PERFORMANCE EVALUATION OF A LAYERED WSN USING AODV AND MCF PROTOCOLS IN NS-2

Apoorva Dasari and Mohamed El-Sharkawy

Purdue School of Engineering and Technology
melshark@iupui.edu

ABSTRACT

In layered networks, reliability is a major concern as link failures at lower layer will have a great impact on network reliability. Failure at a lower layer may lead to multiple failures at the upper layers which deteriorate the network performance. In this paper, the scenario of such a layered wireless sensor network is considered for Ad hoc On-Demand Distance Vector (AODV) and Multi Commodity Flow (MCF) routing protocols. MCF is developed using polynomial time approximation algorithms for the failure polynomial. Both protocols are compared in terms of different network parameters such as throughput, packet loss and end to end delay. It was shown that the network reliability is better when MCF protocol is used. It was also shown that maximizing the min cut of the layered network maximizes reliability in the terms of successful packet transmission of network. Thetwo routing protocolsare implemented in the scenario of discrete network event simulator NS-2.

KEYWORDS

AODV, MCF and NS-2.

1. INTRODUCTION

The advancements in wireless communication technologies enabled large scale wireless sensor networks (WSNs) deployment. As there is no fixed infrastructure between wireless sensor networks for communication, routing becomes an issue in large number of sensor nodes deployed along with other challenges of manufacturing, design and reliability of these networks [5-8].

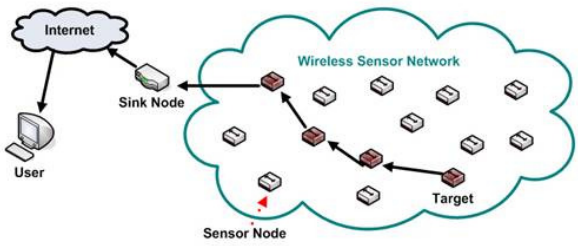

Figure 1: Wireless Sensor Network.

Natarajan Meghanathan et al. (Eds) : ICAIT, ICDIPV, ITCSE, NC - 2014
pp. 55–65, 2014. © CS & IT-CSCP 2014 DOI : 10.5121/csit.2014.4707

The main issue of concern in this paper is Reliability. WSN network architecture is often layered. Reliability issues in layered networks may be often due to two reasons:

- Link failures: A link failure occurs when the connection between two devices (on specific interfaces) is down.

- Device failures: A device failure occurs when the device is not functioning for routing/forwarding traffic.

Lower layers generally experience random link failures. Each link failure at lower level may lead to multiple failures at the upper layers.There are many proposed concepts which tend to improve the reliability of any network. Modern communication networks are designed with one or more electronic layers (e.g.,IP, ATM, SONET) built on top of an optical fiber network. The survivability of such networks under fiber failures largely depends on how the logical electronic topology is embedded onto the physical fiber topology using lightpathrouting.However, assessing reliability performance achieved by a lightpath routing can be rather challenging because seemingly independent logical links can share the same physical link, which can lead to correlated failures. To avoid these kinds of failures, there are various routing protocols that have been proposed for routing data in wireless sensor networks. The mechanisms of routing consider the architecture and application requirements along with the characteristics of sensor nodes. One of the widely used protocols for data transmission in WSN is the following AODV routing protocol.

1.1 AODV Protocol

There are two types of routing protocols which are reactive and proactive. In reactive routing protocols, the routes are created only when source wants to send data to destination, whereas proactive routing protocols are table driven. Being a reactive routing protocol, AODV uses traditional routing tables, one entry per destination and sequence numbers are used to determine whether routing information is up-to-date and to prevent routing loops.

The maintenance of time-based states is an important feature of AODV which means that a routing entry which is not recently used is expired. The neighbors are notified in case of route breakage. The discovery of the route from source to destination is based on query and reply cycles and intermediate nodes store the route information in the form of route table entries along the route. Control messages used for the discovery and breakage of route are Route Request Message (RREQ), Route Reply Message (RREP), Route Error Message (RERR) and HELLO Messages.

