RELIABILITY AND PERFORMANCE EVALUATION OF COMPUTING SYSTEMS AND MANET USING RELIABLE DSR PROTOCOL

[1]ANIL CHOUDHARY, [2]O.P. ROY, T. TUITHUNG, [3]DEEPIKA MAHARAJ

North Eastern Regional Institute of Science and Technology (NERIST), Itanagar, India
NERIST, Itanagar, NERIST, Itanagar, Global Engineering College, Jabalpur, India
Email: anilsa2009@yahoo.in, oproy62@yahoo.com, t_tuithung@yahoo.com, maharajdeepika@ymail.com

Abstract: This paper discusses improved reliable DSR protocol which can be used to improve the reliability and performance of mobile ad-hoc networks. Reliability is the important concern while designing wireless ad-hoc network protocols. Therefore, evaluating and improving reliability has become essential in MANET systems. In this paper a Reliable Node Model is designed to improve the performance of MANET and simulated using NetSim.

Index Terms—Bathtub Curve; Reliability Models; DSR

I. INTRODUCTION

The world is becoming more dependent on wireless and mobile services, but the ability of wireless network infrastructures to handle the growing demand is uncertain. Failures not only affect current voice and data use but could also limit emerging wireless applications such as e-commerce and high-bandwidth internet access. As wireless and mobile systems play greater roles in emergency response, including 26/11and enhanced 26/11 (in Indian scenario) services, network failures take on life-or-death significance. Reliability is a network's ability to perform a designated set of functions under certain conditions for specified operational times. The stated operation conditions include not only network traffic, node mobility, terrain, and weather, but also equipment stochastic failures in the process of network operation. In other words, reliability is the study of failures, their causes and effects. It is the most important characteristics of any system or product or service quality as the things have to be working smoothly before analyzing other quality parameters. Software and hardware have different design principles and mechanisms that also make them different in failure mechanisms. Software failures may be due to errors, ambiguities, misinterpretation of the specification, incompetence in writing code, inadequate testing, and incorrect usage of the software [1]. Protocols, nodes, operator and procedural errors, along with environmental factors can each cause a system failure. Hardware components that may fail include processor, Memories, disk drive hardware, or power supplies [2].

Hardware components exhibit the failure characteristics with time as shown by bathtub curve in Fig. 1. Period A, B and C stands for burn-in phase, useful life phase and end-of-life phase.

Software and hardware have basic differences that make them different in failure mechanisms. Hardware faults are mostly physical faults, while software faults are design faults, which are harder to visualize, classify, detect, and correct [2]. In software, there is no corresponding counterpart for "manufacturing" as hardware manufacturing process.

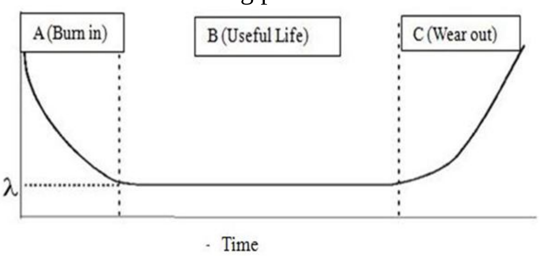

Fig. 1. Bathtub Curve for Hardware Reliability

Failure of a single component or single node can directly influence reliability of the overall system. Software reliability, however, does not show the same characteristics similar as hardware. A possible curve is shown in Fig. 2 if we projected software reliability on the same axes. There are two major differences between hardware and software curves. One difference is that in the last phase, software does not have an increasing failure rate as hardware does. In this phase, software is approaching obsolescence. There is no motivation for any upgrades or changes to the software. Therefore, the failure rate will not change. The second difference is that in the useful-life phase, software will experience a drastic increase in failure rate each time an upgrade is made. The failure rate levels off gradually, partly because of the defects found and fixed after the upgrades.

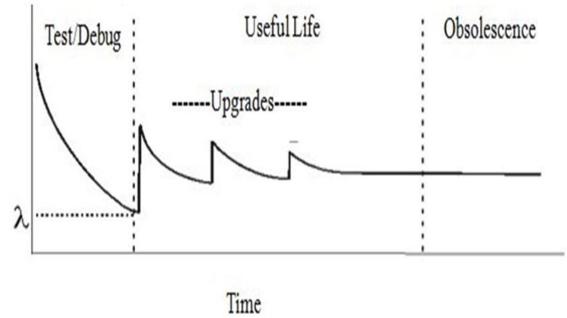

Fig. 2. Revised Bathtub Curves for Software Reliability

The upgrades in Fig. 2 imply feature upgrades, not upgrades for reliability. For feature upgrades, the complexity of software is likely to be increased, since the functionality of software is enhanced. Even bug fixes may be a reason for more software failures, if the bug fix induces other defects into software. For reliability upgrades, it is possible to incur a drop in software failure rate, if the goal of the upgrade is enhancing software reliability, such as a redesign or reimplementation of some modules using better engineering approaches.

