Exploration of Genetic Algorithm in Network Coding for Wireless Sensor Networks

Kenneth Tze Kin Teo, Renee Ka Yin Chin, Shee Eng Tan, Chun Hoe Lee, and Kit Guan Lim
Modelling, Simulation & Computing Laboratory, Material & Mineral Research Unit
Faculty of Engineering, Universiti Malaysia Sabah
Kota Kinabalu, Malaysia
msclab@ums.edu.my, ktkteo@ieee.org

***Abstract*—Wireless network comprises of multiples nodes that work together to form a network. Each node in a wireless network communicates with one another by disseminating information packet among them. Source node and destination node are often far apart from each other, thus the information packet has to be transmitted to intermediate node(s) before it is able to be relayed to its destination. Network coding is introduced to combine several packets from different sources and broadcast 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 improves the throughput for wireless and wired networks but also causes side effects such as complexity of packets management and increases delay for coding opportunity. Hence, genetic algorithm is used to optimize the resources for network coding. Genetic algorithm will search for optimum routes to the destination according to the desired throughput with a desired multicast rate. In this paper, genetic algorithm is further enhanced in searching of optimum route for a packet. The simulation results show the enhanced genetic algorithm can adapt to various situations with different topologies with a better throughput and energy consumption compared to the store-and-forward method used in conventional wireless sensor network.**

Keywords- Network Coding, Wireless Networks, Genetic Algorithm, Coding Nodes, Multicast.

I. INTRODUCTION

Wireless network, formed by multiple wireless nodes, is widely used in industries nowadays due to its mobility, wide signal coverage, robustness and other features that simplify network distribution. Although some protocols in wireless technology have been developed based on custom applications related to wireless sensor networks, most of the wireless networks still suffering from throughput limitation [1]. To overcome this limitation, network coding is introduced to increase network throughput [2-4]. For 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].

In network coding, each node in the network combines multiple packets before forwarding them. In other words, network coding allows buffer to queue multiple packets in a transmission. While conventional wireless technologies transport only single content in a transmission, network coding offers multiple contents transmission, leading to the increase in network throughput, but this may also cause the delay on the packet to reach the destination [7].

Figure 1 and Figure 2 show the scenario of two nodes exchanging packet through an intermediate node, where Figure 1 illustrates conventional store and forward method while Figure 2 shows the transmission with network coding. Considering the scenario shown in Figure 1, node 1 has packet "A" to be delivered to node 3 and node 3 has a packet "B" to deliver to node 1, both going through intermediate node 2. Assuming the channel is ideal so that the packet delivery is assured to be successful. In other words, there will not be any disconnection or collision that might cause the packet to drop. Intermediate node 2 in Figure 1 uses store and forward method, the data is transmitted without any process on the packet. Such method requires 4 transmissions to deliver their respective packets to the destination. To be in detail, packet "A" delivered from node 1 to node 2, then node 2 to node 3, while packet "B" delivered from node 3 to node 2, before conveying from node 2 to node 1. When node 2 transmits the packet, it transmits to both the sender and the receiver.

Figure 2 shows the implementation of transmission with network coding. First, node 1 will deliver packet "A" to node 2, but note that node 2 will retain packet "A" temporarily. Instead of searching for the next hop to node 3 immediately after receiving packet "A" from node 1, node 2 waits for the arrival of packet "B" from node 3. Next, node 3 will deliver packet "B" to node 2 and similar to packet "A", packet "B" will be stored temporarily. However, both the packets are not stored for nothing, node 2 will combine packet "A" and "B" together using XOR binary operator, then broadcast packet "$A \oplus B$" to both node 1 and node 3. At this time, node 1 still has packet "A" and node 3 still has packet "B" respectively, so both node 1 and node 3 can obtain their desired packet by undergoing another XOR operator with the received packet. Therefore, network coding based forwarding approach requires only 3 transmissions to transfer packet "A" and "B", boosting the throughput by approximately 33.33% compared to traditional method.

