Wireless Netw (2017) 23:703–714
DOI 10.1007/s11276-015-1180-0

# Real-time routing algorithm for mobile ad hoc networks using reinforcement learning and heuristic algorithms

**Ali Ghaffari**[1]

Published online: 8 January 2016


**Abstract** Mobile ad hoc networks (MANETs) consist of a set of nodes which can move freely and communicate with each other wirelessly. Due to the movement of nodes and unlike wired networks, the available routes used among the nodes for transmitting data packets are not stable. Hence, proposing real-time routing protocols for MANETs is regarded as one of the major challenges in this research domain. Algorithms compatible with the changes created in the network due to the nodes' movements are of high significance. For reducing data packet transmission time among nodes, not only should route shortness be considered but also route stability should be taken into consideration. Since available factors in different environments have specific behavior patterns especially in human environments, the parameters of link stability and route shortness were taken into consideration and the reinforcement learning was used to propose a method so as to make the best choice among the neighbors at any moment to transmit a packet to the destination. That is, the proposed method was aimed at predicting the behavior pattern of the nodes in relation to the target node through using reinforcement learning. The proposed method used Q-learning algorithm which has more homogeneity to estimate the value of actions. Simulation results in OPNET demonstrate the superiority of the proposed scheme over conventional MANET routing methods.

**Keywords** MANET · Reinforcement learning · Q-learning · Real-time routing · Dynamic programming

✉ Ali Ghaffari
A.Ghaffari@iaut.ac.ir

[1] Department of Computer Engineering, Tabriz Branch, Islamic Azad University, Tabriz, Iran

## 1 Introduction

A MANET refers to a wireless network which can be dynamically configured without any infrastructures. Features such as: dynamic topology [1], high mobility [2, 3] of nodes, low bandwidth of channel for packet transmission and the limitation of energy resource distinguish MANETs from other networks [4–6]. These features highlight the need and requirement for designing new methods and operations for routing protocols in MANETs. MANETs have many significant applications in different fields such as military functions, emergency searches and critical operations [3, 7]. Routing protocols in MANETs should adapt themselves rapidly with frequent changes and unpredictable topology [2] and should consume the processing and communication resources optimally. MANETs encounter numerous issues which are caused by the inherent design [8]. Achieving high throughput while discovering the best end-to-end path in multi-hop networks from source to destination node is of high significance [9]. Due to the unsustainable topology of MANETs and since the movement of nodes is self-organized and not centrally controlled, multi-objective routing [10, 11] is regarded as one of the main challenges in MANETs. Because of the movement of nodes, the created routes among nodes are unsustainable and such unsustainability not only increases packet delivery time but also wastes energy resources and partitions may occur [12, 13]. Indeed, it can be argued that the majority of challenges in MANET are attributed to the topology dynamicity [1] of these networks.

In this paper, an attempt was made to propose an optimal method by predicting the behavioral patterns of nodes and reducing packet transmission delay. To the best of our knowledge, few studies have used reinforcement learning (RL) for routing in MANETs. Hence, this research gap was

addressed in the present study where a method was put forth for MANETs based on reinforcement learning and Q-learning algorithm [14]. The proposed method (RL) relied on the local data of the neighboring nodes and it did not make any assumptions about the environment. Taking the parameters of sustainability and route shortness into account, we used the reinforcement learning based on trial and error to propose a method which can choose the best alternative among all the neighbors for transmitting a packet to the target. In other words, using the reinforcement learning, we tried to predict the behavioral patterns of nodes in relation to the target node. Indeed, the proposed method estimates the movement pattern of nodes indirectly using Q-learning which is considered to be a robust algorithm in the area of reinforcement learning. In case a packet reaches the target, the method will allocate an award for the respective action; updating the value of actions is based on Q-learning. As the number of packet transmission and reception increases in the network and as the actions are repeatedly selected, regarding the number theory, the value of actions get closer to their own real values. Inasmuch as the topology is dynamic and the purpose is to select the best action in any given time, it should not be assumed that a delay in packet transmission should converge to a fixed value. In line with the purpose of making a compromise between exploration and exploitation, the proposed method used *HELLO* packets periodically. This action makes it possible to estimate the values of actions which have not been selected so far. Furthermore, the method proposed in this paper uses the packet delivery success rate for selecting an action.

The rest of the paper is organized as follows: Sect. 2 briefly classifies and reviews the related works in this area. Section 3 describes reinforcement learning scheme. Section 4 describes the proposed method and Sect. 5 reports the implementation, comparison and simulation results of the study. Finally, Sect. 6 concludes the study and recommends directions for further research.

## 2 Related works

Routing algorithms for MANETs are divided into seven groups based on their underlying architectural framework.

### 2.1 Table-driven or proactive routing protocols

Proactive protocols always maintain up-to-date routes from each node to every other node in the network. Routing information is stored in the routing table of each node. These types of protocols are not suitable for highly dynamic networks due to excessive control overhead generated to keep the routing tables fresh for each node in the network. The advantage of table-driven routing protocol is the short response time in determining a good route due to the up to date network topology in each node. This short response time, however, is at the expense of consuming a large portion of network bandwidth for the non-productive control packets to maintain network overview at each node [15]. Protocols such as DSDV [16] and OLSR [17] fall into this category. In the ant colony-based routing algorithm, like ant-net-DSR, backward nodes are transmitted so as to search for path. However, as these backward nodes pass each node, they leave some positive pheromone in the routing chart of the nodes. While routing, the nodes with more remaining pheromone are more likely to be selected [18]. In mobile agent routing, each of the information packets functions as an agent which collects network information. In this algorithm, the agents keep a history of a fixed size which includes nodes whose agents have been visited by them. When these agents reach the target, the target node examines the history of the agent; in case the route in the agent history is better than the route included in the target node' routing table, the table will update itself [19].

### 2.2 On demand or reactive routing protocols

Reactive routing protocols only keep up the required routing for the nodes. In this category, the route is created only when the source requests a route to a destination. The path is created though broadcasting RREQ message and receiving RREPLY. In a reactive routing protocol, a node does not need to periodically broadcast the routing table thereby improving network bandwidth. Protocols such as AODV [20], DSR [21] and TORA [22] fall into this category.

