J. Basic. Appl. Sci. Res., 3(7)117-126, 2013
ISSN 2090-4304
Journal of Basic and Applied Scientific Research
www.textroad.com

Intelligent Routing for Opportunistic Networks Based on Distributed Learning Automata

Chamran Asgari*[1], Ahmad Zareie[2], Rohollah Rahmati Torkashvand[3]

[1]Department of Computer Engineering, Payame Noor University, Iran
[2]Department of Computer Engineering, songhor branch, Islamic Azad University songhor, Kermanshah, Iran
[3]Department of Computer Engineering, Borujerd branch, Islamic Azad University Borujerd, Borujerd, Iran

ABSTRACT

Opportunistic Networks are indeed typical MANET networks. There are mobile nodes in these networks; moreover, there is no available information about network topology; however, network topology is highly dynamic. This type of networks is highly discrete. There may be no direct connection - or even multi-step connection - between source and destination nodes when a message is sent to destination node from source node. In these networks, although there is no connection route between destination and source, network should be able to transmit messages from source to destination. In fact, routing is done in a dynamic way. In this article, we present a novel approach for intelligent routing and transmitting messages from sources to various destinations based on distributed learning automata. This approach aims to find the best carriers which transmit messages. A good carrier should considerably reduce consumption resources, messages duplication and communication latency. Simulation results by NS2 software and comparing them with HIBOP andEPIDEMIC algorithms implied that this approach is more efficient than other presented ones.
KEYWORDS: Distributed Learning Automata, Message propagation, Opportunistic Networks, intelligent Routing.

1.0 INTRODUCTION AND RELATED WORKS

In opportunistic networks, there are mobile nodes and their topology is severely unstable and sometimes it cannot be predicted at all. However, despite above-mentioned issues, opportunistic networks should be able to transmit messages from various sources to different destinations. In these networks, establishing a connection between source and destination is not easily feasible. These networks are highly discrete. Indeed, this fragmentation can be resolved using mobile nodes. These mobile nodes are used as connecting bridges between these two discrete areas. In this article, we deal with two main issues. The first issue is routing and the second issue is transmitting messages. It should be noted that there is not always a specific route between source and destination during transmitting a message from source to destination. In presented approach, we attempt to specify messages' transmitting routes in a dynamic and intelligent way. In other words, next message is sent to next node during finding the next step. Various approaches are presented for routing. Some routing approaches are based on data dissemination in all directions in the network. It might be probably possible that messages will finally reach destination by messages dissemination through network; however, these approaches are associated with high resources consumption. Thus, in this article, we tend to specify several numbers of nodes which transmit messages to destination. Moreover, we attempt to reduce infected nodes in order to reduce traffic caused by data dissemination. Meanwhile, latency issue should be considered here too. This approach aims to find the best carriers for transmitting data using learning automata capability. A good carrier should appropriately reduce resources consumption, messages duplication and communication latency.

In this article, we attempt to reduce network fragmentation problem using some mobile nodes and finally deliver messages from various sources to different destinations. Opportunistic Networks are mainly divided into two categories [1]: networks without infrastructures and networks based on infrastructures. In networks without infrastructures, all nodes has the equal right for routing as well as transmitting message; however, in networks based on infrastructures, specific nodes are used to deliver messages opportunistically to destination – we deal with these kinds of networks, i.e. networks without infrastructures, in this article. In opportunistic networks, messages can be saved in buffers using local buffer for all nodes. Then, message can be sent to other nodes when a connection is established with other

Corresponding Author: Chamran Asgari, Department of Computer Engineering, Payame Noor University, Iran.
E-mail: asgari.chamran@gmail.com

nodes. In an opportunistic network, nodes can be categorized as discrete ones in which nodes related to a group have a direct and multi hop connection with one another; however, groups' connection with each other is established in a specific mode. In fact, data can be exchanged between two groups when mobile nodes are placed within radio range of two groups. Some works has been done regarding routing and message transmitting from one node to another node in opportunistic networks; however, all previously presented works caused typically high consumption of resources. In flooding-based approach, each node transmits the message to its neighbors as soon as receiving it. For this reason, large amount of resources are consumed and lost during transmitting received message to neighbors. Epidemic routing [2] ensures messages delivery to destination. In Epidemic routing, a node in infected either by the message the node itself has created or by the message it has received from other nodes. It saves the message in local buffer. An ACK is disseminated from destination node after message is delivered to destination and cleans infected nodes – which were infected by that message -, i.e. it does not receive and disseminate that message anymore. It should be noted that dissemination is limited in this routing because maximum steps is allocated to it when a message is generated. In Epidemic routing, each node also transmit messages to all other nodes. For this reason, resources are largely consumed and lost. Although several authors attempt to reduce resources consumption in Epidemic routing [3-7], this issue is still one of the important problems in opportunistic networks. In [5]estimated message delivery probability using encountering and transmission history.In routing based on network-coding [5], messages are combined before transmitting, i.e. messages are encrypted. In MV routing protocol [6], nodes meeting is occurred in a geographic situation according to recent history. Routing based on content [8], according to nodes content information, verifies subsequent steps for transmitting data to possible destination. Routing based on content can dramatically reduce messages dissemination and resources consumption; moreover, it can reduce number of lost messages and delays. In routing based on content, nodes content data should be maintained in nodes and disseminated through nodes. In context-aware routing [9], each node verifies the best carrier based on nodes content (text). In presented routing in [10] complete information on destinations should be available, so that messages can be sent to destinations from sources. In routing based on history (HIBOP) [11], delivery probability is estimated based on correspondence between context information on saved destination in the message itself and context information which is saved by encountered node. HIBOP verifies appropriate carriers for messages based on social relationship between users. The rest of the paper is organized as follows. The next section describes the learning automata, distributed learning automata and variable action-set learning automata. In Section 3, the proposed algorithm is presented. The performance of the proposed algorithm is evaluated through the simulation experiments and comparison in Section 4 and Section 5 concludes the paper.