When a source node does not have routing information about destination, the process of the discovery of the route starts for a node with which source wants to communicate. The process is initiated by broadcasting of RREQ. On receiving RREP message, the route is established. If multiple RREP messages with different routes are received then routing information is updated with RREP message of greater sequence number.

a) Setup of Reverse Path:

The reverse path to the node is noted by each node during the transmission of RREQ messages. The RREP message travels along this path after the destination node is found. The addresses of the neighbors from which the RREQ packets are received are recorded by each node.
b) Setup of Forward Path:

The reverse path is used to send RREP message back to the source but a forward path is setup during transmission of RREP message. This forward path can be called as reverse to the reverse path. The data transmission is started as soon as this forward path is setup. The locally buffered data packets waiting for transmission are transmitted in FIFO-queue.

The following example shows how data transmission takes place using AODV protocol:

- Node 1 sends RREQ to 2, 3, 4:
  "*Any one has a route to 15 fresher than 3. This is my broadcast #10*"
- Nodes 2, 3, 4 send RREQ to 5, 6, 7
- Node 3 has 3-5-8-9-10 Sequence #1
- Node 4 has 4-6-8-10 Sequence #4
- Node 4 responds. Node 3 does not respond.

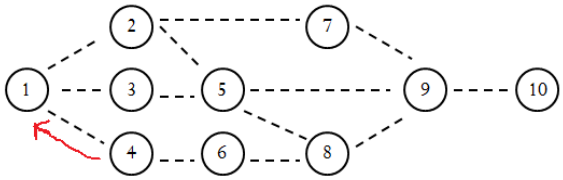

Figure 2: AODV Protocol

1.2 MCF protocol

The MCFlightpath routing algorithm MCFMinCut can be formulated as an integer linear program (ILP): MCFMinCut : Minimize ρ, subject to:ρ ≥X(s,t)∈EL w(s, t)fstij∀(i, j) ∈EPfstij∈ {0, 1}{(i, j) : fstij= 1} forms an (s, t)-path in GP , ∀(s, t) ∈EL,where w(s, t) is the weight assigned to logical link (s, t).

The optimal lightpath routing under this algorithm is determined by the weights w(s, t). For example, if w(s, t) is set to 1 for all logical links, the above formulation will minimize the number of logical links that traverse the same fiber. In other words, this uniform weight function treats each logical link equally, and seeks to minimize the impact of a single physical link failure on the number of disconnected logical links.

However, the connectivity is not well captured under this function since the logical network may remain connected even when a large number of logical links fail. In order to better account for the connectivity, the weight function w(s, t) = 1 /MinCutL(s,t) is used, where MinCutL(s, t) is the size of the min-cut between nodes s and t in the logical topology. Intuitively, this weight function attempts to minimize the impactof a single fiber failure to the logical connectivity, where impact

is defined to be the total sum of weight of the logical links that traverse the fiber. Since the weight is defined to be 1/MinCutL(s,t) , a logical link that belongs to a small cut will contribute more weight than a logical link in a large cut.

2. PROPOSED IMPLEMENTATION

The above two protocols are implemented on a WSN network and the characteristics of the network in both cases, in terms of different network parameters, are compared. There are some network simulators that require commands or scripts while other simulators are GUI driven. In network simulation, the behavior of network models is extracted from information provided by network entities (packets, data links, and routers) by using some calculations. In order to assess the behavior of a network under different conditions different parameters of the simulator (environment) are modified.

Network Simulator (NS) is an object-oriented, discrete event driven network simulator that simulates a variety of IP networks, written in C++ and OTcl. It is primarily useful for simulating local and wide area networks. It implements network protocols such as TCP and UDP, traffic behavior such as FTP, Telnet, Web, CBR and VBR, router queue management mechanism such as Drop Tail, RED and CBR, routing algorithms such as Dijkstra, and more. NS also implements multicasting and some of the MAC layer protocols for LAN simulations. NS develops tools for simulation results display, analysis and converters that convert network topologies to NS formats.

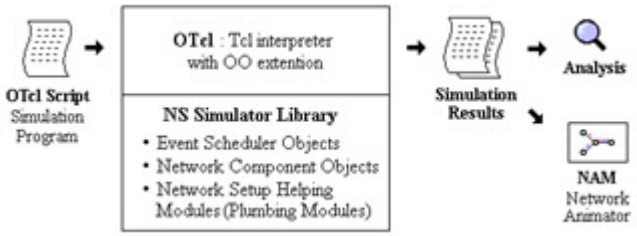

Figure 3: NS Simulator.