Dvir and Vasilakos [3] proposed an alternative, highly agile and dynamic backpressure routing for DTNs, in which routing and forwarding decisions are made on a per-packet basis. In this routing scheme using information about queue backlogs, random walk and data packet scheduling nodes can make packet routing and forwarding decisions without the notion of end-to-end routes. Simulation results show that this scheme has advantages in terms of DTN networks. Zeng et al. [10] proposed a direction routing and scheduling scheme (DRSS) for green vehicle delay tolerant networks by using Nash Q-Learning technique. It optimizes the energy efficiency and packet deliver ratio with the considerations of congestion, buffer and delay. It uses a self-learning method and selects optimal routes within some preferred areas, which helps to route packets efficiently towards the destination. Dowling et al. [23] used collaborative learning to determine the traffic of neighbor channels and used the Boltzmann equation as the policy and criterion for selecting an action.

### 2.3 Hybrid routing protocols

The hybrid routing methods such as zone routing protocol (ZRP) [24] combine elements of table-driven and on-demand routing protocols. By appropriately combining these two approaches the system can achieve a higher overall performance. In [25] the challenges and design issues for different routing metrics are discussed, along with a scheme to develop hybrid routing metrics by combining different metrics together. Protocols like ZRP [24] reduce both the typical delays of the reactive protocols and the communication overhead introduced by the proactive protocols.

### 2.4 Location-aware (geographical)routing algorithms

Location-aware routing protocols such as [26] assume that the individual nodes are aware of the locations of all the nodes within the network. This location information is then utilized by the routing protocol to determine the optimum routes.

### 2.5 Multipath routing algorithms

Multipath routing protocols such as [27] create multiple routes from source to destination. The main advantages of this scheme is that the bandwidth between links is used more effectively with greater delivery reliability.

FMRM algorithm [28] uses the fuzzy logic to select the route. This algorithm uses fuzzy controller. Indeed, using fuzzy input parameters, FMRM [28] obtains the expiration time, packet delivery rate and queue length for each of the neighbors which have a route to the destination. Hence, it obtains a priority for each of the nodes in relation to the transmitted packet. Then, based on the priorities, it transmits the packet. The values of each of the input parameters are classified into low, medium, high and very high.

### 2.6 Hierarchical routing algorithms

Hierarchical routing protocols such as [29] build a hierarchy of nodes, typically through clustering techniques. Cluster heads (CHs) provide data aggregation and data fusion, improving the scalability and the efficiency of routing. Vasilakos et al. [30] proposed the use of a computational intelligence approach to a Reinforcement Learning Algorithm (RLA) for optimizing the routing in ATM and networks based on the Private Network-to-Network Interface (PNNI) standard. The purpose of the solution addresses the QoS issues related to routing, where network resources are allocated wisely to ensure the QoS requirement should be met for each connection available in the network. Through RLA protocol optimizes the various parameters of hierarchical network structure i.e. computation, communication and storage requirements needed for routing. This algorithm, aims at maximizing the network revenue while ensuring the QoS requirements for each connection.

### 2.7 QoS-aware routing algorithms

QoS-aware routing protocols such as [31] consider QoS parameters such as residual energy, delay, static resources capacity [32], dynamic resources availability [32], neighborhood quality [32], and link quality and stability in route discovery path. Many QoS-based routing algorithms have been proposed for MANETs [10–12, 25, 33–42]. Network lifetime and average end-to-end delay are important QoS parameters for a MANET. In order to avoid network partitioning, it is important to maximize the network lifetime before the nodes fail because of battery exhaustion. A routing protocol which allows to reduce the average end-to-end delay is desirable for all real-time applications [43]. In this sub-section we present some of the latter works, by analyzing their main features and drawbacks. Since this work is specifically concerned with the real-time routing protocol, we cover in more detail extensions of these protocols.

Yen et al. [2], proposed a multi-constrained QoS multicast routing scheme using genetic algorithm (GA) with considering available resources and minimum computation time in a dynamic environment. By selecting the appropriate values for parameters such as crossover, mutation, and population size, the GA improves and tries to optimize the routes. This protocol forms a multicast tree and calculate total delay and residual energy of all nodes in the tree. A high fitness value minimizes the delay and maximizes the residual power in the tree. Simulation results demonstrate the superiority of the proposed scheme over other routing methods. Vasilakos et al. [44] proposed very emerging area Information Centric Network (ICN) with the benefits of implementing such network and various open research issues. In [4], the authors proposed a reliable multicast protocol, called CodePipe, with advanced performance in terms of energy efficiency [45, 46], throughput, and fairness in lossy wireless networks. Built upon opportunistic routing and random linear network coding, CodePipe not only simplifies transmission coordination between nodes but also improves energy efficiency, fairness and the multicast throughput significantly by exploiting both intra-batch and inter-batch coding opportunities. CodePipe is able to build a reliable data delivery mechanism in a lossy wireless network. CodePipe was evaluated on NS2 simulator by comparing with MORE and Pacifier. Ghaffari [47] proposed an energy-efficient routing

protocol for wireless sensor networks (WSNs) using A-star algorithm. In this scheme, cost function considers parameters such as residual energy, free buffer and link quality of the next neighbor node for selecting the optimal path. This scheme leads to the optimization of energy consumption and packet delivery ratio. Meng et al. [9] proposed single path and any path routing schemes using the spatial reusability method. This provides high end-to-end throughput. Vasilakos et al. in [13] presented a systematic exploration of DTN. In this book, various chapter contributed by eminent authors cover various aspects and open areas of future research in this domain. In this book various issues related to MANET such as networking, wireless, mobile communications, and technology analysis, an energy-aware routing protocol for DTNs, and a routing-compatible credit-based incentive scheme for DTNs, mobile peer-to-peer systems over DTNs, delay tolerant monitoring of obility-assisted are discussed. They also contributed chapters on various DTNs in satellite, vehicular, mobile, space and wireless sensor network communication. A biological model of Physarum [33, 48], is used for designing a novel biology-inspired optimization algorithm for minimal exposure problem (MEP). First, formulate MEP and the related models and then convert MEP into the Steiner problem by discretizing the monitoring field to a large-scale weighted grid. Physarum is able to find the shortest path to a food source through a maze like structure. Their model maintains low running complexity and still achieves high level of parallelism. They demonstrate that their solution is comparable with that of classical approximation methods and is yet more efficient. Extensive simulations demonstrate that the proposed models and algorithm are effective for finding the road-network with minimal exposure and feasible for the Steiner problem [48]. Vasilakos et al. [49] proposed a EFRouter routing system which sends the data in a feasible path using fuzzy set theory and GA. They use statistical information to predict traffic load on different links. This traffic load is then used to calculate the cost of selecting a path. When a path is selected, future probability of having to refuse further connections is taken into consideration for cost calculation, using the fuzzy model. In this scheme path selection is based on quantization of per rate success probability. Shen et al. [50] proposed the various peer-to-peer media streaming systems that deployed successfully, and corresponding theoretical investigations have been performed in the system. It is responsible in their potential in balancing the load and improving system scalability. MANET with Q-routing protocol used the Q-learning algorithm for routing in wired network so as to reduce packet transmission. In this protocol, the number of hops was used as a reward for nodes [51].

