International Journal of Systems Signal Control and Engineering Application

Enhanced Energy Efficient Associativity-Based Routing Protocol

[1]V. Anitha, [2]Akilandeswari Jeyapal and [3]Lakshmi Nair
[1]Department of Information and Communication Engineering, Anna University, Chennai, Tamil Nadu, India
[2]Department of Information Technology, Sona College of Technology, Salem, Tamil Nadu, India
[3]Department of Electronics and Communication, Engineering, Dayananda Sagar College of Engineering, Bangalore, India

Key words: MANET, ABR, clustering, methods, source

Corresponding Author:
V. Anitha
Department of Information and Communication Engineering, Anna University, Chennai, Tamil Nadu, India

Page No.: 59-66
Volume: 12, Issue 4, 2019
ISSN: 1997-5422
International Journal of Systems Signal Control and Engineering Application

Abstract: Mobile Ad-hoc Networks (MANETs) are infrastructure less networks in which the nodes are in constant motion. Unlike wired networks, the shortest route between the source and the destination may not be the best. Associativity-Based Routing protocol (ABR) makes use of the most stable route to communicate. However, constant emission of beacons by the nodes involved results in high energy dissipation. This study proposes a method in which cluster-based approach is incorporated in ABR to make it more energy efficient. The results are simulated using NS-2 and a comparison drawn between the two. Thereafter, the behaviour of both the methods is implemented on Xilinx Virtex 5 FPGA and power analyzed using Xilinx Xpower analyzer.

INTRODUCTION

Ad-hoc networks represent a class, highly unpredictable in nature. In addition, there is no fixed infrastructure support. As a result, the routing protocol plays an important role in most of the decision making. The routing protocols in MANETs can be classified into two broad categories: proactive and reactive. Proactive protocols maintain updates about routes while reactive or on-demand protocols form routes whenever a demand arises. Reactive protocols are preferred in MANETs where battery power of each node is critical. There are a number of such reactive protocols. Associativity-based routing protocol is a reactive routing protocol in which routes are established based on stability states. Every node emits beacons to identify itself. All nodes in its vicinity receive the beacons and increment the corresponding associativity ticks. A stability threshold is pre-decided and any path with associativity ticks greater than this threshold is considered stable. Such a strategy is ideal for MANETs as the nodes are in constant motion and the shortest route may no longer exist in the next instant of time. This increases the demand for route reconstruction and in turn, increases the latency and bandwidth requirements. The route selected using ABR has a longer duration comparatively.

However, in high node density networks, except for the nodes involved in the path, most of the nodes are in idle state. These nodes continue to dissipate power. A node loses energy while sending, receiving and forwarding packets that belong to a peer that cannot be reached through it. In ad-hoc networks where nodes have limited battery power, this can prove to be a detrimental factor. There are a number of proposed methods where the focus is to reduce the energy dissipated in reaching the destination. However, there is not a good method in ad-hoc networks as the batteries of the nodes along the path may drain out quickly, giving rise to a situation wherein there is plenty of energy left in the network but the packet delivery is disabled as the backbone of the network is out of energy.

This is where the motivation for this study arises. We have proposed a cluster-based ABR to make it more energy efficient. Thus, the goal of this study is to optimize the energy conservation of ABR by introducing clusters. Through this study, we make an attempt to enhance the existing ABR protocol to make it more energy efficient.

Literature review

Associativity-based routing protocol: Associativity based routing (Toh, 1997) is a protocol that takes into consideration new routing metrics such as longevity of a route, relaying load of intermediate nodes supporting existing routes and knowledge of link capacities of the selected route. In short, prime importance is given to the link duration and an attempt is made to almost avoid route reconstruction. The age of a link is determined based on the reception of beacons from neighbouring nodes. A threshold value called $A_{threshold}$ is computed using 2r/v where r is the transmission range of a node and v is the relative velocity between two nodes.

Route discovery involves a broadcast-query and reply cycle. Each node that needs to communicate sends a broadcast query packet to all its neighbours. Every intermediate node checks if the packet is already processed, in which case, it discards the packet. Else, it forwards the packet after appending its address. Routes are initiated by the source and selected by the destination. It is explained by Meghanathan (2007) that stability comes at a cost of power consumption. It is concluded that higher stability implies higher hop count and in turn greater power consumption. Associativity-based cluster formation in ad-hoc networks is talked about by Sivavakeesar and Pavlou (2005). It makes use of the concept of virtual clusters to geographically divide the network into clusters. It makes use of mobility as a criterion in selecting clusters. Location-based information is incorporated in Kummakasikit *et al*. (2005) while for warding packets. Field Programmable Gate Array (FPGA) implementation of enhanced ABR is proposed in Safa *et al*. (2008), in this work overhearing concept is used to detect the misbehavior nodes.