2.0 LEARNING AUTOMATA

Learning automata is a machine which is able to do limited number of actions. When an action is selected from a set of learning automata actions, the selected action is evaluated by a probable environment and the result is sent as a signal to learning automata. Then, learning automata is affected by environment response in the next action. The final goal is that automata learn to select the best action among a limited set of actions. Learning automata performance interaction with environment is presented in following figure [12].

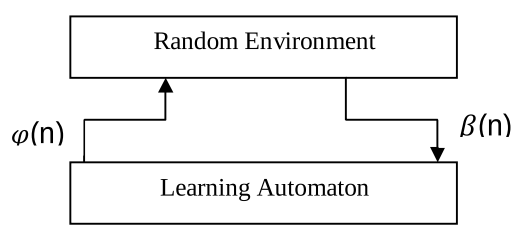

Figure1.The relationship between the learning automata and its random environment

Reward message is as follows: when A_i node receives reward message, probability vector updates its action by rewarding selected actionφ_i^j. Moreover,

$$P_{ij}(n+1) = P_{ij}(n) + \vartheta\left[1 - P_{ij}(n)\right] \qquad (1)$$

WhereP_{ij} is the probability of which A_i node select V_i vertex as the next vertex and fines it's other actions(φ_i^j $for\ all\ x \neq j$), that is to say:

$$P_{ix}(n + 1) = (1 - \vartheta)P_{ix}(n) \qquad (2)$$

Whereϑ is reward coefficient. After rewarding selected action, action probability vector should be reupdated by activating all disabled actions.

Penalty message is as follows: when A_i node receives penalty message, probability vector updates its action by fining selected actionφ_i^j. Moreover,

$$P_{ij}(n + 1) = (1 - \beta)P_{ij}(n) \qquad (3)$$

Where P_{ij} is the probability of which A_i node selects$V_i$vertex as the next vertex and rewards it's other actions (φ_i^j $for\ all\ x \neq j$) that is to say

$$P_{ix}(n + 1) = (1 - \beta)P_{ix}(n) + \left(\frac{\beta}{r}\right) \qquad (4)$$

Whereβ is penalty coefficient. After fining selected action, action probability vector should be reupdated by activating all disabled actions. Ifϑ=β, reinforcement schema in which learning automata updates its action probability vector is as L_{R-P}. If ϑ<<β, reinforcement schema in which learning automata updates its action probability vector is as $L_{R-\varepsilon P}$. Ifβ=0, reinforcement schema in which learning automata updates its action probability vector is as L_{R-I}. In this case, learning automata action probability is activated. When message receives penalty, learning automata action probability remains without any change. In the end, disabled actions from each learning automata will be reactivated.

3.0 THE PROPOSED METHOD

3.1 Problem Statement

In order to implement proposed design, an environment with L surface is considered in which n nodes with R radio range are equally distributed in this environment. These nodes exchange data by wireless connection. This network can be modeled as unit disk graph G= (V, E) where V represents nodes number an E represents edges. Nodes represents unique host and each edge connect two hosts which are located in the same radio range.