Reliability is a measure of the time between failures occurring in a system. Hardware and software components have different failure characteristics. Although formulas exist to predict hardware reliability based on historical data, it is difficult to find formulas for predicting software reliability.

II. RELATED WORK

Many models and protocols have been presented to improve and evaluate network reliability. Issues related to design constraints and their effects on the reliability are discussed in [4-19].

In critical area of operations wireless computer network needs high reliability. Reliability modeling and fault analysis has become an integral part of the computer network design process. It is also important to evaluate the existing MANET reliability before any hardware or software upgrade. Multipath routing has been a important technique in MANETs. Multipath routing provides many performance benefits, including reliability.

The system reliability degrades due to different failure mechanisms. Work done in the area of reliability provides robustness to both short term and long term node failures in ad hoc networks. These failures could be a result of environmental factors, electrical failure or compromises. The number of node-disjoint paths that can be found between a source and a destination depends on the density of nodes in the network. Reliable path is defined in available literature to capture the notion of routing reliability and evaluate the performance of Reliable node deployment strategies in terms of the probability that a reliable path is found between a source and a destination.

The three aspect of modeling k path routing in ad-hoc network load balancing, delay and reliability are investigated in literature.

The methods used to improve the reliability of engineering systems [20] are Usage of better components, System Simplification or removing complexities, De-rating, Redundancy, Controlling the environment, Maintenance, Removing early failures by burn-in etc.

The most important technique is the redundancy for reliability improvement. In this technique more number of components than actually required for operation is connected in parallel. Active, stand-by, k-out-of-n good system, etc. are the important redundancy techniques used. The effect of applying redundancy is increased weight, cost, and volume of the system. Appropriate trade-offs are essential to optimize or maximize the effectiveness of applying redundancies in system design. Redundancy is the easiest method of improving reliability. By using redundancy any level of system reliability can be achieved, but it increases Volume, weight, and cost in direct proportion to the number of redundancies used. This technique should be used as a technique of reliability improvement when all other methods fail or are unacceptable due to technical or managerial reasons.

III. SYSTEM MODELING

The NetSim simulator used doesn't have a mechanism to fail the nodes. Therefore, in the view of realistic scenarios, a node failure model based on the constant hazard model is designed and injected to the simulator to fail the nodes. The node failure model has been simulated to obtain the results and evaluate the reliability and performance of MANET. On the basis of the results the effects of node failure have been observed. Further, a Reliable Node Model is designed to subsidize the effect of node failure model.

A. NODE FAILURE MODEL

Reliability is important concern in wireless ad-hoc networks. Many existing algorithms assume the availability of precise information, which is unrealistic due to the dynamic nature of MANET. Therefore, a node failure model is designed and injected to MANET that accounts for uncertainty and the failures of nodes. The node failure model is based on constant hazard model is designed to study the effect of node failure on reliability and performance of MANET.

Constant Hazard Model

In constant hazard model, the hazard rate function does not change with time. The constant hazard function can be expressed in the form $z(t) = \lambda$, which is a constant and independent of time. Many products, particularly electronic components, exhibit this characteristic. Therefore, it is proposed to use constant hazarded model for performance analysis and prediction of MANET. In this model, the number of iterations is obtained by dividing total time with granularity. For each iteration, the probability is calculated using λ and granularity values. The formula is given below:

$$\text{Probability} = 1 - \exp(-\lambda \times \text{Granularity})$$

Where $\lambda = 1/\text{MTTF}$ (Mean Time to Failure)

A random number is generated for each node in a particular iteration. A node is failed if random number generated is less than the probability value. This procedure is repeated for all the iterations.

Customizing NetSim

Custom code is developed in NetSim (NetSim-Network Simulator) using VC++. In this section,

Visual Studio 2005 is used to create the dynamic link loader (dll). The primitives of the protocol have been modified as per the requirement and the properties have been set up for the simulation scenarios.
The constant hazard model outputs a file which indicates the time at which various nodes fail. Using this as an input to NetSim the following modifications made to the source code to fail the nodes.