Power saving techniques in Ad-hoc networks: In MANETS, nodes have limited energy initially which is consumed each time a packet is received, sent or forwarded. There is considerable power dissipation even when the nodes are in an idle state. Extensive research is available on various power saving techniques deployed in MANETs (Chen *et al*., 2002; Kulkarni *et al*., 2012). In a dynamic energy efficient clustering algorithm is proposed wherein nodes with higher energy and less mobility are elected as cluster heads. The energy of these nodes is continuously monitored and when it reaches a critical value, the role is shifted to another node capable of being the cluster head. The performance of this protocol is valued using NS-2.A power saving technique called Span is described in which aim at electing coordinator nodes such that a connected backbone is formed. It is ensured that cases of bandwidth contention are minimized. A back off delay function is used to delay the announcement of the coordinator node. This delay function is a decreasing function of the remaining energy. It also takes into consideration the additional pairs of nodes that would be connected if a particular node becomes a coordinator. To summarize, a coordinator announcement algorithm is explained, a variant of which is used for the implementation. Performance optimization of a reinforcement learning based routing algorithm is illustrated in as applied to ad-hoc networks.

MATERIALS AND METHODS

Consider a group of mobile nodes, in the initial stage, a local decision is made to elect a few nodes as coordinator nodes or cluster heads. This decision is made based on associativity states. We ensure that connectivity of the network is not severely affected and latency is minimally increased. The power saving achieved is tremendous as the density of the nodes increases. This attributes to the fact that only cluster heads, gateway nodes, source and destination are turned on at any given instant of time. Each node has limited battery power. Therefore, to ensure that the cluster heads are not unfairly burdened and deprived of battery power, cluster head re-election is done. At any given instant of time, it is ensured that the coordinator nodes form a connected backbone. A decision on cluster head election is done based on the remaining energy, the additional pairs of nodes that would be connected if the node becomes the cluster head.

In order to verify the behavior of the network in hardware, a prototype is designed with the topology shown in Fig. 1.

We implemented the behavior of the network on a Xilinx Virtex 5 Field Programmable Gate Array (FPGA) and provided each node with buffers to store the incoming packets. We designed a mealy model finite state machine to act as the controller for each node by providing 58 bits for the broadcast query packet taking into consideration the fact that the topology in Fig. 1 may have up to five intermediate nodes. The broadcast query packet is in tandem with that of ABR proposed in Toh (1997). It takes the form shown in Fig. 2.

As the topology in Fig. 1 has 15 nodes, 4 bits are set aside for the node address and 3 bits each for hop count and sequence number. IN in Fig. 2 indicates intermediate node and Aticks stands for associativity ticks. 4 bits are assigned for the associativity ticks.

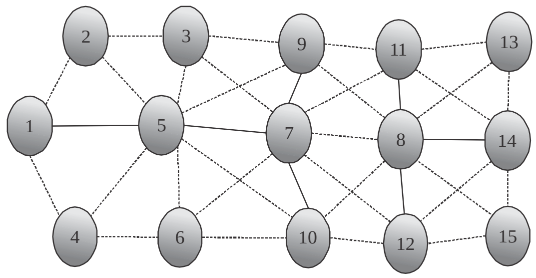

Fig. 1: Network topology with 15 nodes (Dotted lines indicate connectivity)