In order to implement this approach, a network from isomorphism learning automata is formed by equipping each node with learning automata for unit disk graph from considered network. Then in each step, learning automata select one of its actions randomly, so that messages finally reach the destination, selected action is evaluated by random environment, and action probability vectors from learning automata will be updated depending on the response received from the environment. Finally in each process repetition, learning automata converges to general policies of messages transmission with minimum latency and minimum messages duplication. Opportunistic networks are divided into two categories based on specific nodes mobility: opportunist networks with mobile infrastructure and opportunistic networks with static infrastructure. In opportunistic networks with static infrastructure, specific nodes are static ones, while in opportunistic networks with mobile infrastructure, specific nodes in the network move within it based on what was introduced before or in random routes. Assume that node which identification code is 1 wants to transmit a message to a node which identification code is 9. This message contains destination data and address, source address, unique identification code related to it and other control parameters. Learning automats – in cooperation with each other - attempt to transmit data from source to destination with the minimum cost possible. Generally we can extend source to N ones and destination to M ones; however, here we assume the mode in which one source wants to transmit a message – or even messages – to a certain destination. Considering this simple mode is due to better understanding of the matter. First, the node with identification code 1 generates a message and allocates a unique identification to it. Then, the learning automaton – which is embedded in the node – creates its own set of actions. Generally we should consider that nodes are mobile ones and their mobility as well as their mobility route is not predictable, so that we should discover message dissemination route as well as its transmission to destination node in an intelligent and dynamic way. In this article, we assume that nodes can be either mobile vehicles or mobile sensors with R radio range. If mobile vehicles are considered here, we should equip them all with wireless transmitter and receiver with R radio range. In fact, two nodes which are in the same radio range can transmit message to each other. If they are not in the same radio range, message transmission is not possible. Despite wide discreteness of these networks, there may be modes in which two nodes – or even multiple nodes – are located in the same radio range in which they can transmit messages from a discrete area to another one. The node which receives a message with a unique identification and contains that message is typically called infected node - which is infected by that message with that unique identification code. It should be noted that a local buffer is considered for each node. This buffer can only save a limited number of messages. In order to avoid uncontrolled duplication of messages, a timestamp is attributed to each message. This timestamp shows message survival in network. If that message is no longer alive in network, it will not be valid anymore; moreover, it will not duplicate any more. Nodes which do not contain message are called clean nodes. In order to implement our proposed design, we assumed that network topology is stable in a short time interval. Then, network topology changes when time passes by. Off course, we should consider that network topology will not remain the same and it is changing continuously, i.e. it is dynamic.

In order to better understand our proposed algorithm, algorithm implementation process, routing and data dissemination is plotted in figure 2. We assumed that the whole network is divided into two discrete areas. Off course, number of discrete areas is changing continuously. Therefore, in this hypothetical example, we assume that the node with identification code 1 generates a message and wants to transmit this message to node with identification code 9. Node with identification code 1 – as source node – and node with identification code 9 – as destination code – are located in two discrete areas. Then, message can be exchanged between these two discrete areas when they, i.e. two discrete areas, are connected by multiple nodes. In this article, we presented strategies for intelligent routing in this kind of networks by using learning automata capability which is solving complex problems. Traffic reduction, connection latency reduction can be listed as some advantages of this proposed design. Infected nodes exchange their data with other nodes – which are in the same transmission range as their own – in specific time periods. Then, learning automata – embedded in the infected node – creates or updates its set of actions according to data received from neighbor nodes. After that, it selects one of its actions randomly with respect to probability distribution function. In fact, it selects next nodes for infection.

3.2 The Formation Action set method

In the proposed algorithm, to form the action-set of each learning automaton, its corresponding node propagates locally a message to its one-hop neighbors. The nodes which are within the transmission range of the sender node, upon receiving the message, reply it and return their action-set information. The sender forms its action-set on the basis of the received replies, so that each node by which the message is replied is associated with action in the action-set of automaton. Action corresponds to the selection of node as a next node by learning automaton. Therefore, the action-set size of each learning automaton is strongly dependent on the degree of its corresponding node, in which among the action set of each automata we inactivate the actions momentarily that we cannot select. It means that we equal to zero the probability of inactive action selection. At the end of each repetition, the inactive actions of each active learning automaton should be activated for the next repetition [13, 14].

3.3 Proposed Algorithm

We assume that network topology is stable in a short time interval due to the fact that opportunistic networks' topology is dynamic and unpredictable. In this time interval, infected nodes begin to infect their neighbors according to proposed algorithm. Then, network topology changes as time passes. We assume a stable network topology in another short time interval again. Infected nodes begin to infect their neighbor according to proposed algorithm again. This process is repeated till message finally reach destination. Number of times that network topology is considering stable is shown by K.

Thus, in K=n step, a number of nodes are infected and the automata embed in those nodes select an action from their set of actions randomly with respect to probability function and infect corresponding node by that action. Therefore, cost of selected node is considered as W. Off course, node's cost is equal to message delivery probability which value is obtained by formula presented in [8]. It means that if node i selects node j which weight is denoted as W_{ij}. $\overline{W}_{ij}$ represents dynamic threshold in $K = n$ step which is calculated as follows:

$$\overline{W}_{ij} = [(n - 1)\,\overline{W}_{ij} + W_{ij}] \,/\, n \qquad (5)$$

Each time network topology changes, W_{ij} is compared to $\overline{W}_{ij}$. If W_{ij} is higher than $\overline{W}_{ij}$, selected action receives reward and other actions receive penalty and vice versa.