The majority of algorithms proposed so far have not paid significant attention to the movement pattern of nodes or they have considered numerous assumptions and hypotheses about nodes and network topology. Using reinforcement learning can be regarded as a relatively novel idea and concept in routing MANETs. It should be noted that this issue is considered to be a research gap which should be addressed by further research. Thus, as an attempt to fill this research gap, in this paper, proposed a routing algorithm using reinforcement learning and only the local data of nodes. This algorithm was aimed at providing an optimal route for transmitting data packets.

## 3 Reinforcement learning

Reinforcement learning (RL) refers to a type of learning which is achieved through interaction. The learner and decision maker is called agent. The agent selects the actions and functions and the environment reacts to the agent's action. State indicates the new position for the agent (and agent moves to new position). Indeed, based on the action and function fulfilled by the agent, the agent may enter a new state which is likely to be a former state. Also, environment gives a reward for each action of the agent; the reward might be positive or negative. The reinforcement learning issue can be expressed as Markov decision-making process. Markov process consists of four components:

1. The set of states $\{s_1, s_2, s_3, \ldots\}$,
2. The set of actions $\{a_1, a_2, a_3, \ldots\}$,
3. State transition function $Pr^{a}_{ss'} = \Pr\{s_{t+1} = s' | s_t = s, a_t = a\}$.
4. Reward function $R^{a}_{ss'} = E\{r_{t+1} | s_t = s, a_t = a, s_{t+1} = s'\}$ and policy function $\pi(s) \rightarrow a$.

In reinforcement learning, the purpose is to find an optimal policy for which value function is used. State value function estimate *how good* it is for the agent to be in a given state. Accordingly, value functions are defined with respect to particular policies. The following Eq. (1) illustrates this function:

$$V^{\pi}(s) = E_{\pi}\{R_t | s_t = s\} \Rightarrow \sum_{a} \pi(s,a) \sum_{s'} P^{a}_{ss'} \left[R^{a}_{ss'} + \gamma V^{\pi}(s')\right] \quad (1)$$

In Eq. (1), $\gamma$ denotes the effect of the value of next states in the current state $(0 > \gamma < 1)$. Action value function refers to the expected total reward promotions with taking action $\alpha$ in which $\pi$ policy is used from next states until the ending state.

$$Q^{\pi}(s,a) = \sum_{s'} P^{a}_{ss'}\left[R^{a}_{ss'} + \gamma V^{\pi}(s')\right] \quad (2)$$

In Eq. (2), one of the actions is selected based on assumption. The optimal action value function which is independent of policy is defined as follows:

$$Q^{*}(s,a) = \sum_{s'} P^{a}_{ss'}\left(R^{a}_{ss'} + \gamma \max_{a'} Q^{*}(s',a')\right) \quad (3)$$

In Eq. (3), we should select an action which has the highest value.

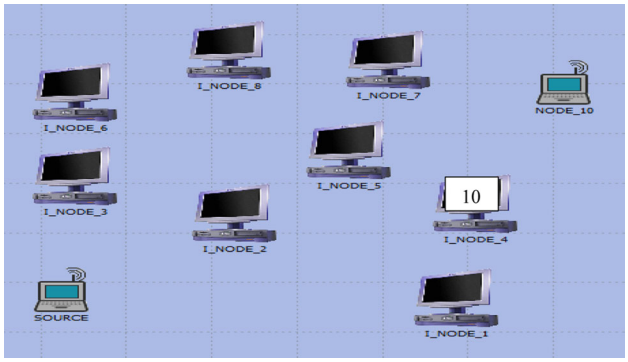


**Fig. 1** Defining state for the proposed method

## 4 The proposed method

Based on the chaos-complexity theory, it can be argued that an order and discipline can be defined for the majority of environments. We can seldom find order and discipline in environments and organizations where humans play the major roles. Hence, a method has been proposed in this paper which is intended to estimate the behavioral pattern of the nodes indirectly; this method uses Q-learning and selects neighboring nodes with little transmission delay in order to reduce the packet transmission time. A set of states was defined for the proposed method.

In this paper, the states of nodes were defined with regard to the target node.

The node which should transmit a packet to another node (ID = 10) is in state 10. For defining actions, the nodes are divided into 5 groups based on their IDs. For example in Fig. 1, nodes with the ID = 1, 2, 3, 4, 5 are within one group. Indeed, in the majority of previously proposed methods, actions are the nodes themselves; however, in the method proposed in this paper, actions are the groups. Firstly, the node selects one node among the available nodes for transmitting data packet. The purpose of grouping nodes is not only to reduce the number of actions but also to rank the value of actions. That is, while selecting actions, the agent firstly selects a group as the transmitter group from among the groups which are at the first level. In selecting groups, the action values of the groups are used. The values of actions are an estimate of the delay which is obtained from the available nodes in the group. In addition to the delay, the success rate of the group in transmitting the packet is also used to estimate the value of groups. After the selection of a group which has the best performance in transmitting packets, the node with the lowest number of proposed hops is selected as the next node to transmit packet. In this way, the probability of the selection of a node with better performance is enhanced. The policy used for selecting an action is based on the following Boltzmann probability distribution:

$$P(s,a) = e^{\frac{Q[s,a]}{T}} / \sum_{a} e^{\frac{Q[s,a]}{T}} \quad (4)$$

In Eq. (4), Q(s, a) is the action value function. The value of actions (groups) for transmitting packet depends on the value of the nodes which are within that group. That is, the value of groups is a product of the value of the nodes of the groups. After the selection of the group to transmit a packet, the node for transmission is selected based on the number of hops offered by the nodes to transmit a packet to the target. The node offering less number of hops is selected as the next hop for packet transmission. The pseudo-code related to route selection section is given as follows:

**Algorithm 1** Pseudo-code of Route Selection

```
For (each action)
    {
      // begin for each
      If (Exist_Node_For_Action())
      {
        //begin if
        Calculate P(s, a);
        Insert to Probability Array;
        //end if
      }
      Else
      {
        //begin else
        P(s, a) =0;
        Insert to Probability Array;
        //end else
      }
    Sort (Probability Array);
      Generate Random Number Between 0,1 using Normal Distribution;
      Select Action;
      Select Node From Action(Group) based on minimum hop count;
      Send packet to selected Node;
      }
```

The value of actions is calculated by means of the Q-learning updating equation.

$$T_D = \mathrm{T_D} + \alpha * (\mathrm{Rt} + (\gamma * \mathrm{Min_{Value}}) - \mathrm{T_D}) \quad (5)$$

$$\mathrm{P_r} = \mathrm{P_r} + \alpha * (\mathrm{Rt} + (\omega * \mathrm{Max_{Value}}) - \mathrm{P_r}) \quad (6)$$

Equation (5) is used for updating transmission delay and Eq. (6) is used for updating success packet deliver rate.

It should be noted that in addition to the transmission time, the success rate in delivering a packet to the destination is taken into consideration in selecting the group to transmit packets. According to the selection policy, the group with a higher action value is more likely to be selected. Hence, the proposed algorithm cannot be expected to select the best action at any moment; this problem is considered as a drawback for the algorithm. Figure 2 depicts the internal structure of the routing module which is the most important section of the routing algorithm.

As shown in Fig. 2, the routing module consists of several states. At any moment, nodes can be in one of the states. State 1 is used for node's decision about transition to the next state. For instance, when a packet is produced for transmission, in state 1, the node makes a decision based on

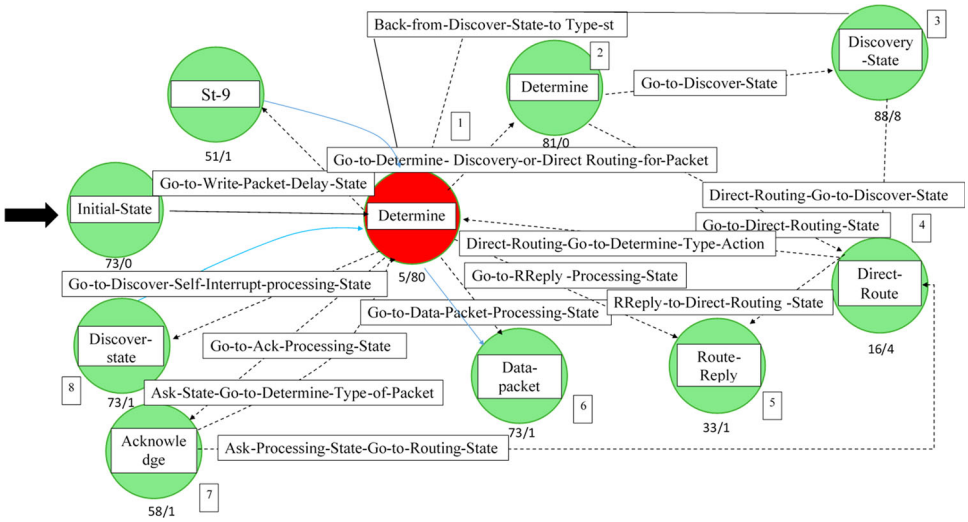


**Fig. 2** The internal structure of the routing module

the received signal to enter state 2. In state 2, the node makes a decision to conduct the route discovery operation or it decides to go towards the available target node and do the routing operation. In state 3, the node conducts the route discovery operation; then, it goes back to state 1 and waits for the input signals so that it can make the appropriate decision. In state 4, the node conducts the routing operation under the condition where it is the target node; then, it goes to state 1. In state 5, the node receives the *RREPLY* and updates the routing table. In case the *RREPLY* packet is intended for the node, after updating the routing table, the node will go to sate 4 for routing. The node will be in state 6 if it can operate as the interface node so as to transmit the packet to the target. In state 7, the node receives the packet *Acknowledge* and carries out the necessary updates for the value of the actions. When the required time for the return of the packet *RREPLY* is spent but the packet is not received, the node will go to state 8.

While transmitting route discovery packets and data packets, the source node produces a list called *node path list* and puts its own address at the beginning of the list. The node path list keeps the addresses of the nodes through which packets have passed. As each middle node receives these packets, in case a packet is not for the node itself, it adds its address to the end of the list and transmits the packet. In case the packet is for the node, the node inverts the node path list and transmits the packet to the first node in the node path list. Also, as each node receives *RREPLY* packet and *acknowledge packet*, they take their own addresses from the beginning of the list and transmits it to the next node in the inverted node path list until the packet is delivered to the destination node. The pseudo-code related to Fig. 2 is given below:

**Algorithm 2** Pseudo-code related to Fig.2

```
1: Program Start;
2:        Initial ϵ, a, Q(s, a);
3:        While (1)
4:        {
5:           Do
6:          // Node receive Packet;
7:           Switch (Packet)
8:           {
9:              Case discover:
10:                Packet is for me send back Route Reply Else Re-broadcast
11:             Case Route Reply:
12:                Packet is for me Update Routing_Table Send packet
13:                 Else Resend Packet
14:             Case Data_Packet:
15:                  Packet Is Not For me exist Path select Next Group& Node& send Packet
16:                  Packet is for me
17:                 Update Q(S, a) & Send Ack_Packet
18:             Case Ack_Packet:
19:               Source_ID is me Update Q(S, a) else Update Q(S, a) & Send Ack_Packet
20:                With new Q(s, a) value
21:          }
22:       }
```

## 5 Performance evaluation of the proposed method

For evaluating the efficiency of the proposed method in transmitting the packet, it was compared with the algorithms E-Ant-DSR [19], dynamic source routing (DSR) [52] and ant-colony based routing algorithm (ARA) [53]. The simulation environment OPNET 14.5 was used. Tables 1 and 2 show the simulation parameters and their values for first and second scenarios respectively.