DOI 10.5013/IJSSST.a.15.06.09

ISSN: 1473-804x online, 1473-8031 print

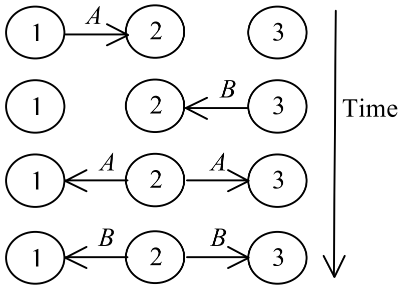

Figure 1. Exchanging packet with bidirectional forwarding.

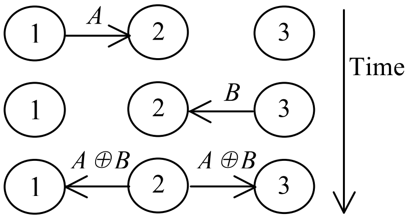

Figure 2. Exchanging packet with network coding forwarding.

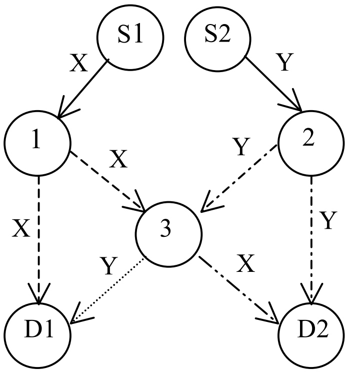

Figure 3. Butterfly network topology using store and forward.

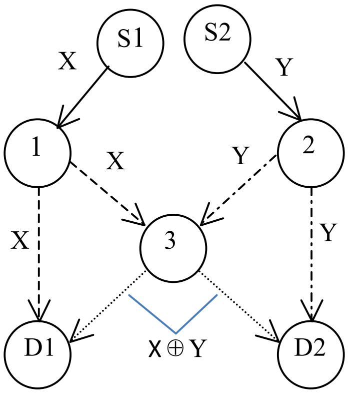

Figure 4. Butterfly network topology using network coding.

Figure 3 and Figure 4 show a butterfly network topology to illustrate uses of network coding in increasing the throughput of a network. Node S1 and S2 are the source of the network while node D1 and D2 are the destination of the network. Assuming that S1 and S2 are going to transmit packet "X" and "Y" to D1 and D2, but they are too far away hence cannot reach directly with each other, forcing the packets to be routed through several nodes before packet "X" and "Y" reach their destination. Figure 3 shows the store and forward method to transfer packet "X" and "Y" to its destination. Before the node identifies how to route the packets to the destination, S1 and S2 will broadcast a route request to the network and wait for route reply to decide on the suitable path to reach D1 and D2. In this case, link quality will affect the selected route. To overcome this problem, adaptive modulation can be introduced to make sure the link quality at every link is reasonably preserved [8]. S1 will send packet "X" to node 1 then forward to node 3 and D1, the same goes to packet "Y". For packet "X" and "Y" to reach D1 and D2, a total of 5 transmissions are needed, as shown by different dashes of line in Figure 3, where each type of dash represents one transmission. In network coding scenario, fewer time slots are required to deliver packet "X" and "Y". If the network follows the route of ad hoc on demand distance vector (AODV) routing protocol [9], every node does not have opportunity to perform network coding, where a different route is required to create coding opportunities. S1 sends the packet "X" to node 1 and S2 sends packet "Y" to node 2 at the same time. Next, node 1 will broadcast packet "X" to D1 and node 3, then node 2 broadcast packet "Y" to D2 and node 3. At this stage, three transmission time slots are required. Afterwards, two more transmissions are needed, which will be the transmission of packet "X" from node 3 to D2, and transmission of packet "Y" from node 3 to D1. On the other hand, Figure 4 shows after first three transmissions, packet "X" and "Y" will reach node 3, node 3 will then combine both packets to a single packet and broadcast to both nodes respectively. At node D1, the packet will be decoded using packet "X" to obtain packet "Y", the same process is carried out at node D2, decoding using packet "Y" to obtain packet "X". The whole transmission involving network coding takes 4 transmissions, which uses less transmission compare to store and forward method. Hence, it can be concluded that fewer time slots lead to higher throughput because network coding is able to deliver the same amount of information in a shorter time than store and forward method.