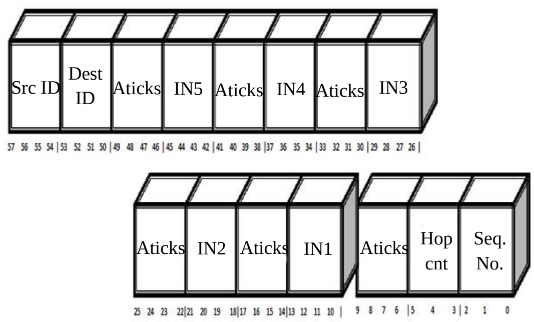

Fig. 2: Broadcast query packet format

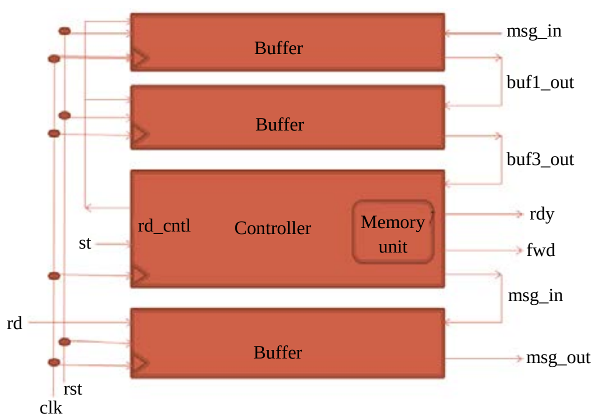

Fig. 3: Implementation of a node

Each node is implemented as shown in Fig. 3. An incoming BQ packet enters the node as msg_in as shown in Fig. 3. This message is read out from the buffer as buf1_out when it receives a rd_cntl signal from the controller. This message is in turn passed into the next buffer from where it comes out as buf3_out when the controller sets the rd_cntl signal as high. The controller, in turn, starts operation when it receives a high 'st' signal.

The controller in Fig. 3 is designed using a mealy model state machine whose state diagram is shown in Fig. 4. The memory unit keeps a tab on the associativity

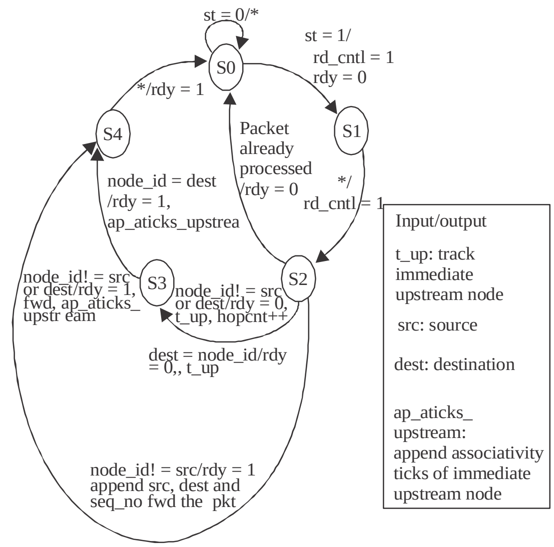

Fig. 4: State diagram of a node during route construction (Broadcast query packet transmission)

ticks of neighbouring nodes. Rdy signal in Fig. 3 and 4 is to indicate that broadcast query has been processed and is now ready to be forwarded. Clk is used to provide synchronization to the network prototype. An external 'rd' signal pushes the processed packet from the buffer as msg_out. The FSM (finite state machine) in Fig. 4 is in the initial state S0 until the 'st' signal goes high. Once 'st' becomes high, it moves on to state S1. From state S1, the node transits to state S2 by making rd_cntl turn high. In state S2, the node ID is compared with the destination ID. If they are found to be equal, a quick transition is made to state S3 after appending the associativity ticks of the immediate upstream neighbour in the BQ packet in the bit positions (49 down to 46). To find the immediate upstream neighbour, the hop count in the position (5 down to 3) is analyzed. For example, if the hop count is 3, it means that the predecessor is the node whose ID is IN3 in the BQ packet. If the node ID doesn't match the destination ID, a check is made to ensure that the packet hasn't been processed earlier. If this is found to be the case, the packet is discarded without further processing and the node goes back to the initial state S0. On the other hand, if the node ID matches the source ID, the initial forwarding of the packet to the source's neighbours happens. In this case, the node moves to state S4 by skipping state S3. If neither of the above cases happen in state S2, it means that the node is an intermediate one. Here, the node analyses the hop count like in the destination case and tracks the upstream node. It finds the immediate upstream neighbour, increments the hop count and appends its own address in the corresponding field in the BQ packet.

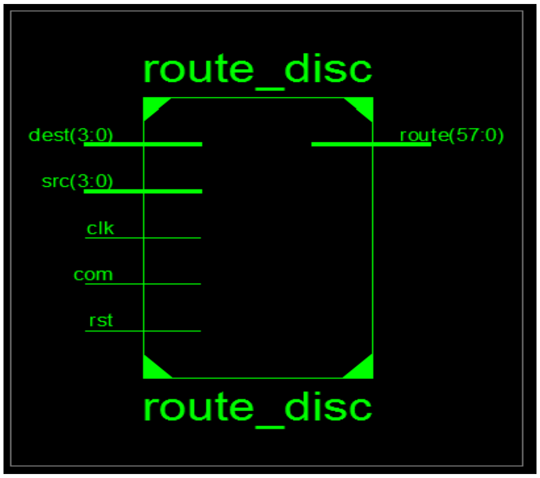

Fig. 5: Black box representation of the main controller

A transition to state S3 is made. In state S3, the associativity ticks of the neighbour from which it received the packet (upstream neighbour) is appended in the corresponding field and rdy is turned high. In state S4, rdy is retained high and the initial state is reached.