In the first scenario, as shown in Fig. 3, the proposed algorithm is relatively more sustainable than the DSR [52] algorithm; at the beginning, the algorithm had more delay than DSR [52]. This can be attributed to the policy of selecting the actions. In fact, the selected policy was based on probability and at the beginning of the simulation, the probabilities are equal and all the actions are equally probable for being selected. As the algorithm time passes, the prediction of the neighbor values is obtained and the delay time in transmitting the packet is improved.

Also, as shown in Fig. 4, it can be observed that, in an environment relatively sustainable in terms of topology, the

**Table 1** Simulation parameters of the first scenario

| Simulation parameter | Value |
|---|---|
| Number of nodes | 20 |
| Dimension | 30 m × 30 m |
| Movement model | Random way point |
| Traffic model | CBR (constant bit rate) |
| Speed of nodes | 0–5 m/s |
| Stop time | 5 s |
| Simulation time | 600 s |

**Table 2** Simulation parameters of the second scenario

| Simulation parameter | Value |
|---|---|
| Number of nodes | 40 |
| Dimension | 50 m × 50 m |
| Movement model | Random way point |
| Traffic model | CBR (constant bit rate) |
| Speed of nodes | 0–10 m/s |
| Stop time | 0–10 s |
| Simulation time | 600 s |

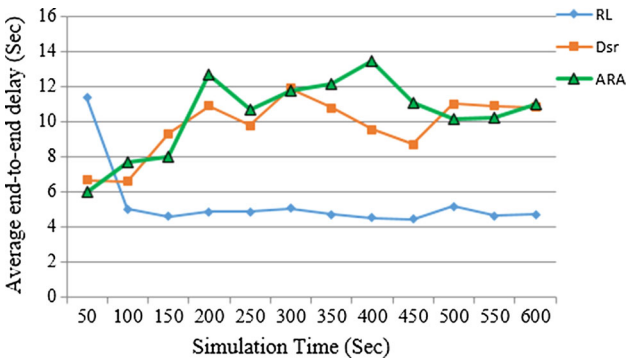


**Fig. 3** Average end-to-end delay (first scenario)

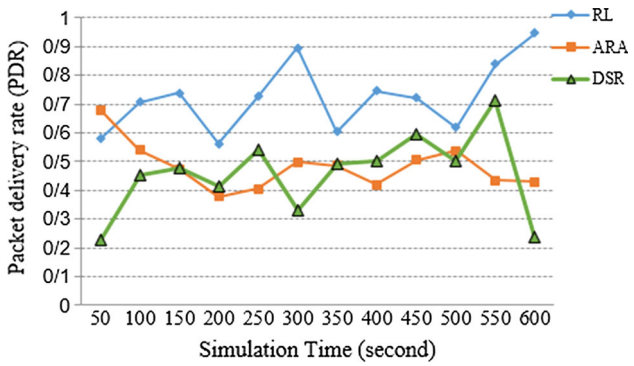


**Fig. 4** Packet delivery rate

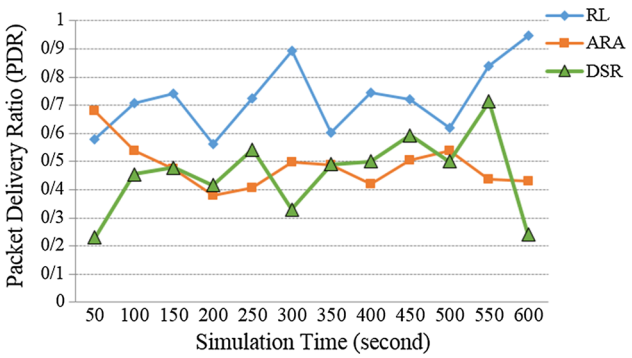


**Fig. 6** Packet delivery ratio

algorithm proposed in this paper performs better than the optimal ant colony routing algorithm with respect to compatibility. Figure 4 indicates that as the number of nodes increases and even when the environed is more sustainable, the proposed algorithm functions better than DSR [52] and ARA [53]. It should be noted that as the number of nodes increases, the number of possible actions for the nodes increases; however, even in such conditions, the proposed algorithm performs better than DSR [52] and ARA [53].

In the second scenario, in addition to the enhancement of the stop time parameter to 10 s, for increasing the movement of nodes, the speed of the nodes was enhanced to 0–10 m/s. The simulation results are given in Fig. 4.

Figure 5 reveals that the proposed algorithm works better than the DSR algorithm in an environment which is less sustainable. With respect to the movement of nodes and many changes in terms of location, it can be argued that the proposed algorithm has significantly more efficiency than the DSR [52] algorithm. The proposed algorithm was compared with ARA [53] and DSR [52] in an instable environment in terms of packet delivery rate. In this scenario, the agreement time of nodes was reduced to a range from 0 to 5 s and the movement speed was enhanced to the range from 0 to 10 m/s.

As shown in Fig. 6, it can be observed that, with regard to the instability of route, the proposed algorithm has higher efficiency than ARA [53]. This can be attributed to the criterion used in the proposed algorithm for selecting the next step and the slowness of ARA [53] in adapting itself to the new topology. Noticing this figure reveals that the proposed algorithm functions better than the optimal ant colony routing algorithm in a relatively sustainable environment. Since congestion [54] is created in the interface nodes as a function of the route selection policy in the ant colony algorithm, it can be pointed out that the proposed algorithm has a better performance.

In the second scenario, the proposed algorithm was compared with the algorithms which used neural network for routing. It was also compared with the one which was based GA. Moreover, the proposed algorithm was compared with the one put forth by [19] which was composed of a combination of DSR [52] and ACO algorithms.