Conventional network coding chooses the route that has the most coding opportunities as this will increase the throughput for the entire network. However, with the increment in network throughput, network coding also causes a complexity of packet management and requires larger buffer size to perform packet encoding. The values of network coding tend to be offset by severe packet collision. This phenomenon happened because poor transmission coordination mechanism among nodes causes the collision of packet to happen more likely. This leads to a conclusion that a route with more coding opportunities might not be the most efficient way in delivering information throughout the network. To further optimize network coding in a given topology, the drawbacks caused by network coding have to be minimized. Kim has proven that optimization of coding resources is a Non-deterministic Polynomial-time hard (NPhard) problem, thus proposed simple genetic algorithm approach to deal with it, with the aim to find minimum unit-capacity link used in network coding [10-12].

II. Genetic Algorithm in Network Coding

Genetic algorithm is a searching method that can potentially search for optimal route for packet by considering the global traffic flow in the wireless ad hoc networks. It mimics the genetic phenomena such as gene reproduction, mutation and survival of the fittest. One of the design challenges in route construction is the complexity of finding the best route for large wireless ad hoc networks with network coding. The wireless node will search the route from the source to the destination based on the existing packets flow in the wireless ad hoc networks. Therefore, in order to have a simple and lightweight stochastic search algorithm, genetic algorithm is chosen in this work.

The mechanism of genetic algorithm is built based on Darwin's biological evolution process. Genetic algorithm works well in solving discrete and continuous problems. Theoretical research of genetic algorithm has been used to solve many optimization problems including the manufacturing industries of batch process control [13].

Fig. 5 shows the process flow chart of genetic algorithm, consisting of mutation, selection and crossover. The process of genetic algorithm starts with converting the problem into a set of chromosomes and placing this population in a desired environment. The process of evaluation, selection, crossover, and mutation will keep on repeating or iterating with the newly generated population to replace the existing solution. This process will iterate until a certain stopping criterion, which is either a certain percentage of equivalent chromosomes in the population or after a predefined number of generations is met.

Throughout the process, the best species or best individual that can fit into the environment will survive longer and finally only the strongest chromosomes will remain. Traditionally, genetic algorithm is formulated in binary as string of 0 and 1, this encoded method offers simple and fast computation for the processing unit.

A. *Chromosome Representation*

Binary encoding is a simple, fast and easy method in computational of the biological evolution process such as crossover and mutation operation because it is straightforward and can be performed by human calculation standards. The use of binary symbol is to represent the individual's existent. In optimization of network coding, the chromosome is used to represent a possible route for a packet to reach the destination. Every node in a given topology with more than one incoming flow will potentially become a network coding node, where each node has an identity (ID) to differentiate every independent node. The chromosome consists of sequences of node ID and each gene in the chromosome is corresponding to a node in the networks. For example, if there are I nodes in a topology that has network coding node potential, a chromosome will split into I segments to represent the link state of each potential node. Consider a potential node I has j incoming flow and k outgoing flow, the total bit of segment I is shown in Eq. (1).