In this way, by each change in network topology and other nodes' infection by infected nodes, learning automata can verify the best carriers and deliver messages to them which are transmitted by lower latency and traffic of messages. Proposed algorithm is shown in figure 2. In each step, all infected nodes use following Pseudocode in order to infect other nodes:

Distributed Learning Automata based onMessage propagation Algorithm
: Let N denotes the Network size
: LetV_{A_i}denotes the selectednode
: Let$N_{V_{A_i}}$denotes the selectednode one hop neighbor
: Let$\propto_i$denotes the selectednode action set
: Let$p_i$denotes the selectednode action probable vector
: Let k denotes the number of created $topology$
: Let$\overline{W}_{ij}$denotes dynamic threshold
: LetW_{CDS}denote weightof $selected\ action$
: $repeat$
: $repeat$
: $if\ (i\ node\ is\ infected)\ then$
: The automata create or update$\propto_i$
: The automat create or updatep_i
: prune inode action set
: Automaton$A_i$chosen one of its actions
: $if\ (\overline{W}_{ij} > W_{ij})\ then$
:Reward the selected action
:else
: Penalize selected action
: $\forall\ i,\ k \leftarrow k+1,\ \overline{W}_{ij} \leftarrow [(k-1)\ \overline{W}_{ij} + W_{ij}]\ /\ k$
:$until\ (\ node\ i\ have\ actions\ for\ select\ )$
: Enable all the disabled actions
: $until\ (\ Message\ survival\ time\&\&Messages\ not\ reached\ their\ destination\ )$

Figure 2.Pseudo code of the proposed algorithm on Message propagation in the Opportunistic Networks

In order to better understand the proposed algorithm, algorithm implementation methods are drawn in figure 3 step to step. In t time, set of actions of s source node is equal to $\{2,\ 3\}$ which infects node 3 with ½ probability. Set of actions of node 3 is equal to $\{1, 4\}$ in which node 1 – which was infected before – is omitted from set of actions. In other words, parent node is omitted from set of node's actions. Then, node 4is infected with 100 % probability. Node 4 set of actions is equal to $\{3\}$ in which node 3 – which was infected before – is omitted form set of actions. Therefore, node 4 has no action to select from. Then we wait till$t + 1$ time. In $t + 1$ time, nodes were displaced. Then, infected nodes create their set of actions in order to select one of them from it with respect to probability function and disseminate infected nodes – which were infected by messages – with respect to proposed algorithm.

It should be noted that if two nodes select the same action, i.e. they want to transmit the same message to the same node, one of messages is not included; moreover, it is not saved in the buffer.

In the first step of $t + 1$ time, set of infected nodes is equal to$\{1, 3, 4\}$, set of node's actions with identification code 3 is equal to$\{2, 12, 1, 8\}$, set of node's actions with identification code 1 is equal to$\{2, 3, 5\}$. After set pruning, each set of actions associated with node with identification code 3, 1 and 4 is as follows: $\{2,\ 12,\ and\ 8\}, \{2,\ 5\}, \{12,\ 5\}$.

In the second step of $t + 1$ time, node with identification code 1 infects node with identification code 2, node with identification code 3 infects node with identification code 8. Then, set of node's actions with identification code 2 is equal to $\{1, 3\}$, while it becomes$\{\}$ after pruning. Set of node's actions with identification code 8 is equal to $\{11,\ 10,\ 7,\ and\ 3\}$ which become$\{7,\ 10,\ and\ 11\}$ after pruning. Set of node's actions with identification code 5 is equal to $\{4, 1\}$, whereas it become$\{\}$ after pruning.

In the third step of $t + 1$ time, node with identification code 8 infects node with identification code 11. Node with identification code 2 as well as node with identification code 5 has no action to select from. Then, set of node's action with identification code 11 is equal to$\{8\}$ which become$\{\}$ after pruning, i.e. there is no action to select from and we should wait tilltime $t + 2$.

In $t + 2$ time, set of node's actions with identification code 1 is equal to $\{5,\ 2\}$after pruning. Node 3was infected before, and then there is no need to include it in set of actions. Set of node's actions with identification code 3 is equal to$\{8,\ 12,\ 2\}$, while set of node's actions with identification code 4 is equal to$\{12,\ 5\}$.

In second step of $t + 2$time, nodes with identification codes 2, 8, 5 are also infected. Nodes with identification codes 2, 5 have no action to select from. Only node with identification code 8 has actions to select from. Set of node's actions with identification code 8 is equal to $\{11,\ 10,\ and\ 7\}$. Node with identification code 8 also infects node with identification code 11 with 1/3 probability. It is repeated so on till finally message is transmitted to destination.

After a message with a unique identification is delivered to destination, ACK is disseminated with the same id within the network and cleans all nodes which were infected by that message and prevents message duplication. It means that it omits the message from local buffer of those nodes, so that it prevents local buffer to be overflowed. Several advantages of this approach are as follows: lower latency in message transmission to destination, lower traffic in the network, reduced number of infected nodes.

