Performance Analysis of Utilizing Genetic Algorithm to Minimize Coding Nodes in Wireless Networks

Shee Eng Tan, Zhan Wei Siew, Scott Carr Ken Lye, Hou Pin Yoong, Kenneth Tze Kin Teo
Modelling, Simulation & Computing Laboratory, Material & Mineral Research Unit
School of Engineering and Information Technology
Universiti Malaysia Sabah
Kota Kinabalu, Malaysia
msclab@ums.edu.my, ktkteo@ieee.org

***Abstract*—Network Coding is a method that effectively combines several packets from different sources and broadcasts the combined packet to several destinations in single transmission time slot. Each destination is capable to extract the intended information by decoding from a common packet. In short, network coding is a method that improves the throughput for wireless networks and wired networks. However, while it increases the throughput of networks, it also causes side effects such as complexity of packets management and increased delay for coding opportunity. In this paper, genetic algorithm is proposed to optimize the resources for network coding. Genetic algorithm will search for suitable routes to the destination according to the desired throughput with a desired multicast rate. The simulation results show that the average end to end delay of network after minimizing coding nodes is less than normal network coding.**

***Index Terms*—Network coding, wireless networks, genetic algorithm, coding nodes, multicast.**

I. Introduction

Wireless technology is applied in many modern applications due to the mobility, robustness, and simplification of the networks distribution. Although some protocols in wireless technology have been developed based on custom applications related to the wireless sensor networks, most of the wireless networks still suffering from the throughput limitation [1]. To overcome this limitation, network coding is introduced to increase the network throughput [2, 3, 4]. For the future application network coding based methods are currently being explored to increase the performance of the networks in terms of coding opportunity in packet waiting time and coding aware routing [5, 6].

Network coding is a method that combines packets from different incoming flows to a single encoded packet and then transmits it in a single transmission time slot. Conventional wireless technology can only transport single content in a transmission but network coding offer multiple contents in a transmission, which lead to the increase in network throughput. Fig. 1 and Fig. 2 show the scenario of two nodes exchanging packet through a middle node. Fig. 1 illustrates conventional store and forward method while Fig. 2 shows network coding method. For Example, node 1 wants to send packet to node 3, but both nodes are out of wireless coverage range, therefore node 1 and node 3 need an intermediate node 2 at the center to relay the packet. Consider the case shown in Fig. 1, Node 1 will forward packet *a* to node 3 and node 3 will forward packet *b* to node 1. In conventional store and forward method, node 1 will send packet *a* to node 2, and then node 2 will forward to node 3. In total, this process requires two transmission time slots. Similarly, node 3 will send packet *b* to center relay node 2 and then forward to node 1. In overall, the exchange process needs four time slots to deliver packet *a* and packet *b* to their destinations. Fig. 2 shows how network coding delivers packets with fewer time slots than store and forward method. Node 1 will send packet a to node 2, now instead of directly forwarding packet a to node 3, node 2 will hold the packet for a short amount of time. Then node 3 will send packet *b* to center relay node 2. In this stage, instead of sending packet *a* and packet *b* separately like store and forward, center relay node 2 will combine both packet *a* and packet *b* into a packet with same size length of packet *a* or packet *b*. The common length depends on which packet has the larger size, packet a or packet *b*. Node 2 will then broadcast the coded packet to both node 1 and node 3, node 1 and node 3 will be able to get the native packet they want since they have another native packet to extract from the coded packet. In total, network coding only requires three transmission time slots to deliver packet *a* and packet *b* to the respective node 1 and node 3.

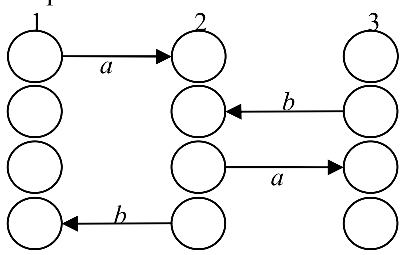

Fig. 1. Exchanging packet with bidirectional forwarding.

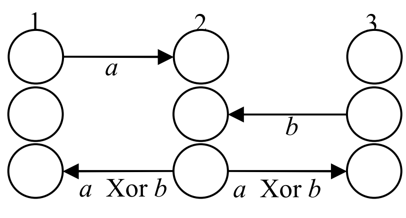

Fig. 2. Exchanging packet with network coding forwarding.