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

Assuming i potential coding is presented in a given topology, V_i represents the i^{th} potential coding node, $V = \{V_1, V_2, V_3 \cdots V_i\}$. The total bits needed for a complete chromosome is shown in Eq. (2).

$$n = \sum_{V_1}^{V_i} g_i \tag{2}$$

Figure 6 shows an example of potential node V_2 with three incoming links and two outgoing links, since V_2 has three incoming links, g_2 has three bits to represent input link x_1, x_2, x_3 to link y_1 and another three bits to represent input link x_1, x_2, x_3 to link y_2, in total g_2 has six bits. Example as in Fig. 6, a_1 = 000 represents link y_1 is not going to send any packet, a_2 = 101 means the node will perform network coding by combining x_1 and x_3 and send through link y_2. Other nodes at the network with single incoming flow will only forward the packet, so chromosome only contains link state for node with several incoming flow or potential coding state.

B. *Fitness Function*

Fitness function is used to evaluate the fitness of each chromosome value in population and least fit chromosomes will be eliminated. It is very important in genetic algorithm, as unsuitable fitness function will lead the solution to another direction which will make it difficult for genetic algorithm to obtain solution. In order to minimize resource of network coding, fitness function is formulated as in (3).

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

where n_{max} represents the quantity of all possible coding node in a given network, n_{node} is the quantity of coding node in chromosome. Feasible test is carried out on the newly generated chromosome, if the test passed, n_{node} will be the fitness value for that combination of chromosome. On the other hand, if it failed then n_{max} + 1 will be the fitness value for it. Besides, the chromosome should meet the multicast requirement. Chromosome that fails to do so will be eliminated. As indicated in Eq. (3), 1 in the chromosome refers to the maximum coding node in the network, which means all coding nodes are chosen, therefore n_{max} will never bigger than n_{node}. In short, feasible test checks the validity of the solution.

III. Genetic Algorithm

In genetic algorithm, implementing fixed parameters will cause failure to obtain the best path and potentially being trapped at local optima. A better solution could be obtained by dynamically changing the parameters such as selection probability of next hop and the mutation rate in genetic algorithm.

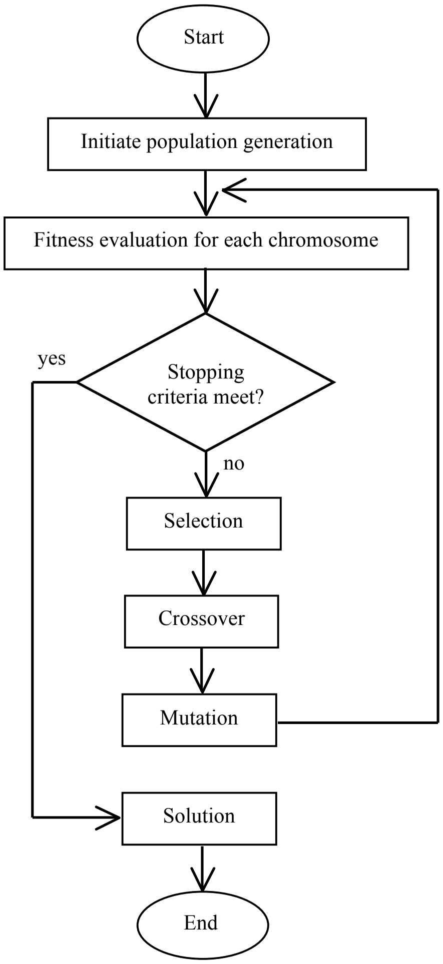

Figure 5. Flow chart of genetic algorithm.

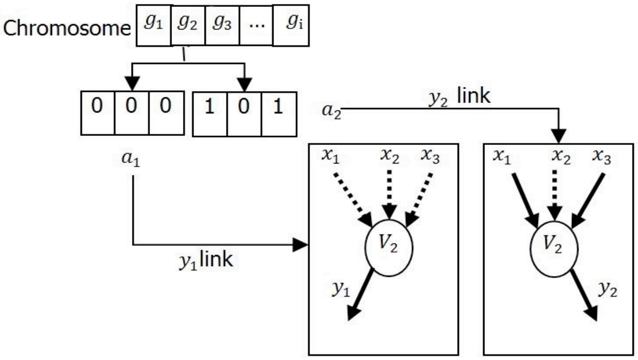

Figure 6. Chromosome representative for node *V* with 3 incoming flows and 2 outgoing flows.