As illustrated in Fig. 7, it can be noted that, in a stable environment, the proposed algorithm has higher efficiency than ANN. Since the proposed algorithm uses clustering [45, 55] to specify the value of actions, it can provide a better estimation of the value of actions. Furthermore, the proposed algorithm uses the law of large numbers to predict the value of actions. Consequently, the

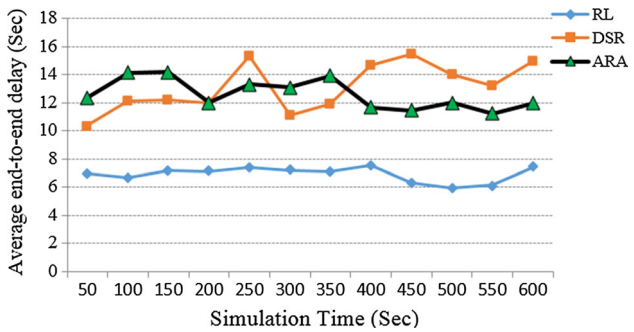


**Fig. 5** Average end-to-end delay

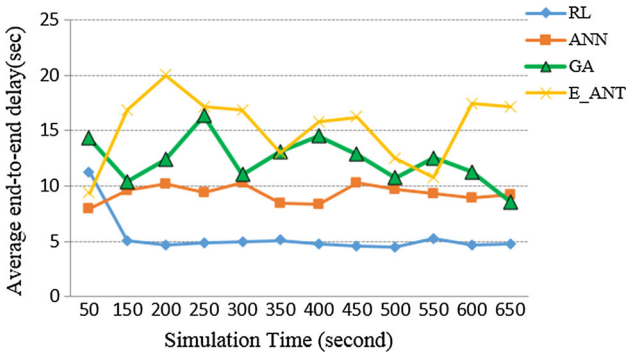


**Fig. 7** Average end-to-end delay

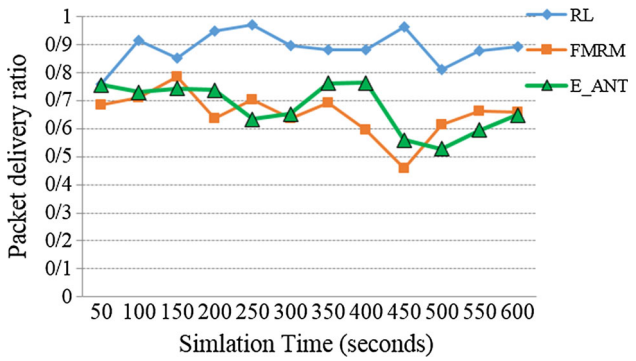


**Fig. 8** Packet delivery rate

values obtained by the proposed algorithm is more real and valid. The ANN algorithm uses neural networks to determine the value of actions where the estimations are based on the previous estimations. The GA merely considers the topology of the environment in selecting the route. Inasmuch as the environment has a dynamic topology, it is prone to error in selecting the optimal route. The E-Ant-DSR [19] algorithm selects the route based on the parameters which vary with time. Consequently, E-Ant-DSR [19] algorithm cannot have a better adaptability in line with the topological changes which leads to more delay in delivering a packet to the target.

As shown in Fig. 8, the proposed algorithm indicates a better packet delivery rate than the other algorithms. It should be noted that the proposed algorithm uses probability to select actions. Since E-Ant-DSR [19] uses ant colony, it operates slower than the proposed algorithm in adapting itself with the constant changes of the environment. As a result, the packet delivery rate is reduced in E_Anet_DSR [19] algorithm. Moreover, inasmuch as FMRM [28] uses parameters which vary with time in selecting the next nodes, its efficiency decreases.

## 6 Conclusions and future works

In this paper, an algorithm based on reinforcement learning was proposed which was based on local information. The obtained results illustrated through the figures indicated that reinforcement learning is a promising concept in MANETs. Furthermore, as mentioned earlier in the paper, it should be highlighted that the proposed algorithm had no assumptions about the environment and it relied only on the local information of nodes obtained from the neighbors. This approach can be considered and followed by future studies. As simulation results showed, as the time passes, delay decreases in relate on to the initial times; this feature can be further examined in future studies. Also, packet delivery rate obtained for the proposed algorithm was promising. It should be pointed that both packet delivery rate and time can be further improved by the selection of better policies. These research agenda should be all addressed by interested future researchers. The following recommendations can be considered in the future studies:

- Action selection policy in the proposed algorithm was based on Boltzmann equation and policy improvement throughout time was another critical factor in enhancing the efficiency of the algorithm. It is recommended that possible policies be examined and searched so that an optimal routing policy is selected. Future studies should examine complementary algorithms as a significant research agenda.
- Grouping type (clustering) in the proposed algorithm was simple. Other grouping patterns can be used in future studies. In other words, using better and more optimal criteria for clustering of the nodes may enhance the efficiency of the algorithms.
- Last but not least, better states and conditions should be defined for nodes.

## References

1. Li, M., Li, Z., & Vasilakos, A. V. (2013). A survey on topology control in wireless sensor networks: Taxonomy, comparative study, and open issues. *Proceedings of the IEEE, 101*(12), 2538–2557.
2. Yen, Y.-S., Chao, H.-C., Chang, R.-S., & Vasilakos, A. (2011). Flooding-limited and multi-constrained QoS multicast routing based on the genetic algorithm for MANETs. *Mathematical and Computer Modelling, 53*(11), 2238–2250.
3. Dvir, A., & Vasilakos, A. V. (2011). Backpressure-based routing protocol for DTNs. *ACM SIGCOMM Computer Communication Review, 41*(4), 405–406.
4. Li, P., Guo, S., Yu, S., & Vasilakos, A. V. (2012). CodePipe: An opportunistic feeding and routing protocol for reliable multicast with pipelined network coding. In *INFOCOM, 2012 Proceedings IEEE, 2012* (pp. 100–108). IEEE.
5. Rahimi, M. R., Venkatasubramanian, N., Mehrotra, S., & Vasilakos, A. V. (2012). MAPCloud: Mobile applications on an elastic and scalable 2-tier cloud architecture. In *Proceedings of the 2012 IEEE/ACM fifth international conference on utility and cloud computing, 2012* (pp. 83–90). IEEE Computer Society.
6. Sheng, Z., Yang, S., Yu, Y., Vasilakos, A., Mccann, J., & Leung, K. (2013). A survey on the IETF protocol suite for the internet of things: Standards, challenges, and opportunities. *IEEE Wireless Communications, 20*(6), 91–98.
7. Xiang, L., Luo, J., & Vasilakos, A. (2011). Compressed data aggregation for energy efficient wireless sensor networks. In *2011 8th annual IEEE communications society conference on sensor, mesh and ad hoc communications and networks* (*SECON*), *2011* (pp. 46–54). IEEE.
8. Yang, M., Li, Y., Jin, D., Zeng, L., Wu, X., & Vasilakos, A. V. (2014). Software-defined and virtualized future mobile and wireless networks: A survey. *Mobile Networks and Applications, 20*(1), 4–18.
9. Meng, T., Wu, F., Yang, Z., Chen, G., & Vasilakos, A. (2015). Spatial reusability-aware routing in multi-hop wireless networks. *IEEE Transactions on Computers*, (1), 1–1.

