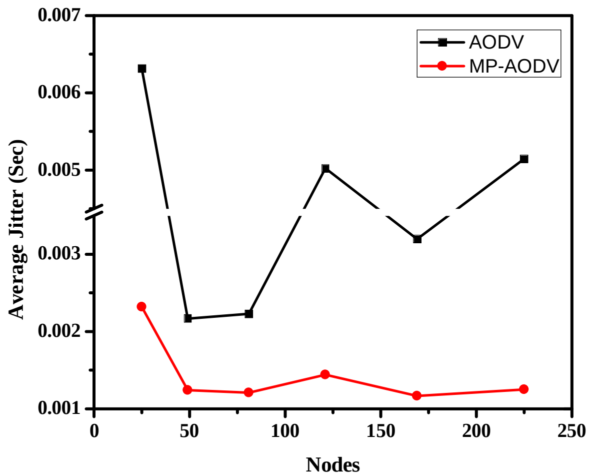

Figure 3.15: Average jitter for various node densities

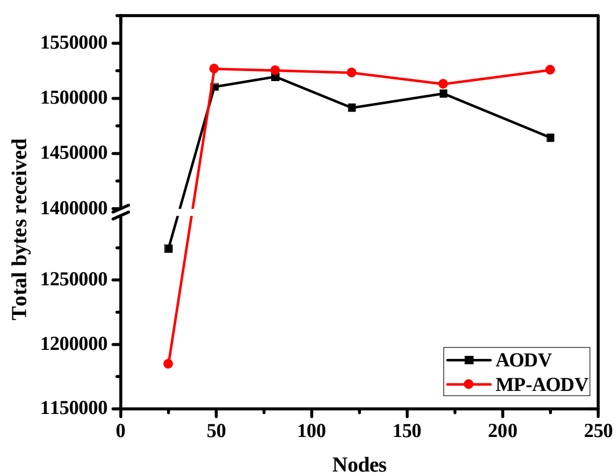

Figure 3.16: Total bytes received for various node densities

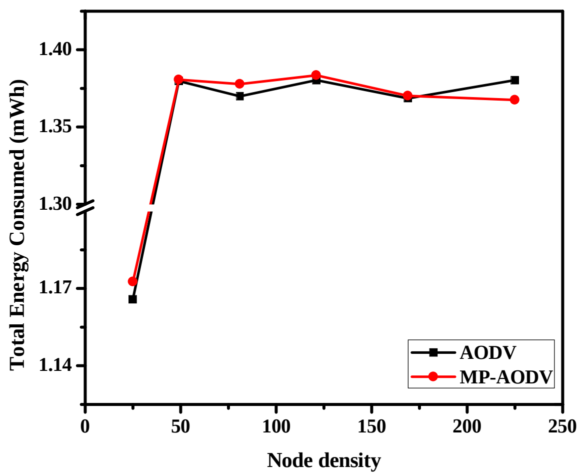

Figure 3.17: Total Evergy consumed for various node densities

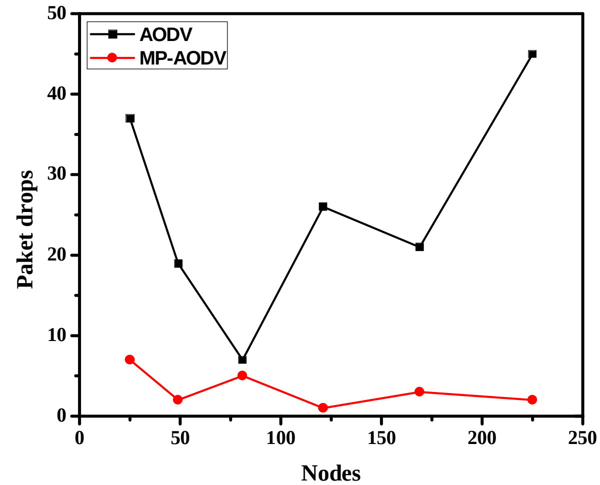

Figure 3.18: Total packet drop for various node densities

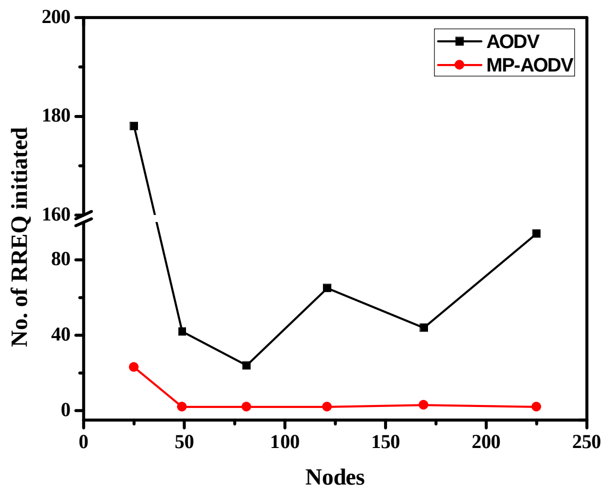

Figure 3.19: Total RREQ initiated for various node densities

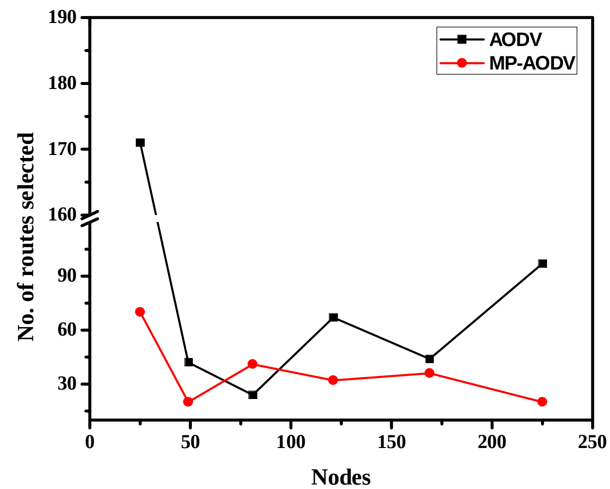

Figure 3.20: Total no. of routes for various node densities

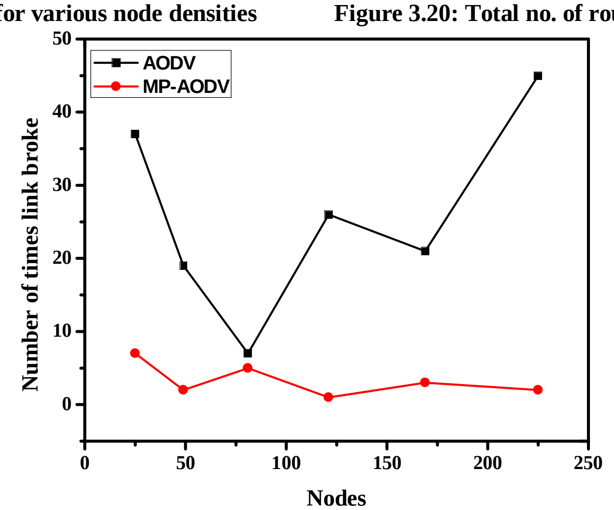

Figure 3.21: Total no. of links broken for various node densities

Figure 3.8 and figure 3.17 illustrate the variation of total energy consumed for various node densities by using AODV and MP-AODV routing protocols. It is observed that the total energy consumed in MP-AODV is less compared to AODV at 1s CBR rate. It is also observed that when the data rate increased by 10 times, the energy consumed in a network is found almost same with respect to AODV and MP-AODV.

Figure 3.9 and figure 3.18 illustrate the variation of total packets drop for various node densities by using AODV and MP-AODV routing protocols. It is observed that packet drop due to retransmission is less in MP-AODV when compared with AODV routing protocol.

Figure 3.10 and figure 3.19 illustrate the variation of total RREQ packets initiated for various node densities by using AODV and MP-AODV routing protocols. It is observed that number RREQ initiated for effective route discovery between source and destination is less in MP-AODV when compared with AODV.

Figure 3.11 and figure 3.20 illustrate the variation of total number of route selected for various node densities by using AODV and MP-AODV routing protocols. It is observed that number of active route selected in MP-AODV is less and sufficient with respect to simulation time for efficient data delivery. The active routes in case of AODV are more due to route rediscovery and link breakage in the selected route.