The probability of each link in the nodes is assigned using genetic operator's selection method, namely ranking selection to assign a probability value according to the angle difference of the destination node and the next node. When the fitness value of the random chromosomes is calculated, the selection probability of next hop varies based on the quantity of coding opportunities detected in the route. The better coding opportunities can be discovered by changing the selection probability of the link which is based on the searching while performing genetic algorithm in network coding. If any route from chromosome passes through the coding path, the selection probability of the link will increase.

By utilizing the improvement on selection probability, the gene in the chromosome is filled with the angle difference between destination node and next hop node. Thus the improvement on choosing intensity for link selection is influenced by mutation. When mutation takes place, genetic algorithm will generate a random number to choose the next hop. Mutation is the only operator that is involved in choosing the next hop based on the selection probability of each link. In genetic algorithm based network coding, the similarity between chromosomes will gradually increase at the later stage while the exploration ability is gradually decreased at the same time.

In this paper, dynamic mutation rate is proposed based on the solution on the population. Dynamic mutation rate can increase the exploration ability of genetic algorithm based network coding. Mutation rate in genetic algorithm offers better chances for solution to move out from the local optima, so that genetic algorithm is able to explore more potential solution besides local solution. However, if the mutation rate is maintained at the high value, the population is difficult to converge because of the exploration ability. Therefore, mutation rate must be dynamically changed according to the population condition. The equation used for mutation rate is shown in Eq. (4).

$$m_d = \begin{cases} m_d + \Delta BS \times \dfrac{1}{G_{\max}} & , \ m_d \le m_{\max} \\ m_{\max} & , \ otherwise \end{cases} \tag{4}$$

In Eq. (4), m_d is the mutation rate of the genetic algorithm, G_{max} is the maximum generation size, and ΔBS is the number for the similar solution that continuously appeared from generation to generation. ΔBS will increase linearly when the best solution in current generation is similar with previous generation and reset to zero every time the fittest solution from previous generation is not the same with current generation. The increment in mutation rate also increases the ability of the solution to move out from local optima. A maximum mutation rate, m_{max} is set by using Eq. (5).

$$m_{\max} = \frac{G_{\max} - G_k}{G_{\max}} \tag{5}$$

where G_k refers to current generation with $k = \{1, 2, 3 \cdots G_{max}\}$ and m_{max} decreased linearly from generation to generation. In the earlier stage, exploration is more important than exploitation, thus the maximum mutation rate can be large to provide more exploration. However, when the generation is approaching the maximum generation set in genetic algorithm, the exploitation is more important, causing mutation rate to decrease.

To further enhance the genetic algorithm, regenerate chromosome is introduced to reduce the tendency of the chromosome from being trapped at the local optima. Regenerate chromosome offers the exploration ability to genetic algorithm with improvement of the searching ability. Chromosome is regenerated by selecting some of the chromosomes in the population and replacing them with a new set of chromosomes. The regeneration quantity in every generation is calculated based on the equation in Eq. (6).

$$R = m_{\max} \times R_{\max} \tag{6}$$

where R_{max} is the predefined quantity of regeneration in genetic algorithm and R is the calculated regenerate quantity of each generation. However, regenerate mechanism need to be triggered by ΔBS value. The trigger value for regenerate mechanism is calculated based on Eq. (7).

$$T = Ceil\left(1 + \frac{10G_k}{G_{\max}}\right) \tag{7}$$

Trigger value T increased at every 10% of the generation. In this work, when ΔBS value is equal or more than T, chromosome in population will be regenerated based on Eq. (6). The fittest chromosome is in the first position on the population due to the elitism strategy, hence it should be excluded. R number of chromosome will be randomly selected from the population and a new set of string will be generated for the selected chromosome.