Failing the Nodes in NetSim

- Using Device Id: Each device has been given a Device ID by which that device is to be identified out of several devices.
- The fn_NetSim_DSR_Run() is the main function to handle all the protocol functionalities. As per the logic the device which satisfies the condition gets failed.

B. RELIABLE NODE MODEL DESIGN USING GRID STRUCTURE

A given source and a destination requires large amount of overhead to provide a reasonable degree redundancy. But, more node-disjoint paths are essential especially in case of critical areas of applications to increase the reliability of network. Thus, reliable nodes can be deployed at junctions connecting multiple node-disjoint segments. In a disaster management one can have unreliable low power handhelds, whereas there could be the more reliable, power capable and secure high end nodes can be deployed at important location. It would be expensive to deploy a large number of these Reliable nodes and therefore at strategic locations only we can deploy few such nodes. The reliability of a wireless network depends not only on the components' reliability but also on the degree of redundancy in the network's topology and the distribution of nodes in the network. This is the reason we have considered the reliable node model to ensure that high level of reliability.
The reliable node model uses the concept of reliable node and reliable path to get the benefits of both models and provide more reliable system. It is important to position the reliable nodes so as to maximize their utility.
To reduce the effect of node failure and improve the reliability of MANET, the new reliable node model is used and it is based on the following considerations:

- Constructed a grid structure to ensure high level of reliability
- Initially nodes stay stall (paused) for a certain time.
- Start to move with average speed for given time within area, after the node reach their destination, they stay stall in their position for some time (pause time).
- After that node again choose another random destination in the simulation area and move towards them.
- It is important to provide redundancy(parallelism) in terms of providing multiple node-disjoint paths from a source to destination
- For providing multiple node -disjoint paths we can place reliable nodes (high end laptops or multiple computing systems) R(nodes), for efficient operation in DSR.

IV. RESULTS AND ANALYSIS

In the scenario, it is tried to extend the number of total node to 39 to show the effect of increase of nodes on all the three models. The scenario setup is shown in Fig. 3. All the three models are simulated and throughput and mean delay are obtained. The speed is set at 0 m/s to arrange the nodes in grid structure to enhance the reliability. Finally, result is compared with MTTF and number of failed nodes.

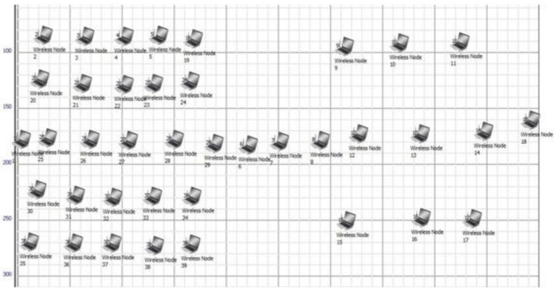

Fig. 3. MANET Simulation Scenario

The protocol that has been used in simulation is Modified DSR. The simulation is run on an Intel Core-2 Duo processor at 400 MHz, 2 GB of RAM with Windows 7 Operating System. The simulation input parameters are given as follows:

- A rectangle area is chosen to have longer distances between the nodes than in a quadratic area, i.e. packets are sent over more hops.
- IEEE 802.11 MAC
- Two ray ground propagation model
- Node mobility defined by random waypoint movement model.
- Constant bit rate traffic
- UDP
- Simulation Time (ms) = 1000000.00
- Number of Nodes = 39
- Speed = 0 m/s
- Source ID- 1 and Destination ID- 18

The numerical data and results are obtained at speed 0 m/s with total 39 nodes. The simulation results of node failure model and reliable node model are shown in the Table I to Table III.
After simulation, various performance parameters have been obtained as results. Graphs have been plotted for various parameters to analyze the effects in all the three models.
The failed nodes vs. throughput characteristics are shown in the Fig. 4. In case of reliable model with 3

and 11 reliable nodes, it is observed that the throughput is almost constant with increase in failed nodes up to 20. After that it is decreasing sharply with increase in the number of failed nodes up to 28. Surprisingly, when number of failed nodes increase beyond 28, throughput is increasing sharply with increase in failed nodes up to 37 in case of 11 reliable node model. In node failure model, throughput is decreasing gradually with increase in number of failed nodes up to 5. Afterwards it is increasing gradually up to 20 failed nodes. But, further increase in number of failed nodes, the throughput decreases significantly. It is to be mentioned that reliability of the MANET is optimum when only three nodes (i. e. 6, 7 and 8) are used at strategic locations.