978-1-4673-4786-0/12/$31.00 ©2012 IEEE

Fig. 3 shows a butterfly network topology to illustrate how network coding can increase the throughput of a network. Node *s* is the source of the network while node 4 and node 5 are the sinks of the network. Node *s* wants to transmit packet *x* and packet *y* to node 4 and node 5, but node *s* cannot reach directly to node 4 and node 5, so these packets will be routed through several nodes before packets *x* and *y* reach node 4 and node 5. Fig. 3 (a) shows the store and forward method to transfer packet *x* and packet *y* to its destination. Before the node identifies how to route the packets to the destination, node *s* will broadcast a route request to the network and wait for route reply to decide the route to reach node 4 and node 5. In this case, link quality will affect the selected route. To overcome the problem, adaptive modulation can be introduced to make sure the link quality at every link is reasonably preserved [7]. Node *s* will send packet *x* to node 1 and then forward to node 4, same goes to packet *y*. For packet *x* and packet *y* to reach node 4 requires four transmission time slots to deliver the packets. Node *s* also uses another four transmission time slots to send packet *x* and packet *y* to node 5 as shown in Fig. 3 (a). Then the total transmission time slots required to deliver packet *x* and packet *y* to node 4 and node 5 are eight transmission time slots. Network coding scenario requires fewer time slots to deliver packet *x* and packet *y* to the same destination. In network coding scenario, the route to node 4 and node 5 are different from store and forward method as shown in Fig. 3. If the network follows the route of AODV routing protocol [8], every nodes do not have opportunity to perform network coding, where a different route is required to create coding opportunities. Node *s* sends the packet *x* to node 1 and packet *y* to node 2*,* this require two transmission time slots. Node 1 will broadcast packet *x* to node 3 and node 4, node 2 broadcast packet *y* to node 3 and node 5. At this stage, node 4 already has packet *x* and node *5* consists packet *y*. Before transmission take place, node 3 will then combine both packets to a single packet and then broadcast to both respective nodes. Node 4 will decode the coded packet using packet *x* to get packet *y*, node *5* doing the same process to get packet *x*. In overall, network coding requires only five time slots while store and forward method needs eight. As a conclusion, fewer time slots lead to higher throughput since network coding is able to deliver more information compare to the store and forward method within a same period of time.

Traditional network coding will choose the route which has the most coding opportunities as these will increase the throughput for the entire network. While network coding is increasing the network throughput, it also introducing complexity of packet management and requiring larger buffer size in order to perform packet encoding. Therefore, the route which has more coding opportunities does not necessary to be the most efficient way to deliver information.

Within a given topology, optimization problem in network coding means minimizing the disadvantages caused by the network coding and at the same time maximizing the throughput of the network. Kim has proven that optimization of coding resources is an NP-hard problem, and thus proposed simple genetic algorithm approach to deal with it [9, 10, 11].

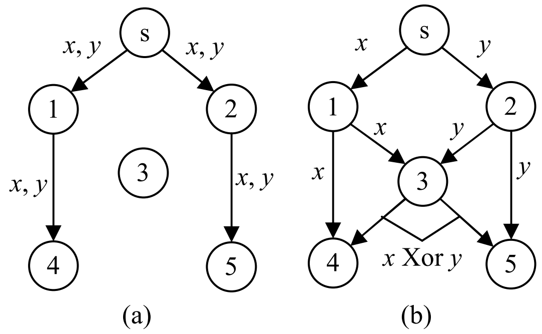

Fig. 3. Butterfly network topology: (a) Store and forward. (b) Network coding.

II. Genetic Algorithm

Genetic algorithm is a searching method that can potentially search a best solution for a given controlled scenario. The mechanism of Genetic algorithm is built based on Darwin's biological evolution process. Genetic algorithm is widely used in education and industry. Theoretical research of Genetic algorithm has been used to solve many optimization problems including the manufacturing industries of batch process control [12]. Fig. 4 shows the process flow chart of genetic algorithm. The mechanism starts with initial set of population that is located in a specific environment. The best species or best individual that can fit into the environment will survive longer and finally only the strongest chromosomes will remain.