IV. Simulation and Results

In this section, the simulation of wireless network and performance analysis of the network will be discussed. 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 coding based simulation, an additional modification on MAC layer and network layer is needed.

In MAC layer, the decision to determine which packet in the queue list will be considered to perform with the first packet. Modification is made on the packets queue in the buffer so that every node can encode the packets that are ready to be sent. Nodes can decide which packets should be combined together so that the next node can decode the packet to obtain the original packet.

Network layer defines the necessity of the packet to be decoded, at the same time reads the header in the packet to obtain the packet in buffer which will be used to decode the received packet. The evaluations on the developed genetic algorithm based coding-aware routing (GACAR), network coding based wireless network transmission and forwarding structure (COPE), and store-and-forward will be carried out with the simulation run under various network topologies. In COPE, the coding scheme principle of not delaying packets is applied. It has been proven that waiting the incoming packets intelligently will lead to an increasing of opportunities of the nodes coding [14-15]. Table I shows the general parameter settings for GACAR.

The evaluation of store-and-forward, COPE and GACAR in wireless ad hoc networks is further carried out. The performance metrics used to evaluate these simulations are the throughput of the network, the total energy consumption to deliver every packet to destination and the expected transmission count metric (ETX) value for every packet. Figure 7 shows the block diagram of how the comparison is made.

Table II shows the basic parameter settings of wireless ad hoc networks in the simulation. 100 wireless nodes are randomly deployed at 200 m × 200 m area. The transmission range of every wireless node in the networks is assumed to be 50 m. The parameter settings used in simulating GACAR are shown in Table III. The wireless environment is assumed to be ideal and there is no error rate between the transmission nodes.

The routes for simulation are calculated under different cases to investigate the performance of the developed methods. Eight different random topologies which are states 1, 7, 11, 17, 21, 27, 31 and 37 will be used to measure the performances of different protocol instead of using a single predefined topology. The random topologies are created using 'state' function in MATLAB to ensure the random number generated is the same set throughout the simulation, ensuring fair comparisons.

TABLE I. Parameters for Genetic ALgorithm based Coding Aware Routing

Parameter	Value
Best selection pressure, η^{+}	1.4
Worst selection pressure, η^{-}	0.6
Maximum selection pressure, $\rho_{\max}$	0.5
Maximum regeneration quantity, $R_{\max}$	20% of population
Stopping criterion	80% of population
Population size	20
Generation	100

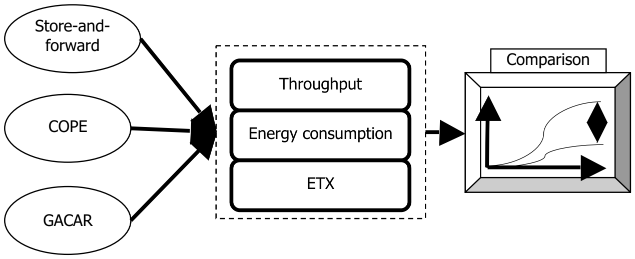

Figure 7. Diagram of methods comparison.

TABLE II. PARAMETERS FOR WIRELESS AD HOC NETWORKS

Parameter	Value
Network size	200 m × 200 m
Number of wireless node	100
Transmission range	50 m
Packet size	10000 bits
Transmission Rate	5 MHz

TABLE III. PARAMETERS USED FOR SIMULATING GENETIC ALGORITHM BASED CODING AWARE ROUTING

Parameter	Value
Population size	100
Maximum generations	500
Tournament size	5% of population
Crossover rate	0.2