Now, an attempt is made to compare the throughput with MTTF for all the three cases.

Table: I Simulation Results of Node Failure Model

Failure Rate (1/s)	MTTF (s)	Granularity	No. of Failed nodes	Packets Transmitted	Packets Received	Payload Transmitted (Bytes)	Payload Received (Bytes)	Throughput (Mbps)	Mean Delay (µs)
0.0001	10000	10	0	49999	83681	4999900	8368100	0.066945	101922213
0.0002	5000	10	1	49999	83681	4999900	8368100	0.066945	101922213
0.001	1000	10	5	49999	77097	4999900	7709700	0.061678	157398953.2
0.01	100	10	20	49999	88890	4999900	8889000	0.071112	350830750.3
0.013333	75	10	28	49999	1036	4999900	103600	0.000829	1700908.077
0.02	50	10	32	49999	3840	4999900	384000	0.003072	4794663.754
0.033333	30	10	36	49999	1339	4999900	133900	0.001071	7811792.596
0.05	20	10	37	49999	3393	4999900	339300	0.002714	8686823.915
0.1	10	10	37	49999	827	4999900	82700	0.000662	1432274.738

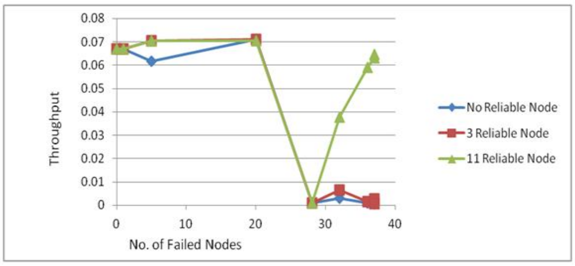

Fig. 4 Number of Failed Nodes vs Throughput Characteristics

The MTTF vs. throughput characteristics are shown in Fig. 5. It is observed that throughput is decreasing slowly with increase in MTTF up to 5000 in case of 3 reliable node and 11 reliable node models. Afterwards it is constant with increase in MTTF. In node failure model, throughput is decreasing sharply with increase in MTTF up to 1000 and afterwards it is increasing gradually up to MTTF of 5000 and then it is remain constant with increase in MTTF.

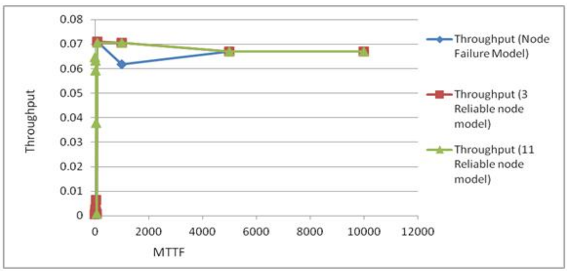

Fig. 5 Throughput vs. MTTF Characteristics

Table: I Simulation Results with Reliable Nodes as 6, 7 and 29

Failure Rate (1/s)	MTTF (s)	Gran-ularity	No. of Failed nodes	Packets Transmitted	Packets Received	Payload Transmitted (Bytes)	Payload Received (Bytes)	Throughput (Mbps)	Mean Delay (µs)
0.0001	10000	10	0	49999	83681	4999900	8368100	0.066945	101922213
0.0002	5000	10	1	49999	83681	4999900	8368100	0.066945	101922213
0.001	1000	10	5	49999	88147	4999900	8814700	0.070518	225539001.4
0.01	100	10	20	49999	88685	4999900	8868500	0.070948	185201187.8
0.013333	75	10	28	49999	1036	4999900	103600	0.000829	1700908.077
0.02	50	10	32	49999	7993	4999900	799300	0.006394	32116038.12
0.033333	30	10	36	49999	2098	4999900	209800	0.001678	15372997.98
0.05	20	10	37	49999	3405	4999900	340500	0.002724	10852588.78
0.1	10	10	37	49999	827	4999900	82700	0.000662	1432274.738

Table: III Simulation Results with Reliable Nodes 2,21,27,33,39,9,13,17,6,7 and 29