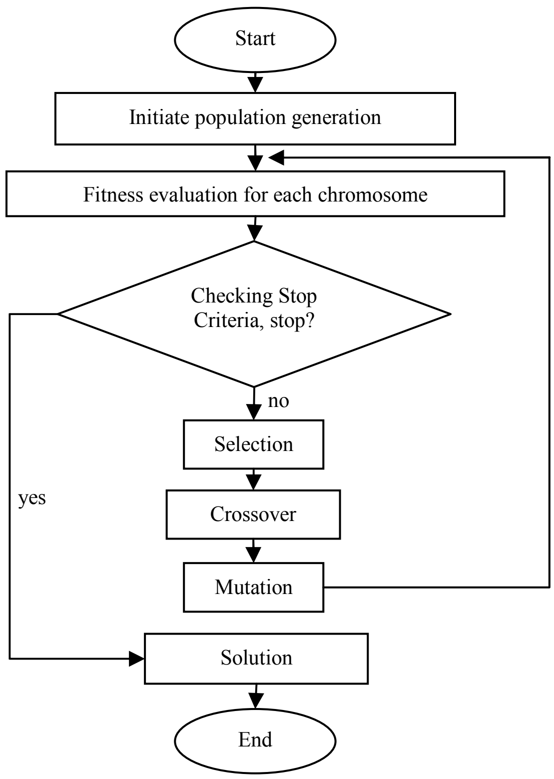

Fig. 4. Flow chart of genetic algorithm.

III. NETWORK CODING IN WIRELESS AD HOC NETWORKS

In this section, the development of network coding implemented in wireless networks will be described in detail. The simulation is programmed using Matlab m-file. Modification on standard 802.11 is based on the mechanism of the network coding. Sachin proposed COPE, a way of doing network coding in a practical wireless network [13]. The simulation of wireless network coding will be programmed according to some of the key theories in COPE. In COPE, the coding scheme principle of not delaying packets is applied. Conversely, it has been proven that waiting the incoming packets intelligently will lead to an increasing of opportunities of the nodes coding [14, 15]. An increment on coding opportunities also leads to increment in network throughput since coding will happen more often.

A. Packet management

The packet management scheme on each node is based on first in first out (FIFO) principle. When a node receives a packet, it will be appended to the bottom of a queue list as shown in Fig. 5. The first received packet will be placed at the top of the list which will have the priority to send first. With the store and forward method, the node will deliver packet *a* immediately once it has been received.

Queue management for network coding has an additional step before the first packet is sent. Before packet *a* proceeds to the physical layer in OSI model, the node will search through the entire queue list to look for another packet that can encode with the packet *a*. Let's say packet *c* qualify to encode with packet *a*, then packet *c* will be extracted from the queue list and broadcasted out together with packet *a* as shown in Fig. 5.

Fig. 5 shows that packet *a* will XOR with packet *c*. Due to the nature of XOR, encoded packet is the same size with packet *a* or packet *c*, depending on whether packet *a* and packet *c* which one has bigger size. In this simulation, the node will search the packet with almost the same size as the first packet because coding gain is not significant if coding occurs with a packet size that differ far from the first packet.

Transferring and exchanging data in network coding consist of bidirectional and cross exchange as shown in butterfly network. Intermediate node can check each native packet route to estimate what kind of packets owned by the neighbors. In bidirectional exchange data, the intermediate node will only check the route of the packets instead of guessing the packets' id at the neighbor nodes. The reason is because the information of every packets that go through the intermediate node will be stored.

For the case of cross exchange as shown in Fig. 6, it is necessary for an intermediate node to recognize the next hop has the required packet to decode it. Fig. 6 shows two different scenarios whether packet *j* can be received or not at node *5*. First, node 3 can XOR both packets in the scenario where packet *j* received at node 5 because both next hop nodes can decode them. However, in another case, node 3 cannot XOR both packets since next hop node 5 cannot decode to get native packet. Hence, the buffer information of a neighbor node can be known from receiving hello packet that used to check

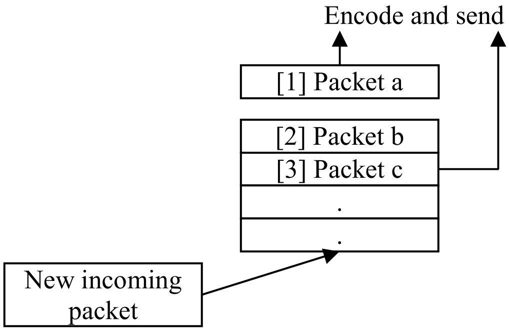

Fig. 5. Queue management in network coding.