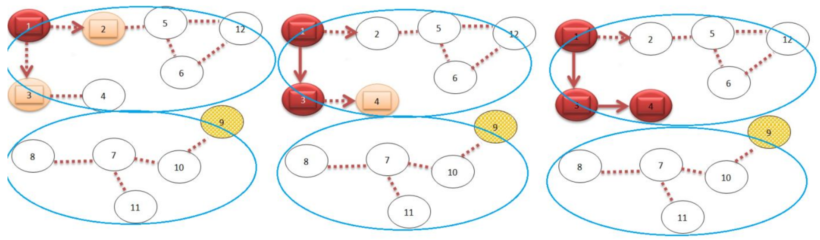

Figure3A1 Figure3A2 Figure3A1

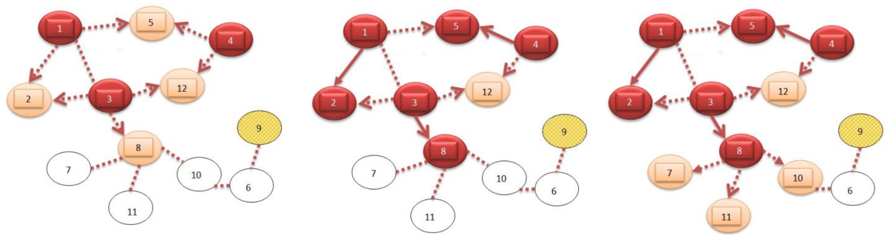

Figure3B1 Figure3B2 Figure3B3

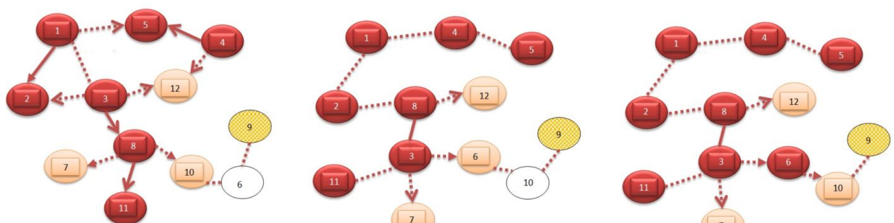

Figure3B4 Figure3C1 Figure3C2

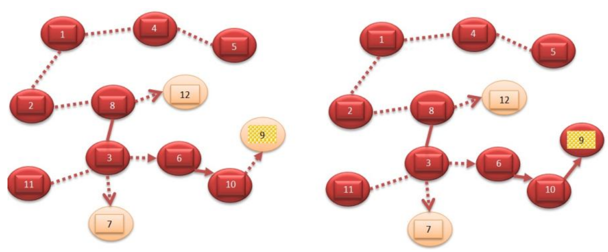

Figure3C3 Figure3C4

Figure 3.Message propagation in the Opportunistic Networks Based on Distributed Learning Automata

3.4 Improvement

In order to improve proposed approach, some limitations are applied in action selection which are called pruning rules. These rules are stated as follows:

The first rule: parent node, i.e. the node which the message is received from, is omitted from set of node's actions.

The second rule: the node which does not have infected neighbor (clean neighbor) is omitted form set of actions (with high probability, nodes are in the same area and message transmission just increase network traffic and do not affect message transmission to destination). By considering pruning rules, number of messages - which are duplicated in the network – can be dramatically decreased. Off course, it should be noted that if infected node establishes a direct connection with destination node, it transmit message to destination without any limitation and no longer act with respect to probability rules. By implementing above-mentioned rules to improve proposed algorithm, algorithm performance can be drawn in various times which is presented in figure 4.

If we assume that in t time, network topology is stable as figure 4A1, node with identification code 1, first transmit a message to its neighbors, i.e. nodes which are in the same radio range as itself, then neighbors respond to that message and exchange their data with node with identification code 1. This node creates its set of actions according to received data. In figure 4A1, nodes with identification codes 2 and 3 are located in radio range the same as node with identification code 1. Therefore, set of node's actions with identification code 1 is equal to {3 , 2} in which one of actions is randomly selected with equal probability for the first time. We assume that is selects and infects node with identification code 3.

In figure 4A2, infected nodes are nodes with identification codes 1 and 3 which should create their own set of actions and select one of them. If a node has created its own action before, it should update its own set of action in the next steps as time passes because set of action of each node is changeable and changes as time passes. In figure 4A2, set of actions of node with identification code 1 updates as {2, 3}. Then set of node's action with identification code 1 is modified as {2} after pruning. Set of node's actions with identification code 3 become{4} and it select this action as the next one. Then, node with identification code 4 is infected which is shown in figure 4A3.

Now assume that in t+1 time, network topology is as figure 4B1. Set of node's actions with identification code 1 is as {2, 3, 5} which become {} after pruning. Set of node's actions with identification code 3 is as {1, 2, 8, 12} which become {8} after pruning. Set of node's actions with identification code 4 is as {5, 12} which become {} after pruning. Then node with identification code 8 is selected for infection. In this approach, it is not necessary to infect additional nodes in order to disseminate message to destination. This feature itself reduces traffic overflow. It should be noted that changes in topology can be seen with a meticulous eye in which mobile nodes play role of mobile relays, so that message can be transmitted from transmitter to receiver, despite directresses of the network.