To facilitate communication between the nodes, a main controller is designed whose RTL schematic is shown in Fig. 5.

The 'com' signal in Fig. 5 acts as the commence signal for the source node. The source node upon receiving the com signal, appends the source address, the destination address and the sequence number before forwarding the packet to is neighbours. The start signal to the neighbours comes by performing a logical OR operation of the forward signals. Each node gives out a forward signal in which all the fields corresponding to its neighbours are high. For example, let us consider node 2. Its forward signal takes the form:

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01
0	0	0	0	0	0	0	0	0	0	1	0	1	0	1

Each node contains a similar signal with the bits corresponding to its neighbours turned high. This fwd signal is assigned a 0 value till the packets are fully processed. Once, the packet is processed and the rdy signal is made high, the signal takes the value as shown above. To obtain the 'st' signal to each node, a logical OR operation between the corresponding fields of the fwd packet is performed. The external 'rd' signal shown in Fig. 3 is made high when the rdy signal goes high. This way it is ensured that only the fully processed packet comes out of the node as msg_out. Buffer facility is provided to store the various BQ packets coming from different routes. In the original route selection process, dividers are used to calculate the stability. As implementation of dividers takes up a lot of resources, this is done away with by coming up with an alternative. Six stability signals are assigned values corresponding to the intermediate nodes and the destination. This factor is assigned a value 1 if the associativity ticks exceed the threshold set which is 5. These stability factors are added together to form a count called stb_cnt. The maximum value this count can take is 6. Now, the hop count is checked. If it turns out to be zero, it means that the source and the destination are neighbours. In the original route selection process, if there are two routes with equal stability factors, the shortest route is selected. In order to induce similar results, a main stability factor is computed. This is computed by comparing the stb_cnt factor to hop count. If the stb_cnt factor is equal to 1 more than the hop count, the stability factor is assigned a value of 6. As the stb_cnt factor decreases so does the stability factor.

In the second cluster based approach, cluster heads are calculated by finding the nodes with maximum stable neighbours. Once the cluster heads are elected, each of the cluster head is assigned members. In addition, a gateway node is assigned to take care of inter cluster communication.

RESULTS AND DISCUSSION

We conduct two methods to evaluate the performance of our approach. In the first method, we examined the power consumption using NS2 simulation. The simulation environment is given in Table 1.

Figure 6 depicts the Energy consumption of nodes with respect to interval. It is found that average energy consumption is 14% less with cluster ABR when compared to existing ABR. Figure 7 shows average energy consumption of nodes with respect to change in packet size. Here, the energy consumption is 3% less by using cluster-ABR as compared with existing ABR.

Figure 14 shows the cluster head election process. Initially, 5,7 and 8 are selected as the cluster heads out of which 7 is chosen as the gateway as it is connected to both cluster heads. In the second approach to evaluate the performance of the modified protocol, FPGA implementation is done. Table 2 depicts the device utilization summary using Xilinx Virtex 5 FPGA. The simulation results for the test case where 1 is the source and 15 is the destination for the topology in Fig. 1 are shown in the consecutive figures. Figure 8 shows the broadcast query packet (msg_in(1)) at the source before it is forwarded to its neighbours. The source node appends its address, the destination address and the sequence number of the packet before forwarding it to its

Table 1: Summary of NS-2 simulation parameters

Simulation parameters	Values
Simulation area	1000*1000 m
Number of nodes	Mobile Nodes (MN) = 16
Mobility model	Random Waypoint
Speed	Uniform-(0-20) (m/sec)
Pause time	0,60,120,180,240 (sec)
Transmission range	350 (m)
Wireless interface	IEEE 802.11b
Traffic flow	CBR
Transmission power	0.4 (mW)
Reception power	0.3 (mW)
Initial energy	5 (J)

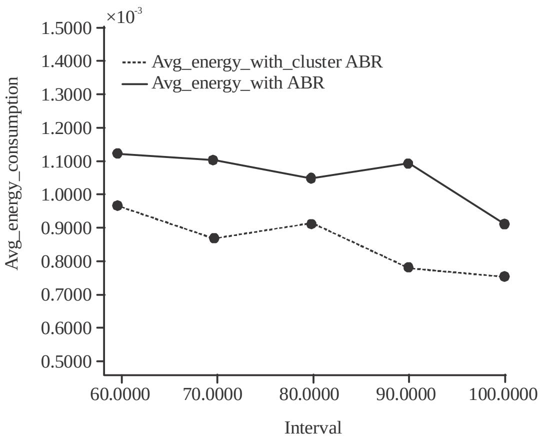

Fig. 6: Interval vs. average energy consumption

neighbours. The appending happens only after the 'com' signal turns high. This signal acts as the 'st' signal of the source node. Figure 9-15 show various stages of the route selection phase of ABR protocol.

