



A cross-layer learning automata based gateway selection method in multi-radio multi-channel wireless mesh networks

A. R. Parvanak¹ · M. Jahanshahi² · M. Dehghan³

Received: 3 November 2017 / Accepted: 25 July 2018 / Published online: 1 August 2018
© Springer-Verlag GmbH Austria, part of Springer Nature 2018

Abstract

Wireless networks' applications are increasing gradually necessitating their performance to enhance. Evolution of these networks over time indicates the need for algorithms which can operate based on their dynamic nature. Wireless mesh networks provide Intranet and Internet access for different applications in various environments. It is expected that the traffic load will be high on these networks. As gateway nodes are responsible for the traffic load transmission, gateway selection is known as one of the important research issues in that it can lead to optimized use of the network capacity and reduce congestion effects. In addition, utilizing multi-radio multi-channel architecture is one of the promising methods for increasing performance and decreasing interference. Channel assignment determines the most appropriate channel-radio associations for transmitting and receiving data through different channels simultaneously. Taking into account this architecture, this paper was written to propose a distributed gateway selection algorithm along with a cross-layer concept which predicts environment dynamics by learning automata. Experimental results demonstrate that the proposed method in various configurations on average improves packet delivery ratio 17.66%, throughput 5.36%, network overhead ratio 6.34%, and average end-to-end delay 15.94% higher than reinforcement learning-based best path routing algorithm (RLBPR), the best studied algorithm; therefore, it leads to more efficient utilization of network capacity compared to nearest gateway, minimum load index, expected transmission count, best path to best gateway, and RLBPR algorithms.

Keywords Gateway selection · Channel assignment · Cross-layer · Learning automata · Multi-radio · Multi-channel · Wireless mesh networks

Mathematics Subject Classification 68T05 (Learning and adaptive systems)

✉ M. Jahanshahi
mjahanshahi@iauctb.ac.ir

Extended author information available on the last page of the article

1 Introduction

A wireless mesh network (*WMN*) consists of mesh routers and mesh clients (Fig. 1). This network is organized in a three-level architecture, including application, infrastructure and Internet. Mesh routers are usually immobile nodes without power constraint that form a multi-hop wireless infrastructure between mesh clients and Internet gateways. Each mesh router not only operates as a host but also as a router, forwards data packets on behalf of other mesh nodes that may not be within direct wireless transmission range of their destinations [1, 2]. The issue of interference reduction in *MRMC WMNs* is typically dealt with by developing a channel assignment strategy which effectively specifies the most appropriate channel-radio associations. However, channel assignment brings about its own complications; in effect, an additional constraint ought to be satisfied for network connectivity in *MCMR WMNs* as compared to the conventional wireless networks; more specifically, two nodes are considered neighbors only if: (1) they are located within the transmission range of each other and; (2) there exists a common channel assigned to the radios of both nodes [3, 4]. To increase the capacity and performance of *WMNs*, mesh routers can be equipped with multiple radios, each tuned on a different channel. Additionally, each mesh router is equipped with a control radio interface to exchange control packets in the network. Using the control radio interface separates the network traffic and also speeds up packet processing in the mesh nodes [5]. Due to the limited number of channels supported by radios, network performance is highly impacted by frequency interferences [6]. Hence, it is necessary to propose an appropriate channel selection scheme to minimize interference and congestion, thereby reducing packet loss and delay.

Gateway nodes are routers with additional features such as higher buffer size, wired and wireless interfaces, etc. Gateway nodes usually are directly connected to a wired network. In many applications, most of the traffic load is conducted to the gateway

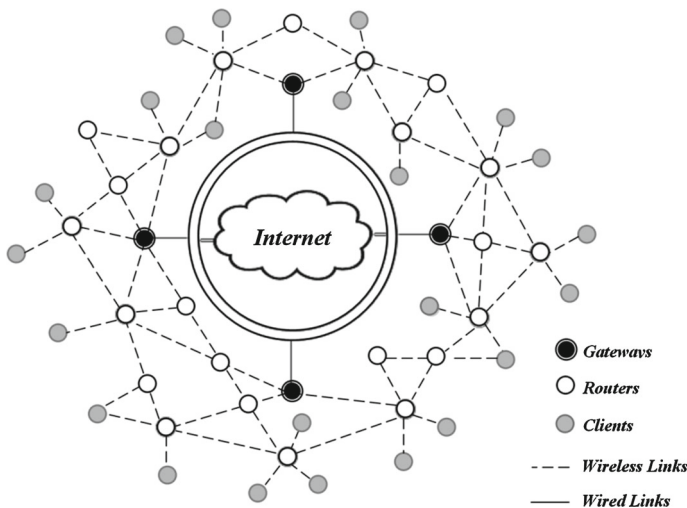


Fig. 1 The infrastructure of *MRMC WMN*

nodes. Therefore, traffic aggregation occurs on paths leading to the gateway nodes and causes congestion. One of the most important considerations is selecting proper gateway in these networks and sending traffic to them [1, 7]. Proper gateway selection has a considerable impact on the WMN efficiency and robustness. In this paper, the proposed LA based gateway selection algorithm (LAGS) selects the gateway nodes based on their performance impact on network traffic and controls which gateway nodes the traffic should be sent to. The proposed method selects the gateway nodes by LA in a distributed fashion on each mesh router. In addition, LAGS selects the proper channel based on LA by determining the most appropriate channel-radio associations. Therefore, LAGS decreases interference and improves efficiency which results in desirable utilization of network capacity. Existence of two approaches, gateway selection in each router on the network layer and channel assignment in each radio on the MAC layer, is according to cross-layer concept. Since the joint gateway selection and channel assignment problem is known as *NP-hard*, use of operation research (OR) methods is not efficient for high dimensions of the given problem in polynomial time order; therefore it is reasonable to apply LA converging to sub-optimal solution.