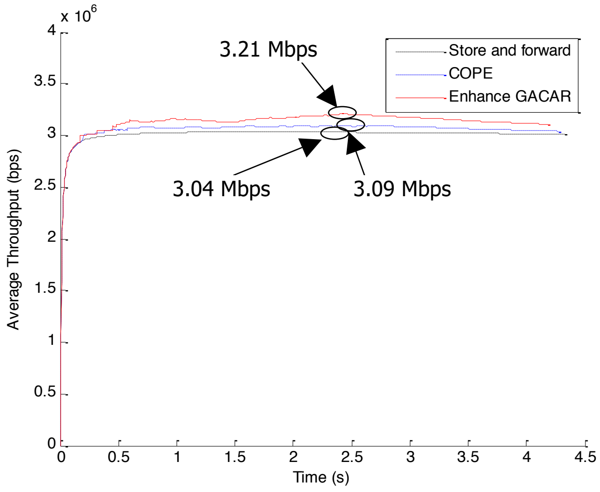

Figure 8. Throughputs Comparison

TABLE IV. AVERAGE THROUGHPUT OF VARIOUS TOPOLOGIES UNDER DIFFERENT STATES

Cases (State)	Average Throughput (Mbps)		
	Store-and-forward	COPE	GACAR
1	3.006	3.04	3.108
7	3.012	3.043	3.121
11	2.998	2.998	3.022
17	3.005	3.009	3.053
21	3.009	3.009	3.054
27	3.004	3.004	3.052
31	2.983	2.983	3.064
37	3.005	3.014	3.177

B. Simulation Results

The total time to deliver all packets to destination for store-and-forward method takes longer time than COPE and GACAR because store-and-forward needs more transmission to deliver the packets to the destinations. However, the shortest path may not be the route that provides the most coding opportunities.

From the simulation results shown in Figure 8, COPE and GACAR are better compared to the store-and-forward protocol. The average throughputs for the entire network in store-and-forward protocol, COPE and GACAR are 3.01 Mbps, 3.04 Mbps and 3.11 Mbps respectively. The peak of the average throughput is approximately 2.5 seconds with GACAR at 3.21 Mbps, COPE at 3.09 Mbps and store-and-forward at 3.04 Mbps. Throughput of the network is the highest at about 2.5 seconds because the packets have arrived at the centre of the sources and destinations. Traffic at that moment is busy and transmission between nodes is continuous. In the continuous transmission, idle time between current packet and the next packet is smaller, thus the throughput of the networks is increased. An increased in throughput also means that the selected route provides more coding opportunities to produce more throughput gain. The introduction of GACAR will automatically adjust the mutation rate, selection pressure on each link and introduces new set of chromosome based on the generation of GACAR to control the exploration and exploitation for each generation. Table IV shows the average throughput of various topologies under different states. GACAR is better than store-and-forward and COPE with the average improvement of 0.07863 Mbps and 0.06888 Mbps respectively in the average throughput. From Table V, energy consumption in delivering packets to destination for GACAR is also relatively less than store-and-forward and COPE. Compared to store-and-forward, GACAR consumed 0.00888 J less energy, while COPE needs 0.00375 J more energy than GACAR on average. Hence GACAR is proved to be able to provide better throughput improvement and also lower energy consumption. Table VI and VII show the ETX for store-and-forward, COPE and GACAR.

ETX metric is an advanced routing metric to search for high throughput paths in wireless network. It is the number of expected transmission packet to be received by the destination without error. From Table VI and Table VII, results show GACAR is better than COPE with the average improvement of 16.33%. On the other hand, an average improvement of 22.27% is shown when GACAR is compared to store-and-forward. It can be noticed that GACAR is a better route selector than store-and forward and COPE.