Figure 16 shows the cluster head election process. Initially, 5, 7 and 8 are selected as the cluster heads out of which 7 is chosen as the gateway as it is connected to both cluster heads.

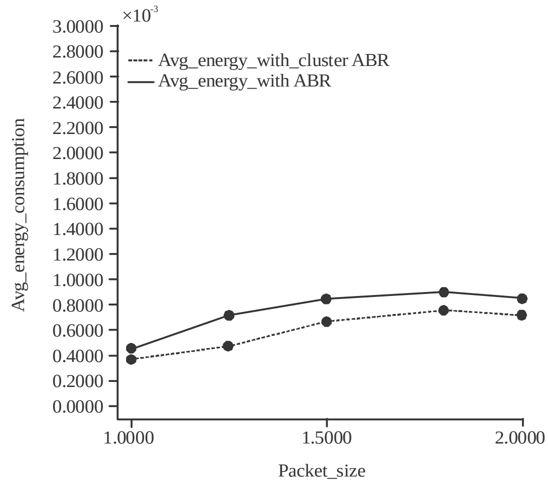

Fig. 7: Packet size vs. average energy consumption

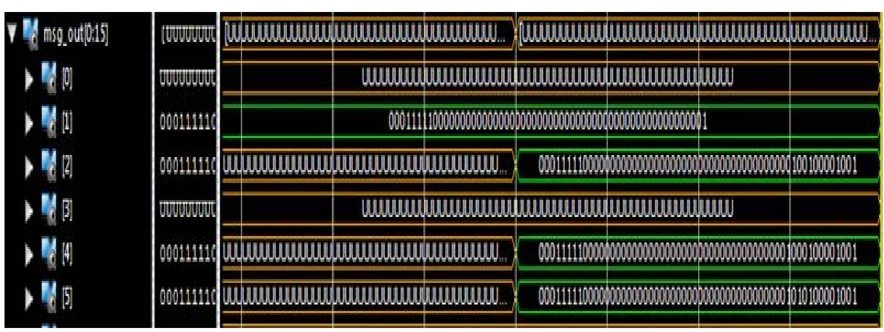

Fig. 8: BQ packet with the source and destination address and the sequence number attached at node 1. Node 1's neighbours (2, 4 and 5) receiving the packet

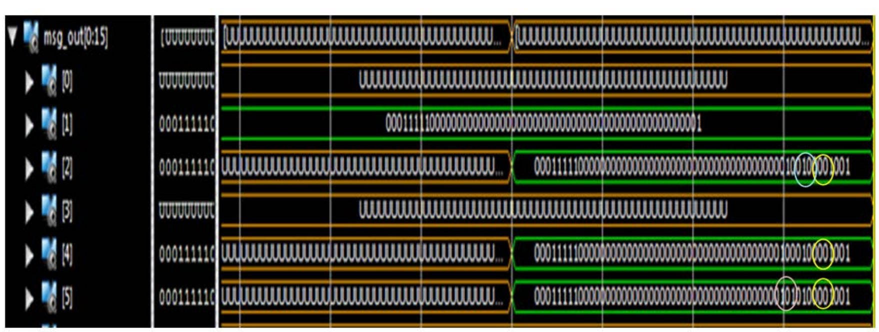

Fig. 9: Neighbours 2, 4 and 5 attach their respective addresses (White circle: Bits 13 downto 10) and associativity ticks (Light Blue circle: Bits 9 down to 6) and increment the hop count to 1 (Yellow circle: Bits 5 down to 3)