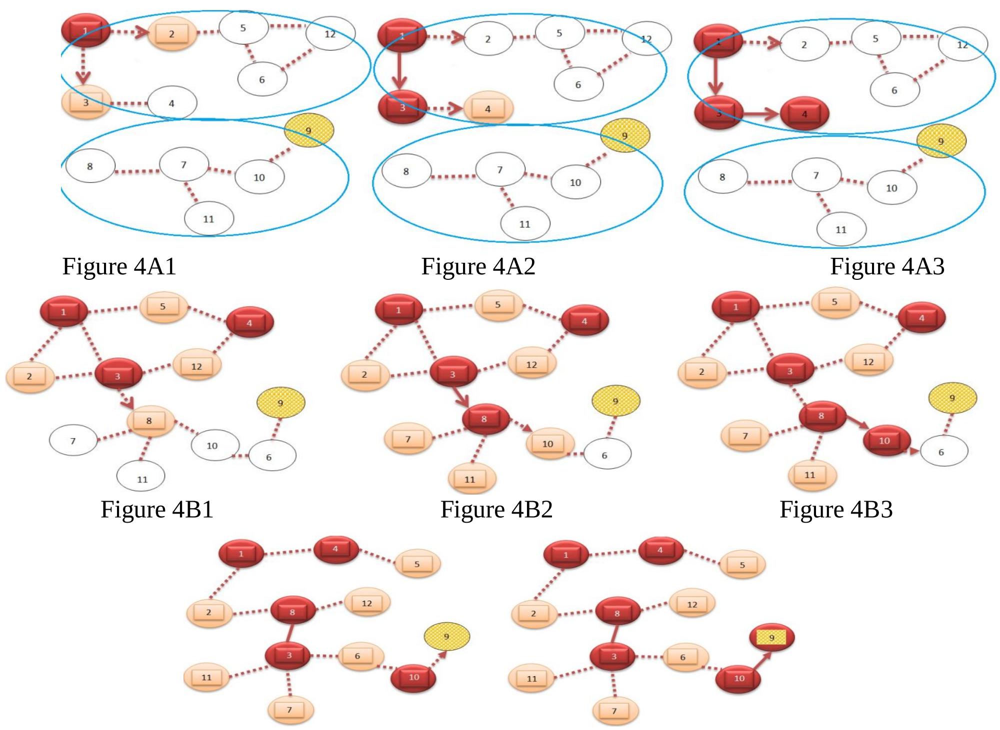

Figure 4A1 Figure 4A2 Figure 4A3

Figure 4B1 Figure 4B2 Figure 4B3

Figure 4C1 Figure 4C2

Figure4.Improvement Message propagation in the Opportunistic Networks Based on Distributed Learning Automata

4.0 EXPERIMENTAL RESULTS

In order to evaluate performance of proposed approach, we simulate performance of proposed approach with NS2 simulator software and compared its performance with two HiBOp and Epidemic protocols. Obtained results implied better performance of this approach compared to other previously presented approaches regarding buffer occupancy and traffic rate.

In order to implement this proposed approach - which is abbreviate known as IRPBDLA – as well as comparing it with HiBOp and Epidemic protocols, we attempted to consider simulated hypotheses of this approach to some extent similar to protocols' ones. Therefore, an environment with $1500\ x\ 1500\ m2$ surface is considered for simulation which is divided into ten discrete areas. Number of nodes is equal to 100 and transmission range is considered as 100 meters. We assumed that nodes are equally distributed in the environment and move with relative velocity of [2-5] meter per seconds. Number of transmitter nodes is considered as 25 and messages size as 50000 bit. Time interval between generating two subsequent messages in each transmitter is 300 seconds; however generated message is not valid any more after 5 hours. Simulation time in each period, i.e. reconfiguration time, is equal to 24 hours. It should be noted that network reconfiguration makes buffer of all nodes to be empty. In this kind of simulation, delays caused by transmission are negligible; therefore, we do not consider them in simulation.

4.1 First Experiment

In the first experiment, no limitation is imposed on nodes' buffer size. We assumed that nodes' buffer size is unlimited. According to simulation results – which is drawn in figure 5 – HiBOp and Epidemic methods occupy more space of buffer memory compared to method introduced in IRPBDLA.