10. Zeng, Y., Xiang, K., Li, D., & Vasilakos, A. V. (2013). Directional routing and scheduling for green vehicular delay tolerant networks. *Wireless Networks, 19*(2), 161–173.
11. Li, P., Guo, S., Yu, S., & Vasilakos, A. V. (2014). Reliable multicast with pipelined network coding using opportunistic feeding and routing. *IEEE Transactions on Parallel and Distributed Systems, 25*(12), 3264–3273.
12. Spyropoulos, T., Rais, R. N., Turletti, T., Obraczka, K., & Vasilakos, A. (2010). Routing for disruption tolerant networks: Taxonomy and design. *Wireless Networks, 16*(8), 2349–2370.
13. Vasilakos, A. V., Zhang, Y., & Spyropoulos, T. (2011). *Delay tolerant networks: Protocols and applications*. Boca Raton: CRC Press.
14. Watkins, C. J., & Dayan, P. (1992). Q-learning. *Machine Learning, 8*(3–4), 279–292.
15. Wang, J., Osagie, E., Thulasiraman, P., & Thulasiram, R. K. (2009). HOPNET: A hybrid ant colony optimization routing algorithm for mobile ad hoc network. *Ad Hoc Networks, 7*(4), 690–705.
16. Perkins, C. E., & Bhagwat, P. (1994). Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. In *ACM SIGCOMM computer communication review, 1994* (Vol. 24, pp. 234–244, Vol. 4). ACM.
17. Jacquet, P., Mühlethaler, P., Clausen, T., Laouiti, A., Qayyum, A., & Viennot, L. (2001). Optimized link state routing protocol for ad hoc networks. In *Proceedings of IEEE international multi topic conference, 2001. IEEE INMIC 2001. Technology for the 21st century, 2001* (pp. 62–68). IEEE.
18. Kok, J. R., & Vlassis, N. (2006). Collaborative multiagent reinforcement learning by payoff propagation. *The Journal of Machine Learning Research, 7*, 1789–1828.
19. Chatterjee, S., & Das, S. (2015). Ant colony optimization based enhanced dynamic source routing algorithm for mobile ad hoc network. *Information Sciences, 295*, 67–90.
20. Perkins, C., Belding-Royer, E., & Das, S. (2003). Ad hoc on-demand distance vector (AODV) routing. No. RFC 3561.
21. Johnson, D. B., & Maltz, D. A., Broch, J. (1996). Dynamic source routing in ad hoc wireless networks. In *Mobile computing* (pp. 153–181). Springer, Heidelberg.
22. Park, V. D., & Corson, M. S. (1997). A highly adaptive distributed routing algorithm for mobile wireless networks. In *Proceedings IEEE INFOCOM'97. Sixteenth annual joint conference of the IEEE computer and communications societies. Driving the information revolution, 1997* (Vol. 3, pp. 1405–1413). IEEE.
23. Dowling, J., Curran, E., Cunningham, R., & Cahill, V. (2005). Using feedback in collaborative reinforcement learning to adaptively optimize MANET routing. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, 35*(3), 360–372.
24. Pearlman, M. R., & Haas, Z. J. (1999). Determining the optimal configuration for the zone routing protocol. *IEEE Journal on Selected Areas in Communications, 17*(8), 1395–1414.
25. Youssef, M., Ibrahim, M., Abdelatif, M., Chen, L., & Vasilakos, A. V. (2014). Routing metrics of cognitive radio networks: A survey. *IEEE Communications Surveys and Tutorials, 16*(1), 92–109.
26. Ko, Y.-B., & Vaidya, N. H. (2000). GeoTORA: A protocol for geocasting in mobile ad hoc networks. In *Proceedings of 2000 international conference on network protocols, 2000* (pp. 240–250). IEEE.
27. Marina, M. K., & Das, S. R. (2001). On-demand multipath distance vector routing in ad hoc networks. In *Ninth international conference on network protocols, 2001* (pp. 14–23). IEEE.
28. Pi, S., & Sun, B. (2012). Fuzzy controllers based multipath routing algorithm in MANET. *Physics Procedia, 24*, 1178–1185.
29. Iwata, A., Chiang, C.-C., Pei, G., Gerla, M., & Chen, T.-W. (1999). Scalable routing strategies for ad hoc wireless networks. *IEEE Journal on Selected Areas in Communications, 17*(8), 1369–1379.
30. Vasilakos, A., Saltouros, M. P., Atlassis, A., & Pedrycz, W. (2003). Optimizing QoS routing in hierarchical ATM networks using computational intelligence techniques. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 33*(3), 297–312.
31. Das, S. K., Mukherjee, A., Bandyopadhyay, S., Saha, D., & Paul, K. (2003). An adaptive framework for QoS routing through multiple paths in ad hoc wireless networks. *Journal of Parallel and Distributed Computing, 63*(2), 141–153.
32. Boukerche, A., Turgut, B., Aydin, N., Ahmad, M. Z., Bölöni, L., & Turgut, D. (2011). Routing protocols in ad hoc networks: A survey. *Computer Networks, 55*(13), 3032–3080.
33. Song, Y., Liu, L., Ma, H., & Vasilakos, A. V. (2014). A biology-based algorithm to minimal exposure problem of wireless sensor networks. *IEEE Transactions on Network and Service Management, 11*(3), 417–430.
34. Busch, C., Kannan, R., & Vasilakos, A. V. (2012). Approximating congestion + dilation in networks via "Quality of Routing"; Games. *IEEE Transactions on Computers, 61*(9), 1270–1283.
35. Marwaha, S., Srinivasan, D., Tham, C. K., & Vasilakos, A. (2004). Evolutionary fuzzy multi-objective routing for wireless mobile ad hoc networks. In *Congress on evolutionary computation, 2004. CEC2004, 2004* (Vol. 2, pp. 1964–1971). IEEE.
36. Wang, X., Vasilakos, A. V., Chen, M., Liu, Y., & Kwon, T. T. (2012). A survey of green mobile networks: Opportunities and challenges. *Mobile Networks and Applications, 17*(1), 4–20.
37. Zhang, X. M., Zhang, Y., Yan, F., & Vasilakos, A. V. (2015). Interference-based topology control algorithm for delay-constrained mobile ad hoc networks. *IEEE Transactions on Mobile Computing, 14*(4), 742–754.
38. Cheng, H., Xiong, N., Vasilakos, A. V., Yang, L. T., Chen, G., & Zhuang, X. (2012). Nodes organization for channel assignment with topology preservation in multi-radio wireless mesh networks. *Ad Hoc Networks, 10*(5), 760–773.
39. Cianfrani, A., Eramo, V., Listanti, M., Polverini, M., & Vasilakos, A. V. (2012). An OSPF-integrated routing strategy for QoS-aware energy saving in IP backbone networks. *IEEE Transactions on Network and Service Management, 9*(3), 254–267.
40. Han, K., Luo, J., Liu, Y., & Vasilakos, A. V. (2013). Algorithm design for data communications in duty-cycled wireless sensor networks: A survey. *IEEE Communications Magazine, 51*(7), 107–113.
41. Xiong, N., Vasilakos, A. V., Yang, L. T., Song, L., Pan, Y., Kannan, R., et al. (2009). Comparative analysis of quality of service and memory usage for adaptive failure detectors in healthcare systems. *IEEE Journal on Selected Areas in Communications, 27*(4), 495–509.
42. Yao, Y., Cao, Q., & Vasilakos, A. V. (2013). EDAL: An energy-efficient, delay-aware, and lifetime-balancing data collection protocol for wireless sensor networks. In *2013 IEEE 10th international conference on mobile ad-hoc and sensor systems (MASS), 2013* (pp. 182–190). IEEE.
43. Costagliola, N., López, P. G., Oliviero, F., & Romano, S. P. (2012). Energy-and delay-efficient routing in mobile ad hoc networks. *Mobile Networks and Applications, 17*(2), 281–297.
44. Vasilakos, A. V., Li, Z., Simon, G., & You, W. (2015). Information centric network: Research challenges and opportunities. *Journal of Network and Computer Applications, 52*, 1–10.
45. Liu, Y., Xiong, N., Zhao, Y., Vasilakos, A. V., Gao, J., & Jia, Y. (2010). Multi-layer clustering routing algorithm for wireless vehicular sensor networks. *IET Communications, 4*(7), 810–816.
46. Chilamkurti, N., Zeadally, S., Vasilakos, A., & Sharma, V. (2009). Cross-layer support for energy efficient routing in