Failure Rate (1/s)	MTTF (s)	Granularity	No. of Failed nodes	Packets Transmitted	Packets Received	Payload Transmitted (Bytes)	Payload Received (Bytes)	Throughput (Mbps)	Mean Delay (µs)
0.0001	10000	10	0	49999	83681	4999900	8368100	0.066945	101922213
0.0002	5000	10	1	49999	83681	4999900	8368100	0.066945	101922213
0.001	1000	10	5	49999	88147	4999900	8814700	0.070518	225539001.4
0.01	100	10	20	49999	88222	4999900	8822200	0.070578	185037593.2
0.013333	75	10	28	49999	1036	4999900	103600	0.000829	1700908.077
0.02	50	10	32	49999	47216	4999900	4721600	0.037773	263657793.4
0.033333	30	10	36	49999	73800	4999900	7380000	0.05904	208032968.6
0.05	20	10	37	49999	78855	4999900	7885500	0.063084	99837973.12
0.1	10	10	37	49999	80878	4999900	8087800	0.064702	115988077.1

Now, it is tried to correlate the effect of number of failed nodes on the mean delay for three models. The failed nodes vs. mean delay characteristics are shown in Fig. 6. In case of node failure model, it is observed that the mean delay is increasing sharply with increase in number of failed nodes up to 20 and afterwards it is decreasing sharply with increase till the number of failed nodes reaches 28. After that it further decreases up to failure nodes 32 and again increases slowly with increase in failed nodes up to 37. In reliable node model with 3 and 11 reliable nodes, mean delay is overlapping in both cases number of failed nodes up to 28 and is usually less in comparison to node failure model. After this, the mean delay is rapidly increases with increase in failed nodes up to 32 in reliable node model with 11 reliable nodes. Further increase in number of failed nodes, mean delay gradually decreases up to 37 and decreases sharply as number of failed nodes increases in case of the reliable node model with 11 reliable nodes. This signifies that mean delay is not varying uniformly but, it is at lower when reliable nodes are introduced and even increased in the network.

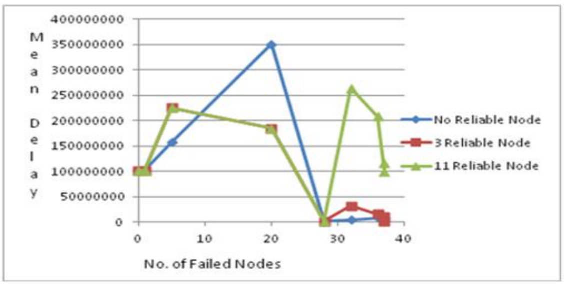

Fig. 6 Number of Failed Nodes Vs Mean Delay Characteristics

The MTTF vs. mean delay characteristics are shown in Fig. 7. In node failure model, it is observed that the mean delay is decreasing sharply with increasing in MTTF up to 5000 and afterwards it remains constant. In case of 3 and 11 reliable node model, mean delay characteristics are overlapping in both cases.

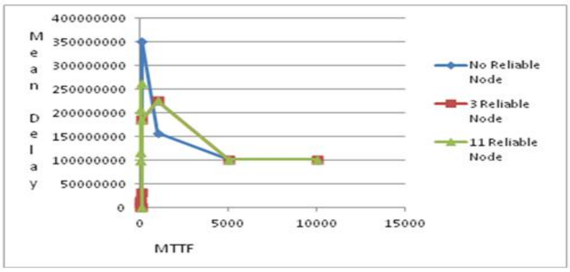

Fig. 7 MTTF vs. Mean Delay Characteristics

CONCLUSION

It is found that in most of the cases, software system reliability increases when the version goes up. Since software robustness is one aspect of software reliability, this result indicates that the upgrade of those software systems should have incorporated to reliability upgrades. In the hardware components of ad-hoc network, the reliability of logic boards and disks are much higher than that of the cables, connectors, LAN cards, and power supply unit. To increase the hardware system reliability, reliability of cables, connectors, LAN cards, and power supply must be increased. The reliability of MANET can be increased by increasing the reliability of hardware or/and software systems. For making MANET reliable in critical area of operations, some high end reliable nodes with grid structure has been used. Speed is set to 0 m/s because in critical area of operation some moment of time reliability of the MANET is essential in comparison to mobility of nodes. The simulation results of reliable node model shows that the overall throughput and reliability are improved with respect to without reliable node model.