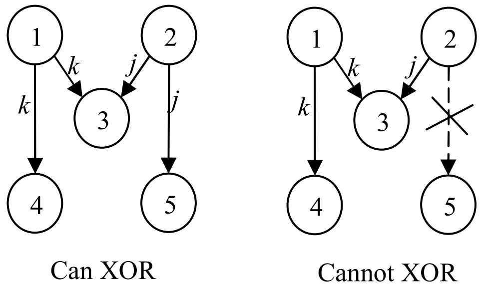

Fig. 6. Network coding scenario of 'can XOR' and 'Cannot XOR'.

the connectivity between two nodes, the packet id will be sent along with the hello packet. Another method to estimate the buffer of next hop is simply by guessing it. After a couple times on receiving the buffer packet of the neighbor, the node is able to predict the probability of the same type packet that will be received on the next hop.

Packets heading to the same next hop with the first packet will not be chosen to perform network coding since it is impossible to decode due to the next hop do not have the decoding packet. The source of the packet will never be considered as a coding packet since it has no other packets to combine with.

The packet that has been chosen to be encoded with first packet is extracted from the queue list where the packet will skip the queue. However, in this strategy the scenario will not add extra time slots or cause extra waiting time for the others.

B. Buffer

In network coding scheme, buffers are separated into two types, buffer that store queue list, and buffer to store any unintentional heard packets. Due to the broadcast nature, any nodes in the wireless coverage range of the transmitter can hear the neighboring packets sent no matter the packets are intended or not. Fig. 7 shows node 1 sending packet *a* to node 2, while other nodes surrounding the transmitter will notice the packet. In traditional network protocol, these packets will be dropped since they are not there for node 1. However, in network coding, these nodes will store the packet for a short amount of time instead of dropping the packet immediately. This is mainly because each packet stored has the potential to be used to decode a coding packet.

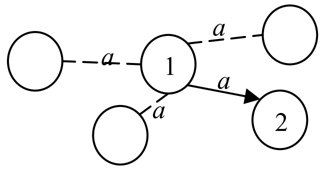

Fig. 7. Source sending a packet to neighbor.

IV. Network Resource Optimization

A. Chromosome Definition

Binary encoding in genetic algorithm is a simple representation because the straightforward computational complexity of binary encoding can be evaluated by human standard. Binary encoding also simplifies the computation processes such as crossover and mutation operation in genetic algorithm. The binary symbol in genetic algorithm is used to represent the individual's existence. But in network coding, the binary symbol represents the existence of the link of the node. Every node in a given topology with several incoming flow will become a potential network coding node. Let's say in a given topology there are i number of nodes that has the potential to become network coding node, then the chromosome will split into i segments to represent the link existence of each potential node. The packet will definitely flow through this link if it exists. Let's say a potential node i has j incoming flow and k outgoing flow, the total bit of the segment node i is shown in (1). The total bit needed for the segment i is g.

$$g_i = j \times k \tag{1}$$

Fig. 8 shows an example of a potential node V_2 with three incoming links and two outgoing links, since V_2 have three incoming links, g_2 have 3 bits to represent input link x_1, x_2, x_3 link to y_1 and another 3 bits to represent input link x_1, x_2, x_3 link to y_2, in total, g_2 have 6 bits. Referring to Fig. 3, a_1 = 000 represents link y_1 is not going to send any packet, a_2 = 101 denotes the node will perform network coding by combining x_1 and x_3, and send through link y_2. Other nodes in the network with only a single incoming flow will forward the incoming packets without delay. So chromosome only contains link state for node with several incoming flow or potential coding node information.

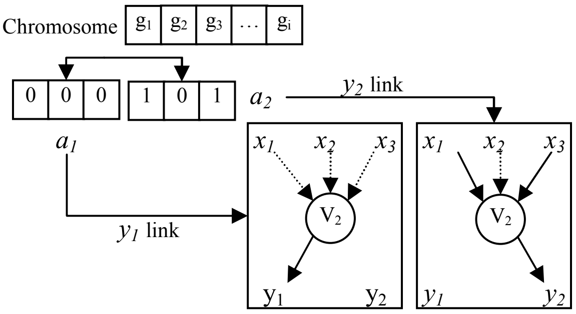

Fig. 8. Chromosome representative for node V with three incoming flow and two outgoing flow.