Table 2: Device utilization summary using Xilinx Virtex 5 FPGA

Device utilization summary

Slice logic utilization	Used	Available	Utilization
Number of slice registers	4,527	12,480	36
Number used as flip plops	3,455		
Number used as latches	1,072		
Number of Slice LUTs	2,843	12,480	22
Number used as logic	2,832	12,480	22
Number using O6 output only	2,817		
Number using O5 and O6	15		
Number used as exclusive route-thru	11		
Number of route-thrus	11		
Number using O6 output only	11		
Number of occupied slices	1,746	3,120	55
Number of LUT flip flop pairs used	5,800		
Number with an unused Flip Flop	1,273	5,800	21
Number with an unused LUT	2,957	5,800	50
Number of fully used LUT-FF pairs	1,570	5,800	27
Number of unique control sets	188		
Number of slice register sites lost to control set restrictions	105	12,480	1
Number of IOBs	69	172	40
Number of LOCed IOBs	48	69	69
IOB latches	58		
Number of BUFG/BUFGCTRLs	16	32	50
Number used as BUFGs	16		
Average fanout of non-clock nets	4.06		

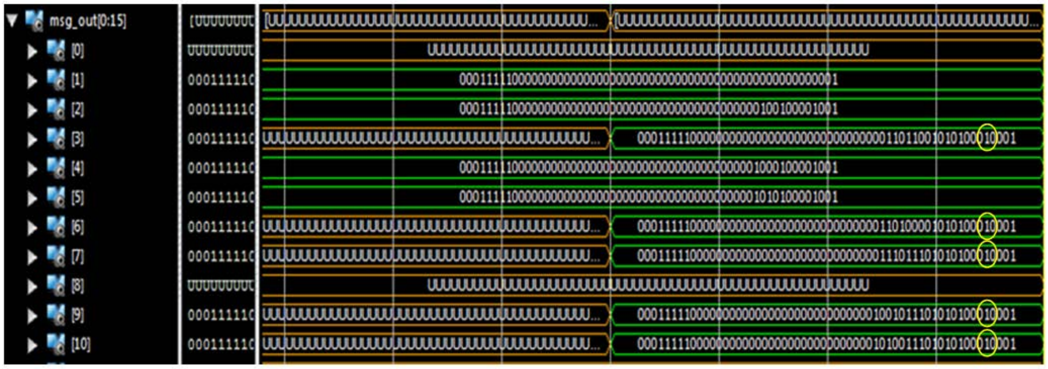

Fig. 10: Nodes 3, 6, 7, 9 and 10 receive BQ packets from various nodes and increment the hop count to 2 (Yellow circle: Bits 5 down to 3)

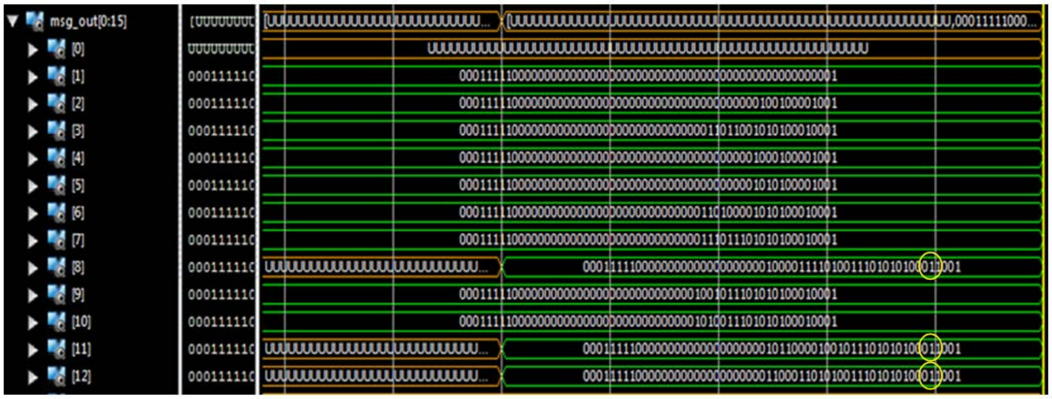

Fig. 11: Nodes 8, 11 and 12 receive the BQ packet and increment hop count to 3 (Yellow circle: Bits 5 down to 3)