This paper is organized as follows: In Sect. 2, the prior art gateway selection methods in the wireless networks are surveyed. The theory of LA along with its interaction with the stochastic environment is briefly reviewed in Sect. 3. In Sect. 4, the proposed algorithm for solving gateway selection and channel assignment sub-problems are described in separate phases. In Sect. 5, performance of the proposed method is evaluated by the OMNET simulator and the experimental results are compared with other methods. Section 6 presents the conclusion of the paper.

2 Literature review

To date a number of studies [8–15] have been done on gateway selection in WMNs. However, they have not considered MRMC architecture which is one of the promising methods for improving performance. Therefore, these studies are not compared with the method proposed in this paper. Some studies [16] have just considered the distance to gateway nodes which leads to performance reduction and the traffic load being unbalanced among gateway nodes as some gateway nodes have low load and others experience overload and congestion states. Hop count metric is not a desirable choice as it might be a shorter path between the source node and gateway node that have high interference and high packet loss. In [11] a centralized gateway selection algorithm with a proactive approach in single-radio single-channel (SRSC) multi-hop wireless networks was introduced. In this study, wireless hosts had the same hop count to gateways. Each end host selected its default gateway by a predefined cost function based on load balancing. This study not only considered traffic intensity and load balancing, but also considered wireless channel interference around the gateways. The results of this study indicated that the proposed scheme, by considering two cost functions, increases gateway throughput. Although this scheme used hop count threshold and had limited gateway selection policy to a subset of gateway nodes, it used hop count metric for gateway selection, which may cause performance reduction and the traffic load being unbalanced. Also it is clear that centralized algorithms have a lower perfor-

mance than distributed algorithms due to high network overhead and low scalability and reliability. In some studies, such as [17], gateway switching occurs heuristically and in a centralized fashion; it does not switch to a better gateway and path. Besides, these studies did not utilize learning approaches which consider the previous state of the network environment for proper gateway and path selection. In [18] proposed a link-state and distributed algorithm in *MRMC*, multi-tier wireless mobile ad-hoc networks (*MANET*) with a proactive approach for dynamic gateway selection in each sub-network. Obviously, hop count and interference have a trade-off. This study is trying to compromise between them. To enhance network responsibility in critical situations, gateway nodes are categorized in two classes of active and backup gateways. However, adding, removing or replacing a gateway node disrupts the network until nodes reconfigure successfully and paths converge again. In addition, by adding backup gateway nodes over time, proper sub-optimal solution cannot be achieved. This algorithm uses the network's current situation and a plan for nodes' future shift to calculate a ranked list of gateway nodes allocation to sub-networks. This list has been ranked according to a combination of link costs and congestion costs with inspired by enhanced interior gateway routing protocol (*EIGRP*). On the other hand, the gateways cost is determined in this list for allocating them to sub-networks in the future. Some studies [17, 19, 20] did not take path state between the wireless routers and gateway nodes into account and only considered traffic load. Moreover, [9, 10] suggested gateway load as a metric to select the best gateway in a *SRSC WMN*. The aim is to achieve load balancing among different gateway nodes. Considering the gateway load as the sole metric for selecting the best route is not a good idea. In fact, there are some cases in which the gateway load is low, however, the interference and packet loss on the paths leading to the gateway is high. Some other studies [8] examined a distributed algorithm, including gateway discovery, route selection and gateway selection with a proactive approach in *SRSC WMNs*. Best gateway selection (*BGS*) algorithm presented in this study used a combination of metrics including gateway load and *ETX*. In addition, this study calculated a logical estimate of the interference and link quality in order to select the path to gateway nodes. However, it measured the approximate expected link quality (*ELQ*) just according to *ETX* without considering other effective parameters on the link quality. Expected transmission time (*ETT*) and *ETX* [21, 22] are routing metrics that only consider delivery ratio without concerning about interference among links. It led to load imbalance, because these metrics do not have true perception of traffic load. In [17], a two-layer heterogeneous *MRMC MANET* with a distributed algorithm were proposed as a solution for gateway selection according to current *MLI* metric. In this study, gateway current load was approximated based on previous estimates and processing traffic volume. Besides, this algorithm ensured that the hosts did not switch among the gateway nodes quickly. When a host receives an advertisement from gateway j with lower load, it can adjust gateway j as its new gateway if it uses gateway i at least for a while. Plus, j load index must be at least a predetermined threshold ΔL less than i load index. Moreover, in the aforementioned conditions, a node just switches the gateway nodes with a specific gateway-switching probability to avoid switch fluctuations among gateway nodes. As it was mentioned, although load based approaches, such as *MLI*, reflect gateway load, they do not reflect path states between the mobile hosts and gateways; this is one of the disadvantages of