The generic script structure in NS-2 has the following steps: Create Simulator object, Turn on tracing, Create topology, Setup packet loss, link dynamics, Create routing agents, Create application and/or traffic sources, Post-processing procedures (i.e. nam), and Start simulation.

For designing any protocol in NS-2, the major steps to follow is define the following in Tcl scripts: Hello Packets, Timers used for Broadcast, Interval, Hello and Functions:

a) General for Packet Handling
b) Routing Table Management
c) Broadcast ID Management
d) Packet Transmission Management
e) Packet Reception Management

The flow of protocol in NS-2 is as follows. Let us consider AODV protocol for example.

1.In the TCL script, when the user configures AODV as a routing protocol by using the command,
$ns node-config –adhocRouting AODV
The pointer moves to the "start" and this "start" moves the pointer to the Command function of AODV protocol.

2. In the Command function, the user can find two timers in the "start"
* btimer.handle((Event*) 0);
* htimer.handle((Event*) 0);
3. Let's consider the case of htimer, the flow points to HelloTimer::handle(Event*) function and the user can see the following lines:

```
agent ->sendHello();
double interval = MinHelloInterval + ((MaxHelloInterval - Min-
HelloInterval) * Random::uniform());
assert(interval ->= 0);
Scheduler::instance().schedule(this,  &intr,interval);
```

These lines are calling the sendHello() function by setting the appropriate interval of Hello Packets.

4. Now, the pointer is in AODV::sendHello() function and the user can see Scheduler::instance().schedule(target , p, 0.0) which will schedule the packets.

5. In the destination node AODV::recv(Packet*p, Handler*) is called, but actually this is done after the node is receiving a packet.

6. AODV::recv(Packet*p, Handler*) function then calls the recvAODV(p) function.

7. Hence, the flow goes to the AODV::recvAODV(Packet *p) function, which will check different packets types and call the respective function.

8. In this example, flow can go to case AODVTYPE HELLO:
recvHello(p);
break;

9. Finally, in the recvHello() function, the packet is received. The general trace format is shown in Figure 4

Every protocol generally uses some weight function for each link to traverse through the network. As we have seen above, AODV protocol uses sequence number to decide on the route which it should traverse. So, sequence number acts as weight in the protocol. In the same way, shortest path algorithm considers minimum number of hops as the weight. As we see by definition on MCF algorithm, it considers 1/MinCutL(s,t) as the weight function where MinCutL(s, t) is the size of the min-cut between nodes s and t in the logical topology.

According to the network topology, Mincut is the number nodes in a route which satisfies both the conditions of minimum number of hops and minimum weight of the route. Here, weight of the route is sum of weights of all the links in the route.

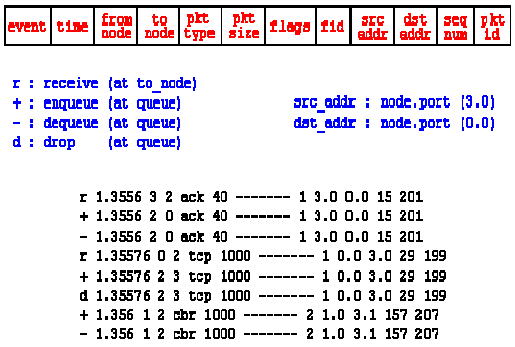

Figure 4: General Trace Format.

MCF algorithm greedily takes the path which has minimum weight and then checks for the condition of minimum number of hops. AODV is taken as the program and it is modified by adding the conditions for MCF algorithm.

3. SIMULATION RESULTS

Software Requirements:

Programming Language: TCL, C++, OTCL
Simulator: NS2 2.35
User Interface: NAM
Operating System Environment: Ubuntu 11.0

Hardware Requirement:

RAM: Min 1GB (Configuration)
Installation: 2GB RAM required
Hard Disk: 40GB
Processor: Minimum Configured with 2.0GHZ speed

Network Parameters:

- Network : WSN
- Number of Nodes : 80
- Routing Protocol : AODV/Lightpath
- Agent : TCP
- Application : CBR
- Communication range 250 unit
- MAC 802.11
- Traffic CBR, 8 Kbps per flow
- # of Flows 50
- Pause Time 5 second
- Max Speed 10 unit / s

In order to analyze statics report of proposed network model, various scenarios are evaluated for determining performance report of cross layer reliability model by employing various experiments. However, link failure always impact performance, the entire study has concerned with security with performance prospective. In any layered networks, link failure always consumes some energy. The main aim here is to compare the two protocols AODV and MCF by computing different scenarios for 80 nodes. Each node configured with defined layered network properties. In order to configure layered network, MAC layer properties and node properties are defined by adjusting parameters.