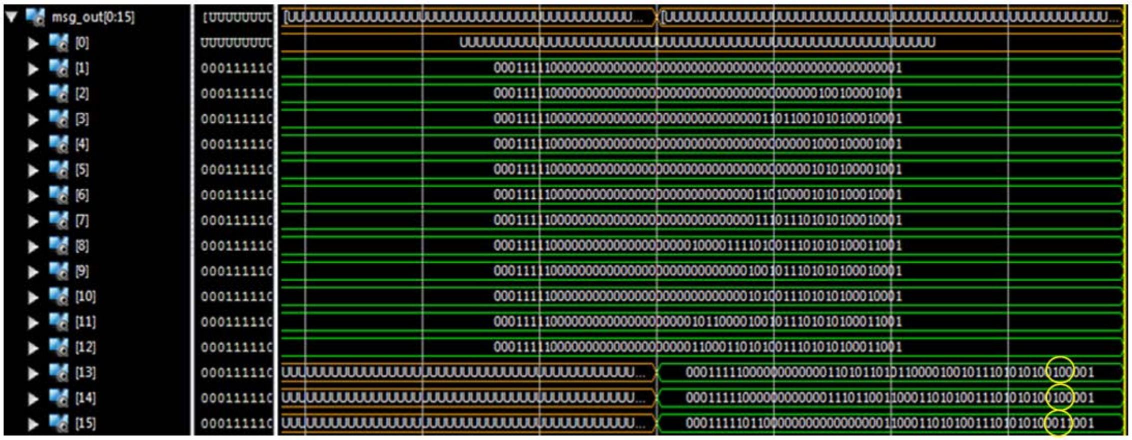

Fig. 12: Nodes 13, 14 receive the BQ packet and increment hop count to 4 (Yellow circle: Bits 5 down to 3). Destination node 15 receives the first BQ packet

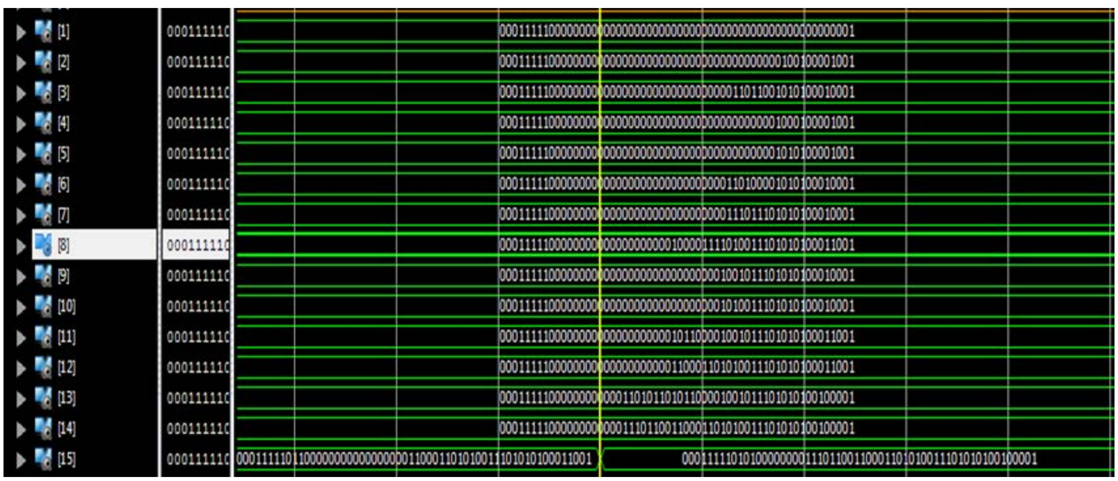

Fig. 13: Destination 15 receives BQ packet from different routes and stores them in its buffer. Highlighted section shows the transition at the destination

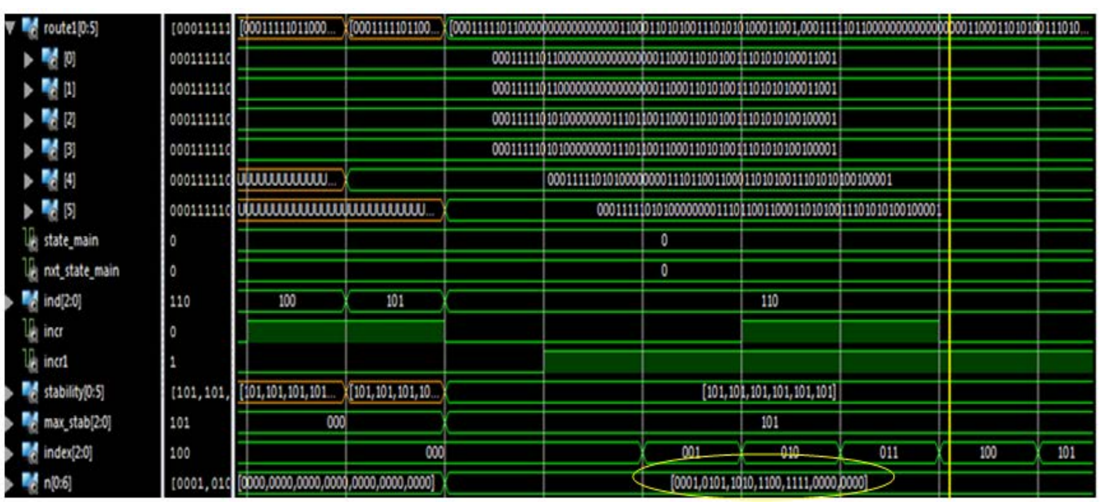

Fig. 14: The route is selected at the destination (1->5->10->12->15)