these approaches. Some studies [6, 23] on gateway selection problem only attended to load balancing and did not consider inter-channel interference and advertisements overhead. In [24], a distributed heuristic algorithm in a *MRMC WMN* was presented to solve the gateway access point selection problem. In this study, a cost function was presented to minimize total traffic among gateways in order to reduce congestion. Plus, to avoid interference, gateway selection and optimization of gateway placement and channel assignment algorithms were evaluated. In this scheme, a simple algorithm was used to select a number of gateway nodes; however, communication patterns in a real network and the method of using selected gateways were not considered. In particular, this study used the hop count metric and the number of under-service hosts for all gateways. These metrics do not consider the interference among different gateways and hosts traffic loads (demands) difference. In [25] a distributed heuristic algorithm with a proactive approach in *MRMC WMNs* was proposed for the best path and gateway selection problem which was called *BP2BG*. This scheme considers gateway load, *ETX* and interference metrics to select the best path in terms of path quality calculation and the best gateway selection with proper load index (*LI*). It not only distributes the traffic load among gateway nodes, but also reduces traffic passing through each gateway. However, due to only considering traffic load and inattention to distance (hop count) from the gateway nodes, it caused reduced total network performance. In some studies such as [23] proposed a distributed meta-heuristic method in *MRMC WMNs* for solving the gateway placement problem and load balancing. This study presented a genetic algorithm to divide *WMN* into limited-to-the radius clusters, under relay load and cluster size constraints. It maintains service quality parameters such as delay and bandwidth. Within each cluster, the cluster head serves as a gateway. A spanning tree rooted at the gateway is used for traffic forwarding. Each node is associated with a tree and can be connected to another tree as an alternative path in case of path failure. This study utilized the local traffic pattern to make a compromise between path length and available bandwidth (λ); as each node sends data only to neighbor gateways within a fixed radius independent of network size. In [6] proposed a distributed meta-heuristic algorithm with a proactive approach in *MRMC WMNs* which is called *RLBPR* problem. The *RLBPR* algorithm learns a proper gateway selection and routing policy depending on various optimization metrics such as packet loss ratio, interference and gateway load. This method was inspired from *BP2BG* [25] and learns the best neighbor and best gateway for sending incoming data packets to them by gateway selection and next hop selection algorithms as well as a learning agent in each router. In [26] presented the conflict graph (*CG*) concept for channel assignment problems in a multi-radio infrastructure and used it to model the interference among mesh routers. In this scheme, a multi-radio conflict graph (*MCG*) forms based on interference among links, then channels were initially assigned to radios greedily via breadth-first search (*BFS*). The search begins with links emanating from the gateway node by giving channel assignment priority to links starting from the gateway and then in decreasing levels of priority to links fanning outward towards the edge of the network. It is the first algorithm that has considered inter-flow interference. Recently, a cross-layer concept optimization framework has been proposed for joint channel assignment and multicast throughput maximization in *MCMR WMNs*. The proposed method is composed of two phases; in the first phase, using Cellular *LA*,

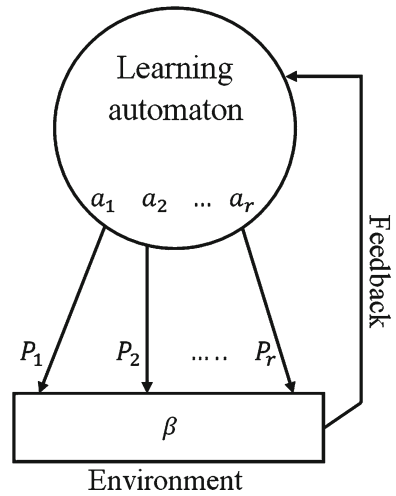
channels were assigned to the links established between the radios of the nodes in a distributed fashion such that the minimal interference coefficient for each link was provided. Then, the resultant channel assignment scheme was utilized in the second phase for throughput maximization within an iterative optimization framework based on Lagrange relaxation and primal problem decomposition [27]. In [28] proposed a distributed load balancing algorithm in *WMNs* to achieve load balancing on gateway nodes which was called *LALB*. This study uses *LA* in order to select the appropriate gateway node to send traffic. The strategy of this paper is to associate nodes with a particular gateway, therefore a domain of nearest gateways will be created for each router. Every time that a router wants to send data, it will choose a gateway from its domain by using its *LA* to send data. In general, the proposed algorithm associates each router with a domain of its nearest gateways. This paper does not utilize routing metrics and just uses the hop count metric for domain creation and gateway selection that is not an exhaustive metric. By using the hop count metric, the routers have to send data packets only to a few particular gateways in their domain; whereas the situation of farther gateways in the moment would be better than gateways inside the domain. It is also possible that a few routers just add a gateway into its domain which its selection probability is always one. In addition, authors in this paper have not considered *MRMC* architecture which is one of the promising methods for improving performance. While in our method, first, each node located inside the broadcasting area, by having multiple radios, receives gateways advertisements. Then, it tries to select the most proper gateway among all gateways in terms of gateway *LI* and routing metrics by its *LA*.

The proposed *LAGS* method presented a distributed meta-heuristic algorithm in a *MRMC WMN*. The *LAGS* performs proper gateway selection on each router and proper channel assignment on each radio interface based on *LA* in a cross-layer manner. Contrary to the existing methods, these two sub-problems will be solved conjointly. The *LAGS* uses a scheme in flooding to reduce network overhead in proactive approach which broadcasts gateway advertisements in the form of a bounded broadcasting area by gateway nodes centrality. The *LAGS* models interference among all neighbor frequency channels and studies the state of the path to the selected gateway.

3 Theory of learning automata (LA)

The main capability of *WMNs* is developing in dynamic and unknown environments resulting in their features to change over time. Changes in many of these features, such as link quality, topology, power consumption, radio resources allocation and traffic patterns, can affect network performance. *LA* can be used in dynamic environments for correct behavior learning and as one of the soft computing tools in various contexts, such as *WMNs*, that lead to substantial improvements in network performance [29]. *LA* is a machine that can perform a finite number of actions. Every selected action is evaluated by a stochastic environment. Then the environment responds to *LA* as a signal. *LA* utilizes this signal and selects its action for the next step. During this process, *LA* learns how to select the appropriate action from among its actions [30]. It does not require prior knowledge of network traffic and does not require complex

Fig. 2 Interaction between LA and environment [29]



analysis of the network during the learning phase. Moreover, each LA needs to keep just one action probability vector (APV) which indicates its lower memory demands. Figure 2 shows the interaction between LA and environment.