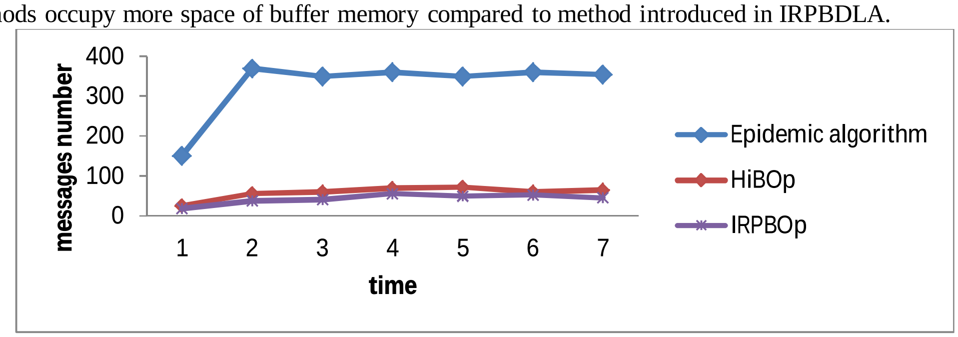

Figure 5.Buffer evolution with unlimited size

4.2 The Second Experiment

In the second experiment, 35number of transmitter is cosidered and nodes' buffer size is limited. We assumed that nodes' buffer size is limited and it only can save up to 200 number of message as maximum size. According to simulation results – which is drwan in figure 6 – HiBOp and Epidemic methods occupy more space of buffer memory compared to method introduced in IRPBDLA – even sometimes buffer overflow occurs, i.e. buffer avergely occupy all nodes completely. Therefore, opposing limitation in buffers size affect dramatically performance of this kind of protocols. However, in the method introduced in IRPBDLA, due to the fact that when time passes and messages are no longer alive, buffers will be empty of messages; moreover, ACK dessiminates in the network by the same message id as messages' one which were delivered to destination. As a result, each node omits the message which has the same id as ACK from buffer as soon as receiveing ACK. Therefore, opposing limitation on buffer size do not change performance of proposed approach; nevertheless, buffere consumption will significantly reduce.

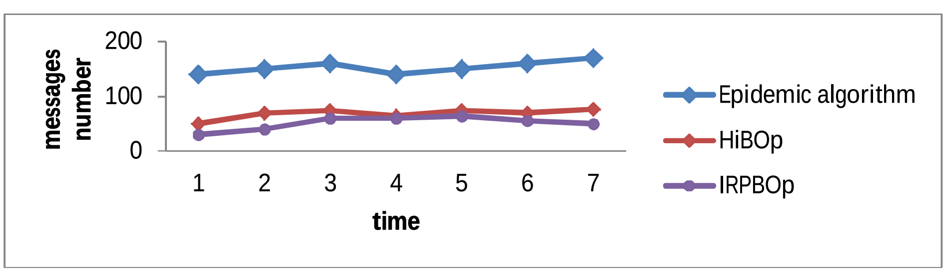

Figure6.Buffer evolution with limited size

4.3 The Third Experiment

In this experiment, performance of proposed approach is evaluated regarding traffic rate section. Traffic rate is expressed as formula 5 [8]:

$$\text{Traffic Ratio} = {}^{DM}/_{TM} \qquad (5)$$

That DM equal to total number of bytes generated over the network andTM equal to total number of bytes successfully delivered to destinations.

As it is shown in figure 7, presented approach has lower traffic compared to other presented approaches. Lower traffic in network leads to lower number of delays in transmitting messages to destination.

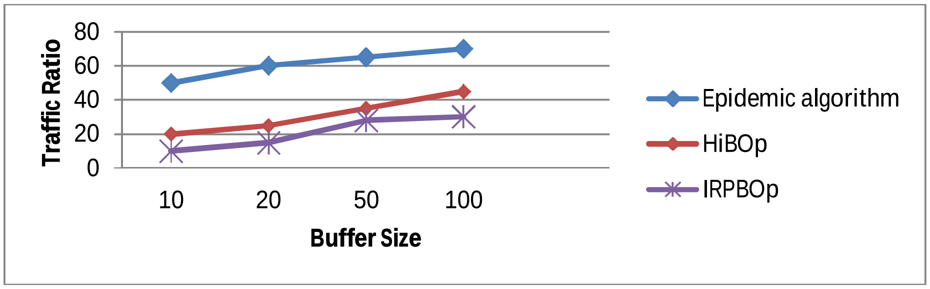

Figure 7.Traffic Ratio

4.4 Distributed Learning Automata Learn Ratio

In this experiment, buffers size is considered as 200 and number of transmitter nodes as 35. In this experiment, we want to show that buffer occupation decreases dramatically in the next rounds as learning automata begins to learn. According to figure 8, it is realized that buffer occupation has decreased dramatically. In the next steps, message transmission gradually reach ideal buffer occupation space which is essential for transmitting messages.

In the next routing steps, we find route using nodes which transmit messages to destination with higher probability. As it is shown in figure 8, first buffer occupation space is relatively high; however, it reduces in the next steps when learning automata begins its learning process and finally converges to minimum one.