The next task is to select the routing protocol and define channels for configured network accordingly.

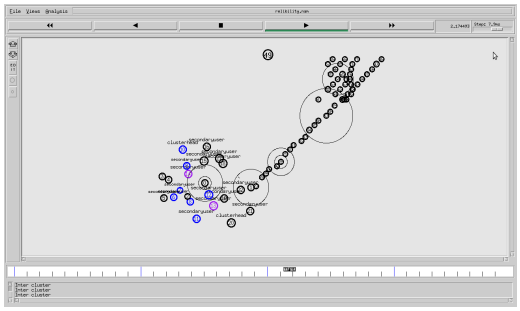

Figure 5: Simulated Network.

The designed network with four source nodes and 80 nodes is shown in Figure 5. After simulating the above network using the two protocols AODV and MCF, trace files are acquired. Then, the comparison is carried out for the two protocols with the help of performance parameters like end-to-end delay, throughput and Packet loss. By observing the results and graphs, the two protocols are analyzed in terms of reliability.

3.1 Throughput

Throughput is the ratio of the total amount of data that a receiver receives from a sender to a time it takes for receiver to get the last packet. The 80 nodes network is configured by assigning server and client nodes. Then, throughput is computed by analyzing server nodes and normal nodes transmission performance. Throughput is defined as the number of packets successfully processed per second. The average throughput rate increases with respect to total amount of packets generated. Figure 6 shows the throughput versus time of the 80 nodes network simulated using both AODV and MCF protocols. The network performance is analyzed in four different time intervals defined as there are 4 source nodes in the network. In each segment, number of packets successfully transmitted through special nodes and average rate of packets delivered in a different timelines are computed. In Figure 6, there is comparison between throughput performance of network using AODV and MCF at a scenario considering the performance of single source node.

The red line represents MCF protocol throughput and the yellow line corresponds to throughput due to AODV in units of bytes/sec. As it is seen from the figure, the simulation of first source node starts at 0.5 sec in both the scenarios. The throughput rate is very high here as Node 1 is the only transmitting node using the entire available bandwidth. This justifies the high performance of Node 1 during the specified interval of time. If we observe, at almost all the points of time, the red line has a higher throughput value compared to the yellow line. This shows that MCF has a better throughput performance.

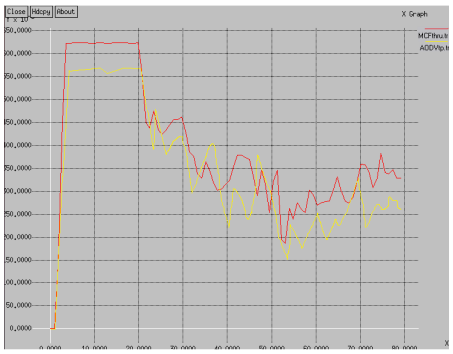

Figure 6: Throughput Comparison.

3.2 Packet Loss

At the physical layer of each wireless node, there is a receiving threshold. When a packet is received, if its signal power is below the receiving threshold, it is marked as error and dropped by the MAC layer.Packets Loss is defined as the total number of packets dropped during the simulation.

Lower the value of packet loss, better the performance of the protocol. In Figure 7, the red line represents AODV protocol and the blue line corresponds to MCF. The performance graph of a third source node which starts at 30 sec in both scenarios is considered. It is clear that packet loss with AODV protocol is higher than that of MCF protocol. At some points of time, packet loss due to AODV protocol is reaching very high peaks around 140 packets lost. The highest loss is around 100 packets in the case of MCF.

3.3 End to End Delay

The End to End delay is the average time taken by a data packet to arrive at the destination. It also includes the delay caused by route discovery process and the queue in data packet transmission. Only the data packets that successfully delivered to destinations are counted.