The LA can be classified into fixed structure (FSLA) and variable structure (VSLA) groups. In the FSLA, the transition and the output functions are time invariant. On the other hand, In the first group different actions probability is fixed but in the last group, actions probability updates in each iteration of the algorithm [31]. Stochastic learning automata (SLA) can be shown by five tuple $SLA \equiv \{\alpha, \beta, P, L, C\}$ in which $\alpha \equiv \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ is a set of environment inputs, $\beta \equiv \{\beta_1, \beta_2, \dots, \beta_r\}$ is a set of environment outputs, $p \equiv \{p_1, p_2, \dots, p_r\}$ is the APV, $p(t+1) = L[\alpha(t), \beta(t), p(t)]$ is a linear function of the learning algorithm that t is time and $c \equiv \{c_1, c_2, \dots, c_r\}$ is a set of LA output actions penalty probabilities. The main idea of all learning algorithms is: if for the n -th repetition of SLA, a_i action is selected and a desirable signal is received from the environment, then $P_i(t)$ probability increases and other actions probability decreases. In the case of receiving an undesirable signal, $P_i(t)$ probability decreases and other actions probability increases. The general learning algorithm is given by:

Upon receiving desirable signal from the environment:

$$\begin{aligned}
 P_i(t+1) &= P_i(t) + \alpha[1 - P_i(t)] \\
 P_j(t+1) &= (1 - \alpha)P_j(t); \quad \forall j; j \neq i
 \end{aligned}
 \tag{1}$$

Upon receiving undesirable signal from the environment:

$$\begin{aligned}
 P_i(t+1) &= (1 - b)P_i(t) \\
 P_j(t+1) &= \frac{b}{r-1} + (1 - b)P_j(t); \quad \forall j; j \neq i
 \end{aligned}
 \tag{2}$$

In Eqs. (1) and (2), a , b and r denote reward and penalty parameters and the number of actions, respectively. According to different values that are considered for a and b

parameters, different algorithms are achieved. If $a = b$, linear reward-penalty (L_{RP}) algorithm is achieved. If $b \ll a$, linear reward- ϵ -penalty ($L_{R\epsilon P}$) algorithm is obtained. If $b = 0$, algorithm is called linear reward-inaction (L_{RI}) [32, 33].

4 The proposed method

In the proposed method, WMN backbone is modeled as a topology graph $G = (V, E)$. V is a set of mesh routers, $V_r, V_g \in V$, and E is a set of radio links. Each mesh router in V is equipped with multiple radios. Each radio interface can be tuned on an available channel at the same time. In addition, every node in V_r and each radio interface is equipped with one LA . There are gateway nodes in V as V_g which provide connection to the Internet. The problem is how to select the proper gateway and channel assignment based on LA conjointly.

4.1 Preliminaries and assumptions

In what follows, the gateway advertisement packet (GWA) structure and effective metrics in gateway selection, such as LI and airtime, will be discussed in separate sections.

4.1.1 Customized advertisement packet

The proposed algorithm uses the zone creation (ZC), GWA , data, and gateway reply (GWR) packets. By using ZC packet, the broadcasting area is formed and the depth of gateway advertisements sending is limited. The GWA packet is a gateway advertisement that each gateway broadcasts to area routers periodically in defined intervals to reduce network overhead and effort for reducing frequent gateway switching. GWA is applied to gateway discovery in broadcasting area routers and is used as a feedback of gateway and network state. Besides, this packet updates routing metric and traveled path in the routing table of the different routers. GWR packet is a gateway reply in response to receiving the data packet. The GWA packet format of the proposed method is presented in Fig. 3.

GWA fields include: (1) $GWNo.$ field is a gateway number, (2) $GWSeq.No.$ field is the sequence number of n -th advertisement packet sent by this gateway, (3) $LoadIndex_i$ field is LI of gateway i , (4) $AirtimeMetric_i$ field is the best traveled path airtime metric for all GWA s from gateway i to the intended router, (5) $HopMetric_i$ field is the number of hops from gateway i to the intended router, (6) $RouterSeq.No.$ field is the relay router sequence number of GWA , and (7) $RadioNo.$ field is the radio number the GWA is received from.

$GW\ No.$	$GW\ Seq.\ No.$	$Load\ Index_i$	$Airtime\ Metric_i$	$Hop\ Metric_i$	$Router\ Seq.\ No.$	$Radio\ No.$
-----------	-----------------	-----------------	---------------------	-----------------	---------------------	--------------

Fig. 3 Gateway advertisement packet

4.1.2 Effective metrics in the gateway selection

In order to quantify the gateway load amount, the LI parameter is defined by Eq. (3).

$$LI_i = \frac{\sum_{k \in N} \beta k_i \times T_k}{C_i} = \frac{T_i}{C_i} \quad (3)$$

where LI_i , factor βk_i , T_k , T_i , C_i indicates the LI of the gateway i , a fraction of k -th node's traffic that is sent by gateway i -th to the Internet, generated total traffic by node k , the total traffic of gateway i , and the capacity or bandwidth of gateway i Backhaul links, respectively. The LI value is in interval $[0, 1]$ where 1 indicates 100% loaded gateway [28].

The airtime metric estimates the amount of channel resources consumption as a function of loss rate and bandwidth which is defined by Eq. (4).

$$C_a = \left[O_{ca} + O_P + \frac{B_t}{w} \right] \frac{1}{1 - e_{pt}} \quad (4)$$

Here O_{ca} , O_P and B_t denote the channel access overhead, the protocol overhead and the number of bits in a test frame, respectively. Parameters w and e_{pt} are the bit rate in terms of Mbit/s and the frame error for the test frame size B_t , respectively [34].