B. Fitness Function

Fitness function is very important in genetic algorithm as inappropriate fitness function will cause the solution set to diverge, thus, unable to obtain the most optimized solution. For the case of minimizing network coding resources, fitness function used in the simulation is shown in (3).

$$F(R) = \begin{cases} n_{node}, feasible = 1 \\ n_{\max} + 1, feasible = 0 \end{cases} \tag{3}$$

n_{max} represents the total number of all the possible coding nodes whereas n_{node} is the amount of coding nodes used in the network. Feasibility test will be conducted on the newly generated chromosomes, if the chromosome passed the test, n_{node} will be the fitness value for that combination of chromosome. However, if the test failed, then n_{max} + 1 will be the fitness value. Besides that, the chromosome should also meet the multicast scenario requirement. The chromosome which fails to do so will be eliminated. It can be observed that if all chromosomes are 1 then it implies a solution of the maximum possible coding node in the network, which means that all coding nodes are chosen, therefore n_{node} will never bigger than n_{max}.

V. Simulation and Performance Evaluation

In this section, the simulation setup of the wireless network and performance analysis of the network will be discussed in detail. The route for the source to the destination will be determined using genetic algorithm where the best solution will be tested in the simulation.

A. Simulation Setup

The simulation of wireless ad hoc network with network coding capability has been programmed in Matlab m-file and the protocol in the simulation follows 802.11 standards. However, this protocol is only limited to conventional store and forward method. To develop the network simulation network coding based, an additional modification on MAC layer and network layer is conducted accordingly to the mechanism that has been discussed on the previous sections.

In MAC layer, the decision to determine which packet in the queue list will be taken to perform coding with the first packet is implemented. Network layer defines whether packet needs to be decoded or not. This layer will also read the header in the packet to obtain which packet in the buffer will be used to decode the receive packet.

In this paper, fourteen nodes will be generated based on the topology shown in Fig. 9 and Fig. 10. Node s has prepared 100 packets send to node t_1, t_2, t_3, and t_4. The performance of the throughput for the entire network and end to end delay of each packet will be evaluated. The simulation will terminate when the last packet is delivered to its destination. The performance of network coding with minimization of coding nodes using genetic algorithm will be compared with the normal network coding.

Node 8, node 9, and node 10 will become potential network coding nodes. All these nodes have two outgoing links, so in total there will be 12 bit in chromosome to represent the packet flows. The crossover rate of the simulation in genetic algorithm is fixed at the rate of 0.7. The population size is set to 10 chromosomes, and the stopping criterion is either 80% of the total population becomes the common chromosome or reaches 100 generations. The mutation rates used in this paper are set to 0, 0.05, 0.1 and the final dynamic mutation rate.

B. Simulation Result

After using genetic algorithm to search an optimal route, only coding node 9 will be used, It can be observed that the link from node 4 to node 8 as well as node *7* to node *10* are save. It means communication failures between these two nodes are not important and hence packets missing between them can be ignored.

Table I shows the convergence results of genetic algorithm used in the topology shown in Fig. 9. Mutation rate of 0, 0.05, 0.1 and dynamic mutation rate are able to achieve the best minimized number of node which is 1. However, the average results obtained during mutation rate 0 is the worst although the convergence speed is the fastest among all. Therefore mutation rate 0 is not suitable in this scenario. Mutation rate 0.1 takes more generations to converge as compared to mutation rate 0.05, and dynamic 0 to 0.1 mutation rate where the solution obtains an average of 1.5. In average, dynamic mutation rate produces better solution than mutation rate 0.05, although both of the mutation rates converge at almost the same generations in average.

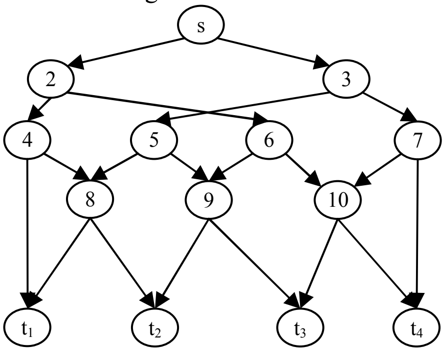

Fig. 9. Simulation topology of multiple nodes.

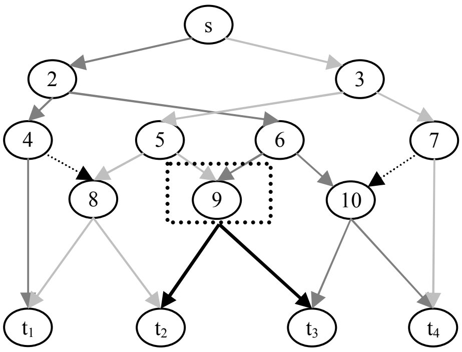

Fig. 10. Minimize coding nodes using genetic algorithm.