wireless sensor networks. *Journal of Sensors*, *2009*, 134165. doi:10.1155/2009/134165.
47. Ghaffari, A. (2014). An energy efficient routing protocol for wireless sensor networks using A-star algorithm. *Journal of Applied Research and Technology, 12*(4), 815–822.
48. Liu, L., Song, Y., Zhang, H., Ma, H., & Vasilakos, A. V. (2015). Physarum optimization: A biology-inspired algorithm for the steiner tree problem in networks. *IEEE Transactions on Computers, 64*(3), 819–832.
49. Vasilakos, A., Ricudis, C., Anagnostakis, K., Pedryca, W., & Pitsillides, A. (1998). Evolutionary-fuzzy prediction for strategic QoS routing in broadband networks. In *The 1998 IEEE international conference on fuzzy systems proceedings, 1998. IEEE world congress on computational intelligence, 1998* (Vol. 2, pp. 1488–1493). IEEE.
50. Shen, Z., Luo, J., Zimmermann, R., & Vasilakos, A. V. (2011). Peer-to-peer media streaming: Insights and new developments. *Proceedings of the IEEE, 99*(12), 2089–2109.
51. Di Caro, G., Ducatelle, F., Gambardella, L. M., & Dorigo, M. (2005). AntHocNet: An adaptive nature-inspired algorithm for routing in mobile ad hoc networks. *European Transactions on Telecommunications, 16*(5), 443–455.
52. Maltz, D. B. J. D. A., & Broch, J. (2001). *DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks* (pp. 13891–15213). PA: Computer Science Department Carnegie Mellon University Pittsburgh.
53. Günes, M., Sorges, U., & Bouazizi, I. ARA-the ant-colony based routing algorithm for MANETs. In *Proceedings of the international conference on parallel processing workshops, 2002* (pp. 79–85). IEEE.
54. Ghaffari, A. (2015). Congestion control mechanisms in wireless sensor networks: A survey. *Journal of Network and Computer Applications, 52*, 101–115.
55. Xiang, L., Luo, J., Deng, C., Vasilakos, A. V., & Lin, W. (2012). DECA: Recovering fields of physical quantities from incomplete sensory data. In *2012 9th annual IEEE communications society conference on sensor, mesh and ad hoc communications and networks* (*SECON*), *2012* (pp. 182–190). IEEE.

**Ali Ghaffari** received his B.Sc., M.Sc. and Ph.D. degrees in computer engineering from the University of Tehran and IAUT, TEHRAN, IRAN in 1994, 2002 and 2011 respectively. As an assistant professor of computer engineering at Islamic Azad University, Tabriz branch, IRAN, his research interests are mainly in the field of wired and wireless networks, Wireless Sensor Networks (WSNs), Mobile Ad Hoc Networks (MANETs), Vehicular Ad Hoc Networks (VANETs), networks security and Quality of Service (QoS). He has published more than 50 international conference and reviewed journal papers.