4.2 LAGS: learning automata based gateway selection algorithm

The proposed method utilizes the proactive flooding mechanism. To reduce overhead, $LAGS$ calculate the depth of the broadcasting area in the form of a zone with gateway nodes centrally. Then mesh routers of this area in the gateway discovery phase insert the discovered gateway nodes to its routing table; thereafter routers by using its LA , select the proper gateway and send data packets on the radio which has received the best selected gateway advertisement from it according to the hop by hop routing protocol with the defined destination. Each radio, upon receiving a data packet with a defined destination selects the proper channel by its LA ; then sends the data packet to the next hop router. The next hop router receives the data packet with the defined gateway destination; then according to its routing table forwards the data packet on one of its radios that has realized which next hop is the most appropriate for this destination. This process repeats in each router until the data packet reaches the considered gateway. Finally, the source router surveys received feedback from the gateway and routing metrics; in case the selected gateway is the proper gateway, the proposed algorithm rewards the gateway by LA , and increases the corresponding act selection probability with the selected gateway in APV for the next step; otherwise it will penalize the corresponding act and reduce its selection probability. In what follows, all the phases will be discussed.

4.2.1 Network connectivity graph (NCG) formation and initial channels assignment phase

While channel assignment is an efficient method for network communication in WMNs, two important issues should be considered at the same time in such a network implementation. The first is to locate the neighboring nodes within the transmission range of each other and make a clean and utilizable connection, and the second is the existence of a common channel assigned to the radios of both nodes [35]. In this phase, at first by transmission range of routers, connectivity links among the same range of routers in the network are established. First, radio interfaces of mesh routers are tuned on different channels temporarily by *BFS* [26]. In fact, the initial channels are assigned to the links. Then *LAGS* algorithm selects the proper channels for radio interfaces by each radio *LA*. Thereafter, the network connectivity graph (*NCG*) is formed and according to it, *MCG* is constructed. The purpose of the channel assignment phase is effort to increase the difference of the same range of channels number based on the number of considered standard channels.

4.2.2 Broadcasting area construction phase

In this phase, by setting up time-to-live (*TTL*) parameter in gateways *ZC* packet with values of 1 to n hops, an area of routers is constructed with gateway nodes centrally. In fact, until the packet is alive, it reaches the special routers and each router by receiving *ZC*, minus 1 unit from the *TTL*, sets its *D* flag to 1 as a sign of membership in the broadcasting area. Of course, 1 and n hops setting for *TTL* is passed up because of preventing the negative effects of the proactive approach (Fig. 4).

LAGS: Initial channel assignment and broadcasting area construction

```

Run BFS_CA algorithm on network topology
Create APV for each radio
Create MCG
For each  $i = \text{gateway } GW_i$  do {
  // First create Advertisement zone for each gateway
  Create GWA packet $i$ ;
  Create ZC packet $i$ ;
  Set TTL for ZC packet $i$ ;
  Broadcast ZC packet $i$  to NCG
  For each  $j = \text{router } R_j$  do {
    Receive ZC packet $i$ ;
    Create routing table $j$ 
    If  $TTL \neq 0$  then
       $TTL = TTL - 1$ 
      Set  $D_{flag} = 1$ 
      Rebroadcast ZC packet $i$ (TTL)
    Else
      Expire ZC packet $i$ 
      Set  $D_{flag} = 0$  }
  Create  $Zone_{GW_i}$  // Subset of routers which recieved the ZC packet $i$  }
  Create Advertisement zone =  $\sum_{i \in GWs} Zone_{GW_i}$  // Joint all  $Zone_{GW_i}$ 

```

Algorithm. 1. *LAGS* algorithm pseudo code, the initial channel assignment and broadcasting area construction

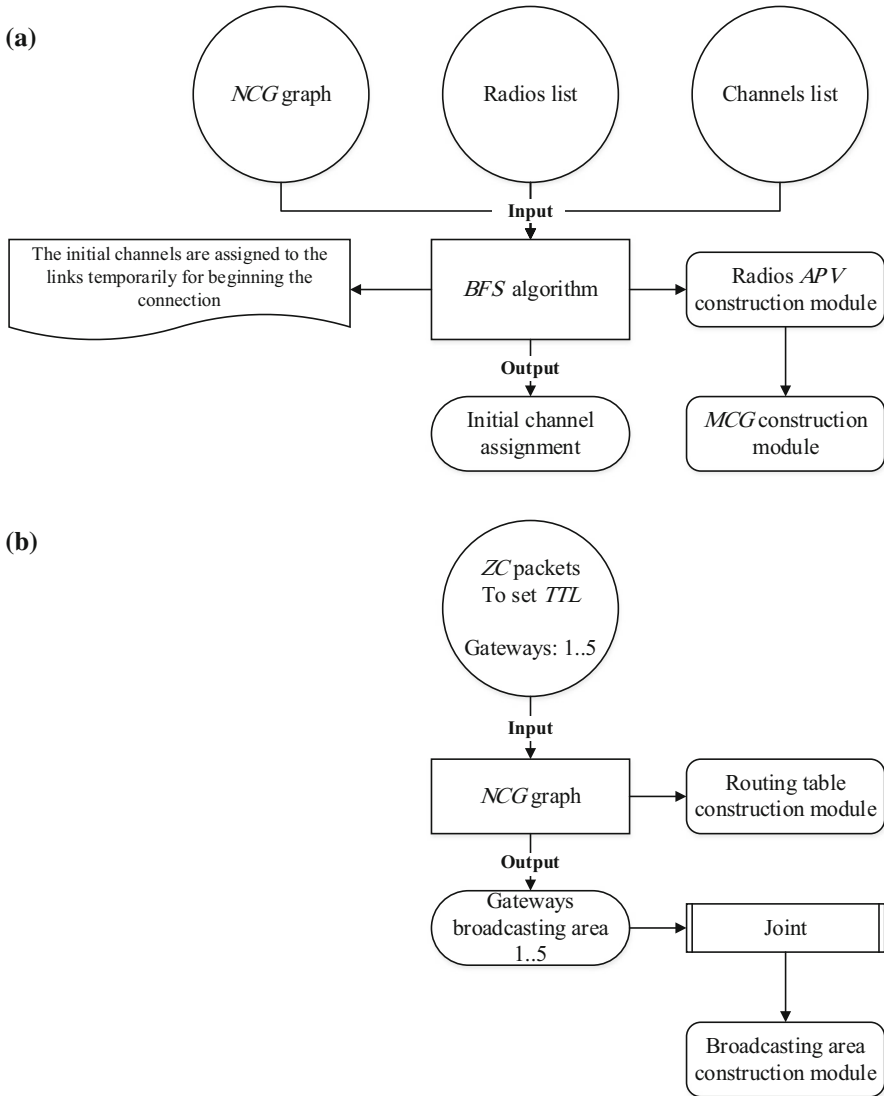


Fig. 4 Block diagram of the *LAGS* algorithm. **a** The initial channel assignment and **b** broadcasting area construction