TABLE I. GENETIC ALGORITHM CONVERGENCE TABLE

Mutation rate			
	Minimal coding node	***Average result***	***Average Generation needs***
0	1	1.8	22.8
0.05	1	1.4	30.4
0.1	1	1.5	47.9
[0 0.1]	1	1.2	30.7

Fig. 11 shows the throughput measurement for the entire network, store and forward has the lowest throughput performance comparing to the minimize and non-minimize routes. However, packet delivery speed for minimize route is faster than store and forward, and non-minimize route.

Comparisons of end to end delay of each packet are presented in Fig. 12 and Fig. 13. End to end delay for minimize route are less than end to end delay of non-minimize route since minimize route simplified the traffic routing of the network.

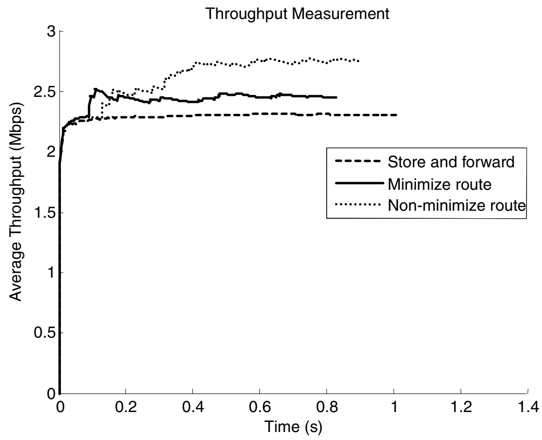

Fig. 11. Average throughput.

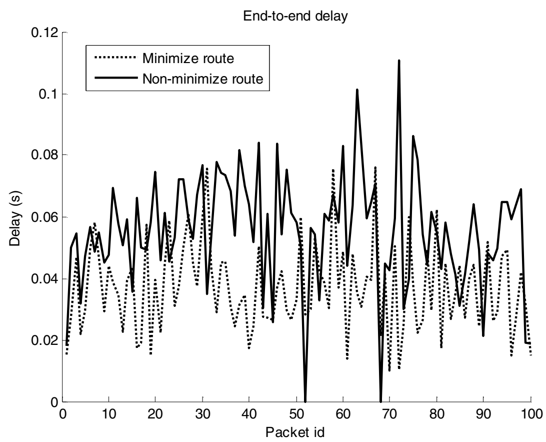

Fig. 12. Comparison of minimize route against non-minimize route.

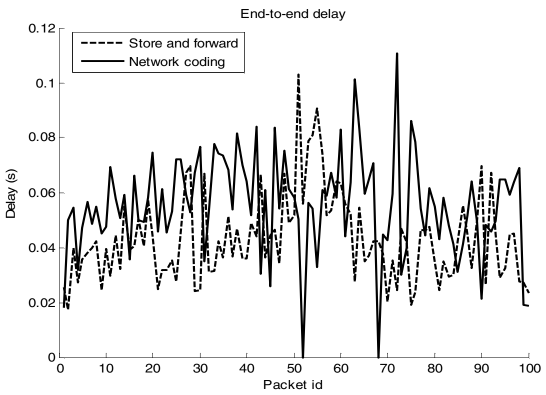

Fig. 13. Comparison of store and forward against network coding.

VI. CONCLUSION

Network coding is a proven method that increases network throughput. In a given network, the more coding node will increase overall network throughput. However, network coding also provides a shortfall of delay and network overhead cost. Therefore it is not necessary use all the coding nodes in a network. In this paper, genetic algorithm is used to minimize the usage of the coding nodes in the network in order to reduce the deficit of network coding while increasing the throughput. The simulation result shows that after minimizing the coding nodes, network coding with genetic algorithm provides less end to end delay and more throughput as compared to store and forward network. For the future work, genetic algorithm in wireless network with multiple sources and multiple destinations will be considered.

ACKNOWLEDGMENT

The authors would like to acknowledge the financial assistance of the Ministry of Higher Eduation of Malaysia (MOHE) under Exploratory Research Grant Schemes (ERGS), grant no. ERG0021-TK-1/2012, Fundamental Research Grant Schemes (FRGS), grant no. FRG0220-TK-1/2010 and scholarship support under MyMaster program.