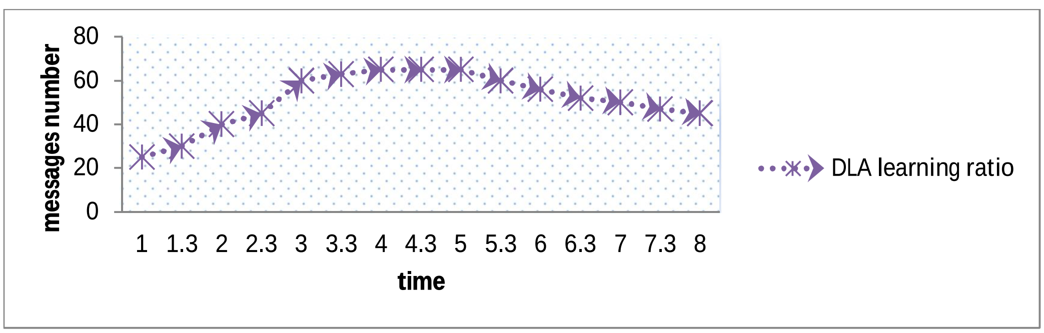

Figure 8. Distributed Learning Automata Learn Ratio

5.0 CONCLUSION

In this article, we focused on routing in a kind of networks known as opportunistic netwrorks which topology is dynamic and somewhat unpredictable. In these kind of networks, despite lack of a connection route between source and destination, network should be able to transmit messages from source to destination. In this article, we presented a novel approach based on distributed learnig automata in order to discover routes and transmit messages. The main objective of this article is finding the best carriers in to transmit messages. Obtained simulation results in traffic overflow and consumed memory sections showed optimum performance of this approach compared to other previously presented approaches.

REFERENCES

[1] Pelusi L, Passarella A, and Conti M.*Opportunistic Networking: Data Forwarding in Disconnected Mobile Ad hoc Networks* . IEEE Communication MagazineNovember 2006; Vol. 44(11): 134-141.

[2] Vahdat A and Becker D.pidemic routing for partially connectedad hoc networks. Technical Report CS-200006;2000;In Duke University.

[3] Burgess J, Gallagher B, Jensen D, and Levine B N. Maxprop:Routing for vehicle-based disruption-tolerant networks. In Proceedings of IEEE INFOCOM 2006; pp. 1–11.

[4] Tan K, Zhang Q and ZhuW.Shortest path routing in partiallyconnected ad hoc networks. In Proceedings of IEEE Globecom2003;vol. 2;pp. 1038–1042.

[5] Lindgren A, Doria A, and Schelen O. Probabilistic routing inintermittently connected networks. In ACM SIGMOBILE MobileComputing and Communications Review; vol. 7(3);2003;pp. 19–20.

[6] Burns B, Brock O, and Levine B N.Mv routing and capacitybuilding in disruption tolerant networks. In Proceedings of IEEEINFOCOM; vol. 1; 2005; pp. 398–408.

[7] Spyropoulos T, Psounis K, and Raghavendra C S.Spray andwait: An efficient routing scheme for intermittently connected mobilenetworks.In Proceeding of the 2005 ACM SIGCOMM workshop onDelay-tolerant networking WDTN '05; 2005; pp. 252–259.

[8] BoldriniC, Conti M, Iacopini I, and Passarella A.HiBOp: a History Based Routing Protocol for Opportunistic Networks .in Proceedings of the IEEEWoWMoM; 2007.

[9] Musolesi M, Hailes S, and Mascolo C. Adaptive Routing for Intermittently Connected Mobile Ad Hoc Networks.in Proceedings of the 6th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks; 2005.

[10] LeguayJ, Friedman T, and Conan V. Evaluating Mobility Pattern Space Routing for DTNs. in Proceedings of the IEEEInfocom; 2006.

[11] Boldrini C, Conti M, and PassarellaA.Impact of Social Mobility on Routing Protocols for Opportunistic Networks .in Proceedings of the IEEEWoWMoM AOC Workshop; 2007.

[12] AkbariT J, Meybodi M R. An intelligent backbone formation algorithm for wireless ad hoc networks based on distributed learning automata.Computer Networks. vol.54 ;2010; pp. 826–843.

[13] Asgari CH, Akbari T Jand ZareieA.A new approach for improving lifetime in wireless sensor networks based on distributed learning automata .International Journal of Physical Sciences.Vol 7(14) ; 201; pp. 2209 - 2219.

[14] Asgari CH, Akbari T J. A New Approach to the Data Aggregation in Wireless Sensor Networks. International Journal of Computer Science Issues. Vol. 9(1); 2012; pp. 328-325.

BIOGRAPHY

ChamranAsgari *received M.S. degree in Computer Engineering in 2012, Arak, Iran. He is a Lecturer in Computer Engineering Department at Payame Noor University, Iran. His research interests include data aggregation algorithms in wireless sensor networks, learning systems, parallel algorithms, virtual Backbone formation and minimum spanning tree and Opportunistic Networks and task migration...*