4.2.3 Gateway discovery phase in broadcasting area routers

In this phase, each gateway node broadcasts the *GWA* packet in the created area periodically every t seconds. The *GWA* includes updated information on the gateway and route states. *GWA* fields are filled by the intended gateway as follows: *GWNo.* field equal to its gateway number, *GWSeq.No.* equal to *GWA* sequence number, *LoadIndex_i* field equal to its gateway *LI*, and *AirtimeMetric_i*, *HopMetric_i*, *RouterSeq.No.*, *RadioNo.* fields equal to zero.

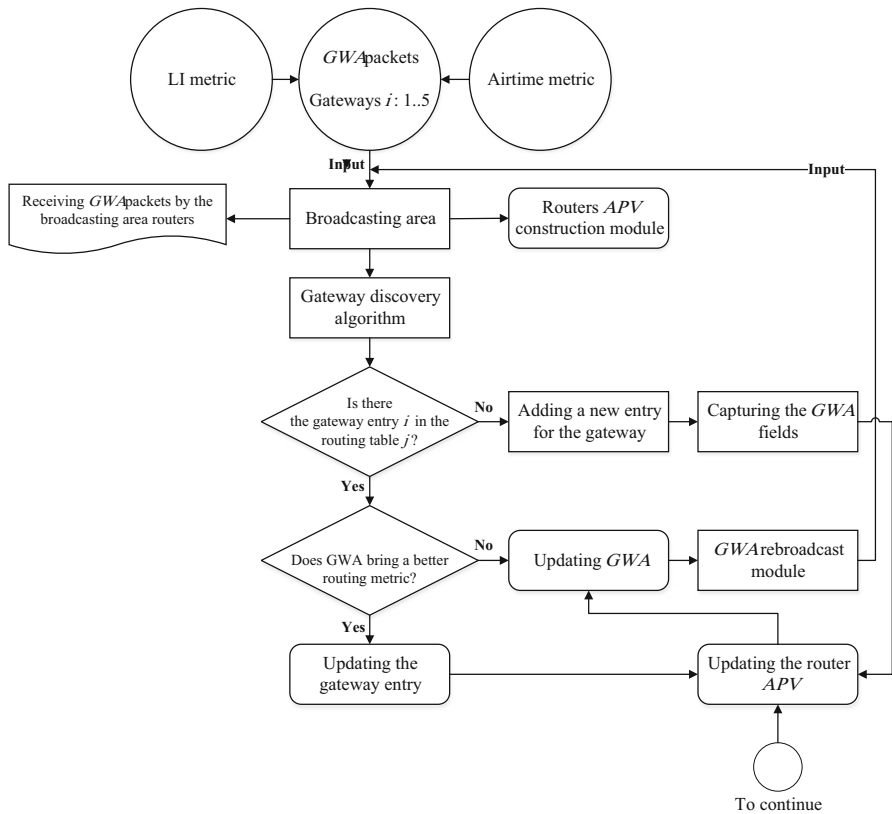


Fig. 5 Block diagram of the gateway discovery phase

Thereafter, the broadcasting area routers create an entry in the routing table and store the necessary information if there is not any entry for the intended gateway after receiving the *GWA*. However, in the case of existing entries, if the received *GWA* is from a different neighboring router or brings a better routing metric, the routers update the intended gateway entry. Then each router updates the *GWA* fields of each gateway and rebroadcasts the updated *GWAs* to the neighboring routers within the area. Subsequently, the *LA* of each router that has received a *GWA* forms its *APV* by the number of discovered gateways and determines the initial probability of each gateway’s corresponding action in this vector. In the proposed method, the initial probability of all actions in *APV* is equal (Fig. 5).

4.2.4 Gateway selection phase

Once all routers located in the broadcasting area receive a *GWA* from each gateway, gateway discovery is performed. Then the *APV* of each area’s router is formed. Every time data exists in a router for sending, the proposed algorithm selects the

proper gateway through generating a random number in the interval [0, 1] and matches it with the *APV* members' probability. Then the router sends the data packets that contain the source node and destination gateway fields according to its routing table to the best next hop (the best radio). This next hop has been recognized based on better airtime metric when receiving the selected gateway's *GWAs*. At the next hop, the forwarding operation is repeated again according to the routing table until the data packet reaches the intended destination gateway. Then the gateway sends the *GWR* reply packet to the source router through the incoming path unicast-based. Each router upon receiving a *GWA* from the gateway at a specific time interval, as feedback from the network, updates its *APV* and routing table by the learning algorithm. When the selected gateway is proper its selection probability in *APV* increases based on the learning algorithm; otherwise, its selection probability decreases. Therefore, over time the proper gateway selection probability increases and the probability of other gateways selection in *APV* correspondingly decreases. Each router upon receiving a *GWA* that contains gateway *LI* and airtime metrics, as feedback from the gateway and network states, rewards and penalizes the selected gateway node's corresponding action by the learning algorithm as follows.