Figure 3.12 and figure 3.21 illustrate the variation of total number of link broken for various node densities by using AODV and MP-AODV routing protocols. It is observed that the number of link breaks with respect to MP-AODV is less when compared with AODV.

Also, the implementation of the protocol is analyzed on the testbed and observed that the protocol MP-AODV performs better compared to AODV routing protocol.

3.6 Conclusion

In this work, MP-AODV protocol is developed by extending AODV protocol by adding extra fields in route request packet such as next hop and last hop information and in the seen- table. The duplicate RREQ are not discarded instead stored and analyzed whether the RREQ has reached a unique next hop and last hop node to have a node disjoint path. From the simulation and emulation results, it is evident that MP-AODV routing protocol performs better than AODV because of multi reserve path characteristics which has led to improvement in lower average jitter, end-to-end delay and throughput.

3.7 REFERENCES:

[1] Mahesh K. Marina, Samir R. Das," Ad hoc on-demand multipath distance vector routing", Wirel. Commun. Mob. Comput. 2006; 6:969–988.

[2] Manveen Singh Chadha, Rambir Joon, Sandeep, " Simulation and Comparison of AODV, DSR and AOMDV Routing Protocols in MANETs", International Journal of Soft Computing and Engineering (IJSCE), ISSN: 2231-2307, Volume-2, Issue-3, July 2012

[3] S. Malini E. Kannan A. Valarmathi C. Daniel Sundar Rajan," Traffic based Energy Consumption Analysis of AOMDV Protocol in a Mobile Ad Hoc Network", International Journal of Computer Applications, Volume 14– No.6, February 2011

[4] Rakhi Sharma, " Survey of Load Balancing and Load Sharing through AOMDV Protocol in MANET", International Journal of Scientific & Engineering Research, Volume 4, Issue 5, May-2013

[5] Archana Shukla, Sanjay Sharma, " Queue Length based Load Balancing Technique using with AOMDV Protocol in MANET", International Journal of Scientific & Engineering Research, Volume 4, Issue 10, October-2013

[6] S.R.Biradar et al, "Performance Evaluation and Comparison of AODV and AOMDV", International Journal on Computer Science and Engineering Vol. 02, No. 02, 2010

[7] Qualnet development library, http://www.scalable-networks.com :http://www.qualnet.com.