REFERENCES

[1] Software Reliability, Carnegie Mellon University, Dependable Embedded Systems, Spring 1999.
[2] Implementing Systems for Reliability and Availability, Library, Microsoft Technet, Microsoft © 2013.
[3] Computer Systems Engineering/Reliability models, a Wikimedia Project, modified on 8 June 2009.
[4] Subhasish Mitra, Nirmal R. Saxena and Edward J. McCluskey, A design diversity metric and reliability analysis for redundant systems, IEEE Transactions on Computers Vol. 51 No. 5, 2002.
[5] Ngo Duc Thuan, Hiroki Nishiyama, Nirwan Ansari, and Nei Kato, On Performance Evaluation of Reliable Topology Control Algorithms in Mobile Ad Hoc Networks(Invited Paper),, 978-1-4244-3574-6/10/ © IEEE 2010.
[6] Trisha Biswas, Redundancy-based Approaches in Wireless Multihop Network Design, North Carolina State University, 2011.
[7] Ivanovitch Silva , Luiz Affonso Guedes , Paulo Portugal and Francisco Vasques , Reliability and Availability Evaluation of Wireless Sensor Networks for Industrial Applications, OPEN ACCESS sensors, ISSN 1424-8220, 2012.
[8] Soon Y. Oh and Mario Gerla, Robust MANET Routing using Adaptive Path Redundancy and Coding, University of California, Los Angeles Los Angeles, COMSNETS 2009.
[9] V. Janardhan babu, S. Jeelan, G. Bala Gangadhara, and D. Arun Prasad, Improving Network Reliability Evaluation of Distance-Vector Routing with Nodes Reliabilities Stochastic in Nature, International Journal of Computer Science & Communication Networks,Vol 1(3), 354-359, December - January 2011-2012.
[10] Seyed Ali Sadat Noori and Elham Sahebi Bazaz, Improving Survivability in Wireless Ad Hoc Network, World Academy of Science, Engineering and Technology, 2012.
[11] Mrs. K. Deeepika Reddy and K. Ramakrishna, Reliable Computing In Ad Hoc Networks International Journal of Computer Trends and Technology (IJCTT) – volume 4 Issue 6–June 2013.
[12] Jose Emmanuel Ramirez-Marquez and David W. Coit, Multi-state Component Criticality Analysis for Reliability Improvement in Multi-state Systems, Reliability engineering and System safety, Vol. 92, No. 12, pp. 1608-1619 Special Issue on ESREL 2005.
[13] Tolga Numanoglu, Improving the Reliability and Performance of Real-TimeCommunications in Mobile Ad Hoc Networks, University of Rochester Rochester, New York, 2009.
[14] Maggie Cheng, Yingshu Li and Ding-Zhu Du (Eds.), Combinatorial Optimization in Communication Networks, Kluwer Academic Publishers, 2005.
[15] Hossen Mustafa, Xin Zhang, Zhenhua Liu, Student Member, IEEE, Wenyuan Xu, Member, IEEE, and Adrian Perrig, Senior Member, IEEE, Jamming-Resilient Multipath Routing, IEEE Transactions on Dependable and Secure Computing, Vol. 9, No. 6, November/December, 2012.
[16] K. S. Wang, F. S. Hsu, and P. P. Liu, Modeling the Bathtub shape hazard rate function in terms of reliability, Reliability Engineering and System Safety, 75, 397-406, 2002.
[17] Zhenqiang Ye, Srikanth V. Krishnamurthy, Satish K. Tripathi, A Framework for Reliable Routing in Mobile Ad Hoc Networks, IEEE INFOCOM- 2003.
[18] Kuong-Ho Chen, Chyi-Ren Dow, Sheng-Chang Chen, Yi-Shiou Lee and Shiow-Fen Hwang, HarpiaGrid: A Geography-Aware Grid-based Routing Protocol for Vehicular Ad Hoc Networks, Journal of Information Science and Engineering 26, 817-832, 2010.
[19] Sushil Chandra Dimri Kamlesh C. Purohit Durgesh Pant, Improvement of performance of mobile ad hoc network using k- path splittable traffic flow scheme, Int. J. Comp. Tech. Appl., Vol 2 (6), 1911-1917, 2011.
[20] V.N.A. Naikan, a book on “Reliability Engineering and Life Testing”, PHI Learning Private Ltd. New Delhi, Copyright 2009.
[21] Anil Choudhary, Dr. O. P. Roy and Dr. T. Tuithung “Reliability Improvement Techniques in Mobile Ad-Hoc Network”, CONIAPS-2014.

★★★