Whenever gateway *LI* (as $\beta_i(t)$) is less than P_1 of the average *LI* of other gateways within *APV*, it is rewarded by 3α parameter (desirable signal). Reward and penalty parameters are calculated by Eqs. (5) and (6). If the *LI* is more than P_1 and less than P_2 of the average *LI* of other gateways, the selected action is rewarded by Eq. (7). If the *LI* is more than P_2 and less than 100% of the average *LI* of other gateways, the selected action is penalized by Eq. (8) (undesirable signal). Finally, if *LI* is more than 100% of the average *LI* of other gateways, the selected action is penalized by $3b$ (Fig. 6).

$$a = \delta + \frac{LI_i + \frac{(MaxAirtime - Airtime_i) - L}{U - L}}{MaxLI + \frac{MaxAirtime - L}{U - L}} \rightarrow \theta \tag{5}$$

$$b = \delta + \frac{(AvgLI - LI_i) + \frac{Airtime_i - L}{U - L}}{AvgLI + \frac{MaxAirtime - L}{U - L}} \rightarrow \theta \tag{6}$$

$$V_{Reward} = \left(1 + \frac{1 - \frac{LI_i}{AvgLI}}{(p_2 - p_1)} \right) \alpha \tag{7}$$

$$V_{Penalty} = \left(1 + \frac{\frac{LI_i}{AvgLI} - 1}{(1 - p_2)} \right) b \tag{8}$$

The LI_i parameter refers to the *LI* in the gateway node that a *GWA* is received from. $Airtime_i$ is the calculated airtime metric from gateway *i* to the router. *MaxAirtime* is the maximum value of airtime routing metric from the *GWA* receiving router to the furthest gateway. By calculated hop count metric in each *GWA*, the *MaxAirtime* is calculated. *AvgLI* is the average *LI* of other discovered gateway nodes within the routing table of the router. *MaxLI* is the highest *LI* among dis-

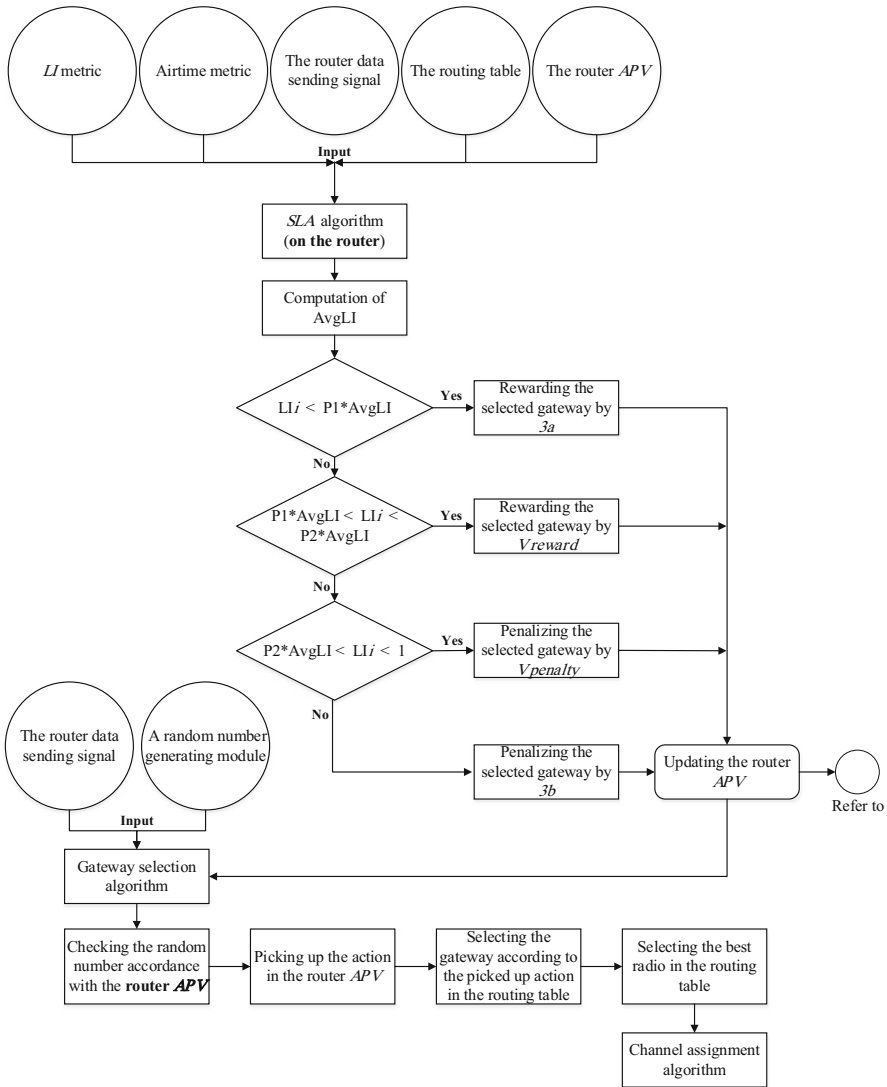


Fig. 6 Block diagram of the gateway selection phase

covered gateway nodes for the router. The rationale behind the use of δ and θ is intuitive. δ parameter indicate the minimum acceptable values for reward (a) and penalty (b) parameters, respectively. θ parameter are selected as the reward (a) and penalty (b) parameters' value is not greater than a certain threshold. LI and $Airtime$ metrics have different measurement units; therefore, L and U parameters are defined for the alignment of these metrics measurement units. The L and U parameters are lower and upper limits of the considered metric, respectively.