Average delay = $\sum$ (arrive time – send time)/$\sum$ Number of connections. The lower value of end to end delay means the better performance of the protocol. The end-to-end delay over a path is the summation of delays experienced by all the hops along the path. In order to compute this metric over a wireless channel, each node needs to monitor the number of packets buffered at the network layer waiting for MAC layer service, as well as measuring the transmission failure probability at the MAC layer. The transmission failure probabilityis the probability that a MAC-layer transmission fails due to either collisions or bad channel quality. Figure 8 shows the end to end delay performance for the 80 nodes network.

The red line represents AODV protocol and the blue line corresponds to MCF. As shown from the figure, the performance graph of a third source node which starts at 30 sec is considered for both scenarios. It is clear that end to end delay of packets in the network with AODV protocol is higher than that of MCF. When AODV protocol is used, the peak delay of a packet reaches 700 µ sec where as it is around 550 µ sec in MCF.

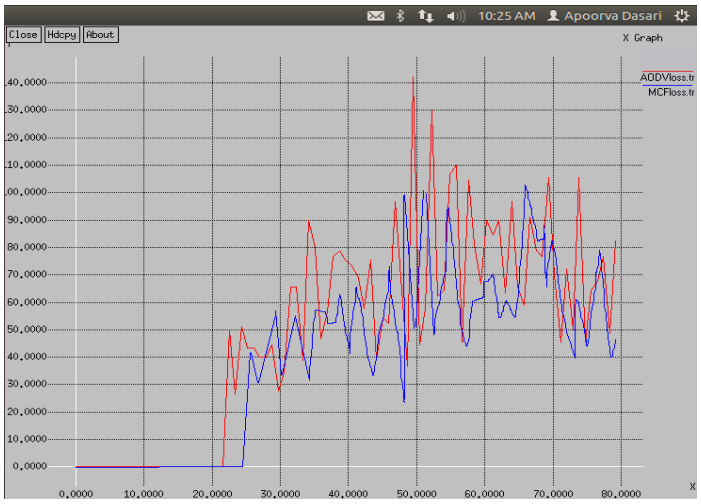

Figure 7: Packet Loss.

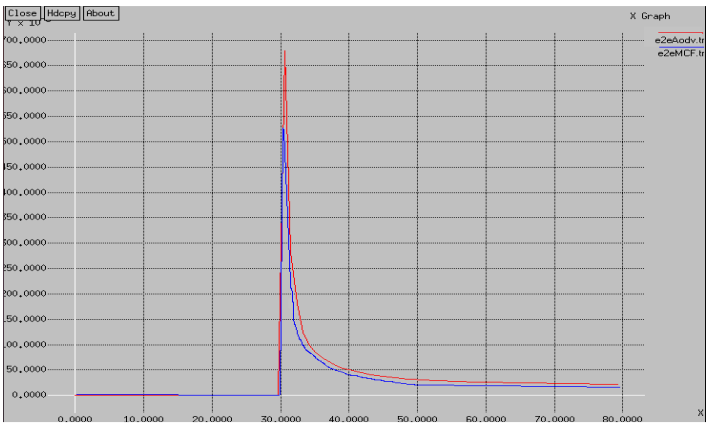

Figure 8: End to End delay.

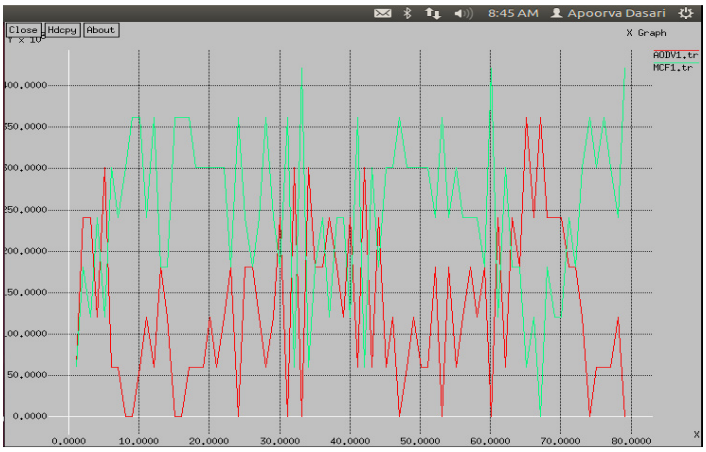

Figure 9: Successful Packet Transmission.