V. CONCLUSION

In wireless ad hoc network with network coding, shortest path is not the absolute best route as it does not consider coding opportunities that could surpass the shortest path determined through the conventional store-and-forward method. Results showed that GACAR achieved an average improvement of 22.3% over store-and-forward and 16.33%

TABLE V. ENERGY CONSUMPTION OF VARIOUS TOPOLOGIES UNDER DIFFERENT STATES

Cases (State)	**Energy Consumption (J)**		
	Store-and-forward	**COPE**	**GACAR**
1	1.975	1.960	1.937
7	2.224	2.206	2.170
11	1.744	1.744	1.728
17	1.859	1.855	1.840
21	2.278	2.278	2.325
27	2.064	2.064	2.040
31	1.536	1.536	1.511
37	2.133	2.129	2.191

TABLE VI. ETX FOR STORE-AND-FORWARD, COPE, AND GACAR UNDER VARIOUS NETWORK TOPOLOGIES

Cases (State)	**ETX Value**		
	Store-and-forward	**COPE**	**GACAR**
1	26	22	20
7	30	24	20
11	23	23	22
17	25	24	21
21	31	28	25
27	28	28	22
31	20	20	15
37	28	27	19

TABLE VII. IMPROVEMENT OF ROUTE SELECTED UNDER VARIOUS NETWORK TOPOLOGIES

Methods	**Average Improvement of ETX**	
	Compare with Store-and-forward	**Compare with COPE**
COPE	+ 7.11 %	-
GACAR	+ 22.27 %	+ 16.33 %

over COPE. It can be concluded that GACAR provides better route in the networks, increasing coding opportunities while improving the throughput, at the same time reducing the energy consumption in various defined network topologies.

ACKNOWLEDGMENT

The authors would like to acknowledge the financial support from the Ministry of Education Malaysia (KPM) under Exploratory Research Grant Schemes (ERGS), grant no. UMS/ERG0046-ICT-1/2013, and Fundamental Research Grant Schemes (FRGS), grant no. UMS/FRG0365-ICT-1/2014.

REFERENCES

[1] Z.W. Siew, C.H. Wong, C.S. Chin, A. Kiring, and K.T.K. Teo, “Cluster Heads Distribution of Wireless Sensor Networks via Adaptive Particle Swarm Optimization,” Proc. 4th International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN), 2012, pp. 78-83, doi: 10.1109/CICSyN.2012.25.

[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] K-H. Alireza, and H.R. Rudolf, “Bounds on the Benefit of Network Coding for Wireless Multicast and Unicast,” IEEE Transactions on Mobile Computing, 2014, vol. 13, no. 1, pp. 102-115, doi: 10.1109/TMC.2012.234.

[4] T-H. Kim, H. Choi and H-S Park, “Centrality-Based Network Coding Node Selection Mechanism for Improving Network Throughput,” IEEE Transactions on Advanced Communication Technology (ICACT), 2014, pp. 864-867, doi: 10.1109/ICACT.2014.6779083.

[5] S.E. Tan, Z.W. Siew, Y.K. Chin, S.C.K. Lye, and K.T.K. Teo, “Minimizing Network Coding Nodes in Multicast Tree Construction via Genetic Algorithm,” Proc. 4th International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN), 2012, pp. 399-404, doi: 10.1109/CICSyN.2012.79.

[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, pp. 157-159, doi: 10.1109/WIMESH.2006.288636.

[7] S. Sengupta, S. Rayanchu, S. Banerjee, “An Analysis of Wireless Network Coding for Unicast Sessions: The Case for Coding-Aware Routing," Proc. 26th IEEE International Conference on Computer Communications (INFOCOM), 2007, pp.1028-1036, doi:10.1109/INFCOM.2007.124.

[8] 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, pp. 215-219, doi: 10.1109/AMS.2012.40.

[9] 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.

[10] 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, pp. 1991-1999, doi: 10.1109/INFCOM.2007.231.

[11] 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.

[12] 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.

[13] M.K. Tan, H.S.E. Chuo, H.J. Tham, and K.T.K. Teo, “Exothermic Batch Process Optimisation via Multivariable Genetic Algorithm,” Proc. International Conference on Advanced Computer Science Applications and Technologies (ACSAT), 2012, pp. 43-48, doi: 10.1109/ACSAT.2012.19.

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

[15] 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.