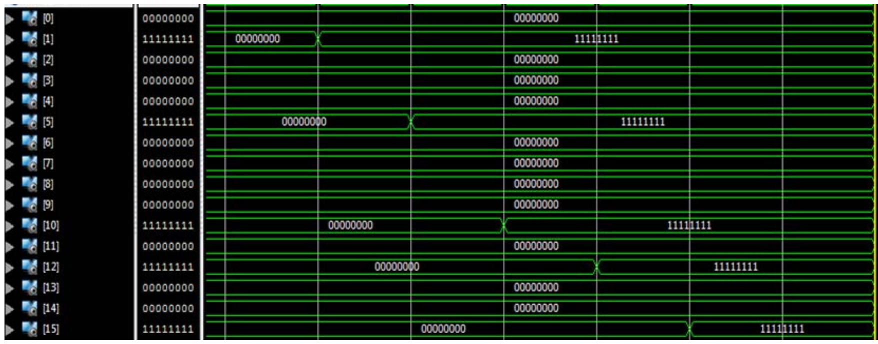

Fig. 15: Message (11111111) is transmitted along the selected path (1->5->10->12->15)

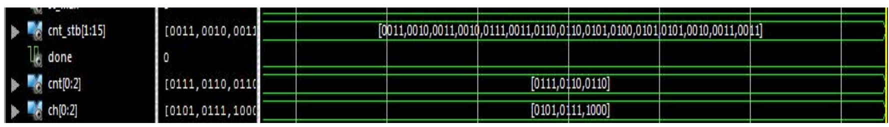

Fig. 16: Cluster head election

CONCLUSION

In ad hoc networks where battery power is of utmost importance, clustering helps in conserving energy. By rotating the role of the cluster head, no node is unfairly burdened.

Those nodes that are not directly involved in communication are put to sleep mode. A 14% saving in energy is achieved using cluster based approach in ABR. This increases considerably with increase in the density of nodes.

By making use of the best features of ABR and cluster formation, considerable energy saving is achieved without compromising network connectivity. The latency increased is also found to be marginal.

RECOMMENDATION

In future, we would like to apply the energy optimization frame work to a secure mobility model to gain insights into optimal mobility complementing energy conservation issues.

REFERENCES

Chen, B., K. Jamieson, H. Balakrishnan and R. Moriis, 2002. Span: An energy-efficient coordination algorithm for topology maintenance in Ad Hoc wireless networks. Wireless Net., 8: 481-494.

Kulkarni, S.A., G.R. Rao and N.I.E. Mysore, 2012. Performance optimization of reinforcement learning based routing algorithm applied to ad hoc networks. http://www.idrbt.ac.in/PDFs/shrirang-ambaji-kulkarni.pdf.

Kummakasikit, M., S. Thipchaksurat and R. Varakulsiripunth, 2005. Performance improvement of associativity-based routing protocol for mobile ad hoc networks. Proceedings of the 5th International Conference on Information, Communications and Signal, December 6-9, 2005, Bangkok, Thailand, pp: 16-20.

Meghanathan, N., 2007. Stability-energy consumption tradeoff among mobile ad hoc network routing protocols. Proceedings of the 3rd International Conference on Wireless and Mobile Communications, March 4-9, 2007, Guadeloupe, French Caribbean, pp: 9-9.

Safa, H., O. Mirza and H. Artail, 2008. A dynamic energy efficient clustering algorithm for MANETs. Proceedings of the IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, October 12-14, 2008, Avignon, France, pp: 51-56.

Sivavakeesar, S. and G. Pavlou, 2005. Associativity-based stable cluster formation in mobile ad hoc networks. Proceedings of the 2nd Conference on Consumer Communications and Networking, January 3-6, 2005, Las Vegas, Nevada, USA., pp: 196-201.

Toh, C.K., 1997. Associativity-based routing for ad hoc mobile networks. Wireless Personal Commun., 4: 103-139.