3.4 Successful Packet Transmission:

The trend of successful packet transmission is observed in both the protocols. Figure 9 shows the number of successfully transmitted packets during the simulation time. The green line represents packet transmission in MCF and red line in AODV protocols. At almost all points, MCF has higher successful transmission rate compared to AODV protocol.

4. CONCLUSION

In this paper, Wireless Sensor Network is implemented with AODV and MCF protocols. MCF uses mincut as weight function. Comparison of the performance of both the protocols in terms of different network parameters such throughput, packet loss and end to end delay is carried out. It is observed that in terms of all the network parameters, MCF protocol shows better performance compared to AODV protocol. The comparison in terms of successful packet transmission rate is also observed which showed that MCF protocol has better reliability compared to AODV. Therefore, a more reliable network with better performance can be designed using MCF protocol.

REFERENCES

[1] V. Srivastava and M. Motani, "Cross-layer design: a survey and the road ahead," Communications Magazine, IEEE, vol. 43, no. 12, pp. 112–119, 2005.
[2] K. Lee, H.-W. Lee, and E. Modiano, "Reliability in layered networks with random link failures," MIT, Tech. Rep., 2009.
[3] K. Lee and E. Modiano, "Cross-layer survivability in WDM-based networks," in IEEE INFOCOM, Rio de Janeiro, Brazil, Apr. 2009.
[4] D. Karger, "A randomized fully polynomial time approximation scheme for the all terminal reliability problem," SIAM Review, vol. 43, no. 3, pp. 499–522, 2001.
[5] W. Su and T. L. Lim, "Cross-layer design and optimization for wireless sensor networks," in SNPD-SAWN '06: Proceedings of the Seventh ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, Washington, DC, USA: IEEE Computer Society, 2006, pp. 278–284.
[6] A. Aggarwal, A. Bar-Noy, D.Coppersmith, R. Ramaswami, B. Schieber, and M. Sudan., "Efficient routing in optical networks,"Journal of the ACM, 43(6), November 1996.
[7] Y. Kim, I. Moon and S .Cho, "A comparison of Improved AODV Routing Protocol based on IEEE 802.11 AND IEEE 802.15.4," Department Of Information Media Engineering, School Of Engineering, Taylor's University, Vol: 4, No.2(2009), 132-141.
[8] J. S. Provan and M. Ball, "The complexity of counting cuts and of computing the probability that a graph is connected," SIAM J. Comput., vol. 12, no. 4, pp. 777–788, Nov. 1983.
[9] http://www.ece.ubc.ca/~teerawat/NS2.htm
[10] L. Breslau et al., "Advances in network simulation," IEEE Computer, 33(5):59-67, May 2000.
[11] NS-2 tutorials on Youtube.
[12] Google forums on NS-2.

AUTHORS

Apoora Dasari was born in India in 1991. She received the B. A. Technology Electronics and Communication Engineering degree from Jawaharlal Nehru Technology University, India, in 2012 and M.S degree in Electrical and Computer Engineering from Purdue University, Indianapolis, Indiana in 2014. Her main areas of research interests are networking and wireless communication.

Mohamed A. El-Sharkawy: received the Ph.D. degree in electrical engineering from Southern Methodist University, Dallas, TX, in 1985. He is a Professor of Electrical and Computer Engineering at Purdue School of Engineering and Technology. He is the author of four textbooks using Freescale's and Motorola's Digital Signal Processors. He has published over two hundred papers in the areas of digital signal processing and communications. He is a member of Tau Beta Pi and Sigma Xi. He received several million dollars of industrial research grants, supervised large number of graduate theses and consulted for a large number of companies. He received the Outstanding Graduate Student Award from Southern Methodist University. He received the Abraham M. Max Distinguished Professor Award from Purdue University. He received the US Fulbright Scholar Award in 2008. He received the Prestigious External Award Recognition Award from Purdue School of Engineering and Technology, 2009. He is a reviewer for the National Science Foundation and Fulbright.