REFERENCES

[1] Z.W. Siew, A. Kiring, H.T. Yew, P. Neelakantan and K.T.K. Teo, "Energy Efficient Clustering Algorithm in Wireless Sensor Networks using Fuzzy Logic," Proc. International Colloquium on Humanities, Science and Engineering Research (CHUSER), 2011, pp. 481-486, doi:10.1190/CHUSER.2011.6163758.

[2] R. Ahlswede, N. Cai, S.R. Li, and R.W. Yeung, "Network information flow," IEEE Transactions on Information Theory, 2000, vol. 46, no. 4, pp. 1204-1216, doi: 10.1109/18.850663.

[3] S.Y. Li, R. Yeung, and N. Cai, "Linear network coding," IEEE Transactions on Information Theory, 2003, vol. 49, no. 2, pp. 371-381, doi: 10.1109/TIT.2002.807285.

[4] R.W. Yeung, Information Theory and Network Coding. USA: Springer, 2008.

[5] S.E. Tan, H.T. Yew, M.S. Arifianto, I. Saad, K.T.K. Teo, "Queue Management for Network Coding in Ad Hoc Networks," Proc. 3rd International Conference on Intelligent Systems, Modelling and Simulation (ISMS), 2012, pp. 657-662, doi: 10.1109/ISMS.2012.113.

[6] B. Ni, N. Santhapuri, Z. Zhong, and S. Nelakuditi, "Routing with Opportunistically Coded Exchanges in Wireless Mesh Networks," 2nd IEEE Workshop on Wireless Mesh Networks 2006 (WiMesh 2006), 2006, pp. 157-159, doi: 10.1109/WIMESH.2006.288636.

[7] S.C.K. Lye, M.S. Arifianto, H.T. Yew, C.F. Liau, and K.T.K. Teo, "Performance of Signal-to-Noise Ratio Estimator with Adaptive Modulation," Proc. 6th Asia International Conference on Mathematical Modelling and Computer Simulation (AMS 2012), 2012, pp. 215-219, doi: 10.1109/AMS.2012.40.

[8] C.E Perkins, E.M Royer, "Ad-hoc on-demand distance vector routing," Proceedings of Mobile Computing Systems and Applications, 1999, doi: 10.1109/MCSA.1999.749281.

[9] M. Kim, M. Medard, V. Aggarwal, U.M. O'Reilly, W. Kim, C.W. Ahn, and M. Effros, "Evolutionary Approaches to Minimizing Network Coding Resources," Proc. 26th IEEE International Conference on Computer Communications (INFOCOM 2007), 2007, pp. 1991-1999, doi: 10.1109/INFCOM.2007.231.

[10] R. Koetter, and M. Medard, "An Algebraic Appraoch to Network Coding," IEEE/ACM Transactions on Networking, vol. 11, no. 5, 2003, pp. 782-795, doi: 10.1109/TNET.2003.818197.

[11] T. Ho, M. Medard, R. Koetter, D.R. Karger, M. Effros, J. Shi, and B. Leong, "A Random Linear Network Coding Approach to Multicast," IEEE Transactions on Information Theory, vol. 52, no. 10, 2006, pp. 4413-4430, doi: 10.1109/TIT.2006.881746.

[12] M.K. Tan, Y.K. Chin, H.J. Tham, and K.T.K. Teo, "Genetic Algorithm based PID Optimization in Batch Process Control," Proc. 2011 IEEE International Conference on Computer Applications and Industrial Electronics (ICCAIE 2011), 2011, pp. 162-167, doi: 10.1109/ICCAIE.2011.6162124.

[13] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, J. Crowcroft, "XORs in the air: pratical wireless network coding," IEEE/ACM Transaction on Network, 2008, vol 16, no. 3, pp. 497-510, doi: 10.1109/TNET.2008.923722.

[14] J. Zhang, and P. Fan, "Optimal Scheduling for Network Coding: Delay v.s. Efficiency," Proc. 2010 IEEE Global Telecommunications Conference (GLOBECOM 2010), 2010, pp. 1-5, doi: 10.1109/GLOCOM.2010.5683947.

[15] Y.E. Sagduyu, and A. Ephremides, "Some Optimization Trade-offs in Wireless Network Coding," Proc. 2006 40th Annual Conference on Information Sciences and Systems, 2006, pp. 6-11, doi: 10.1109/CISS.2006.286422.