LAGS: Gateway discovery and gateway selection

```

For Schedule  $t = 10s$  do{
  For each  $i = \text{gateway } GW_i$  do {
    Calculate  $Ll_i = \frac{\sum_{k \in N} Bk_i \times T_k}{C_i}$  //described in section 4.1.2
    Broadcast GWA packet $_i$ (fields)to each router  $R_j \in$  advertisement zone

    For each  $j = \text{Router } R_j \in$  advertisement zone do { // if  $D_{flag} = 1$ 
      Receive GWA packet $_i$ (fields) //from all  $GW_i$ ;
      Calculate  $\text{airtime}_i = \left[ O_{ca} + O_p + \frac{Bt}{w} \right] \frac{1}{1 - e^{-pt}}$  //discribed in section 4.1.2
      Create APV Router  $R_j$ 
      If gateway entry $_i$  exists in routingtable $_j$  then
        If gateway entry $_i$ (Radio No.)  $\neq$  GWA packet $_i$ (Radio No.) or
           $-\text{gateway entry}_i(\text{Airtime Metric}_i) < \text{GWA packet}_i(\text{Airtime Metric}_i)$  then
            Update gateway entry $_i$ (GWA packet $_i$ (fields)) in routing table $_j$ 
          Else
            Add gateway entry $_i$  to routing table $_j$ 
            Insert GWA packet $_i$ (fields) to gateway entry $_i$  }
            Insert discovered gateway ( $GW_i$ ) to APV Router  $R_j$  //Base on Hop count
          };
          //LA algorithm on the router:
          If  $Ll_i < P_1 \text{Avg}Ll$  then //  $P_1, P_2$  are the percentages of the average  $Ll$  of other gateways
            Reward the selected gateway by  $3a$ 
          If  $P_1 \text{Avg}Ll < Ll_i < P_2 \text{Avg}Ll$  then
            Reward the selected gateway by  $V_{\text{Reward}}$ 
          If  $P_2 \text{Avg}Ll < Ll_i < 1$  then
            Penalize the selected gateway by  $V_{\text{Penalty}}$ 
          If  $Ll_i > 1$  then
            Penalize the selected gateway by  $3b$ 
          Update APV Router  $R_j$ 
          Update GWA packet $_i$ 
          Rebroadcast GWA packet $_i$  to advertisement zone }
          //gateway selection algorithm
          If data packet  $R_j$  (ready to send) in router  $R_j$  then
            Generate random number in the interval  $[0, 1]$ 
            Check member of APV Router  $R_j$ 
            Select GW in APV Router  $R_j$ ;
            Select selected gateway radio  $I_j$  in routing table $_j$  //Base on Airtime
            Send data packet  $R_j$  to radio  $I_j$ ;
          }
    }
  }

```

Algorithm. 2. LAGS algorithm pseudo code, gateway discovery and gateway selection**4.2.5 Channel assignment phase**

Channel assignment in *MRMC WMNs* is the assignment of channels or radio frequencies to radio interfaces of each mesh router, as it uses most radio frequencies efficiently and minimizes the radio frequencies' interference. By considering the dynamic states of *WMNs*, it seems impossible to access some quite non-overlapping (orthogonal) channels constantly. Therefore, whatever the rate of variation and convergence towards non-overlapping channels is higher, the algorithm will be more efficient. To achieve improved efficiency, the proposed method uses *LA* in each radio for proper channel selection. The main advantage of *LA* in this algorithm is high flexibility in reaching sub-optimal solutions. *LAGS* algorithm selects channels with the least interference by considering the aforementioned cases and using the local information of each node.

As a result, inter-channel interference and end-to-end delay reduces significantly and *PDR* increases. In *IEEE* 802.11g, there are 13 available channels among which four channels are orthogonal. Channels that have less than four-channel separation with only 20 MHz separation are referred to as partially overlapped channels and could be very useful for interference reduction. Efficient utilization of partially overlapping channels allows significant enhancement in parallel transmissions and overall network throughput [36]. To formulate the channel assignment sub-problem u , $I(u)$, R_T , R_I , $d(u, v)$, e_i , k , e_k , and $Int(e_i)$ denote the mesh router, number of mesh routers radios, transmission range, interference range, distance between u and v nodes, selected channel on the intended radio, number of interferer links with the intended link selected channel in *MCG*, interferer channels (overlapping channels) number with selected channel, and the most interference percentage on the selected channel e_i , respectively. *MCG* is utilized to check inter-channel interference. Channels interference percentage is calculated according to the channels correlation coefficient formulation in the first part of Algorithm 3 (Fig. 7).

<i>Channels correlation coefficient and channel assignment</i>
<pre> // Channel assignment algorithm: e_i = the selected channel on the intended radio For each e_k = 1 to k // In the conflict graph Array[1..k] = (e_i - e_k) Int(e_i) = min(Array[1..k]) convert to percentage Check inter - channel interference in MCG //LA algorithm on the radio: if Int(e_i) ≥ 0.4 then Reward the selected channel base on Int(e_i); Else Penalize the selected channel base on Int(e_i); Update APV radio I_j Generate random number in the interval [0, 1] Check member of APV radio I_j Select Channel in APV radio I_j Send data packet R_j to next hop router //Data sending </pre>

Algorithm. 3. Pseudo code of channels correlation coefficient and proper channel selection

MCG shows some links have interference with the intended radio link. The correlation between any two channels is known for all possible channel separations, as depicted in Fig. 8. For example, the separation between channels 1 and 3 is 2. Thus, the correlation coefficient between them equals 0.5267 [37]. The separation between each interferer link channel number and the intended radio selected channel number in *MCG* is calculated. Based on the minimum difference value, the correlation coefficient in the selected channel and same range neighbor channels is determined. Then the interferer link with the least difference value is considered for determining reward and penalty. Selected initial probability for each channel in *APV* is considered equal. The learning algorithm rewards and penalizes selected channel action in *APV* according to Fig. 8. If the channel separation between two interferer links is 2 and less, the selected channel action is penalized; and if the channel separation equals 3 and more, the selected channel action is rewarded. There is not any inter-channel interference for the channel separation 4 and 5.

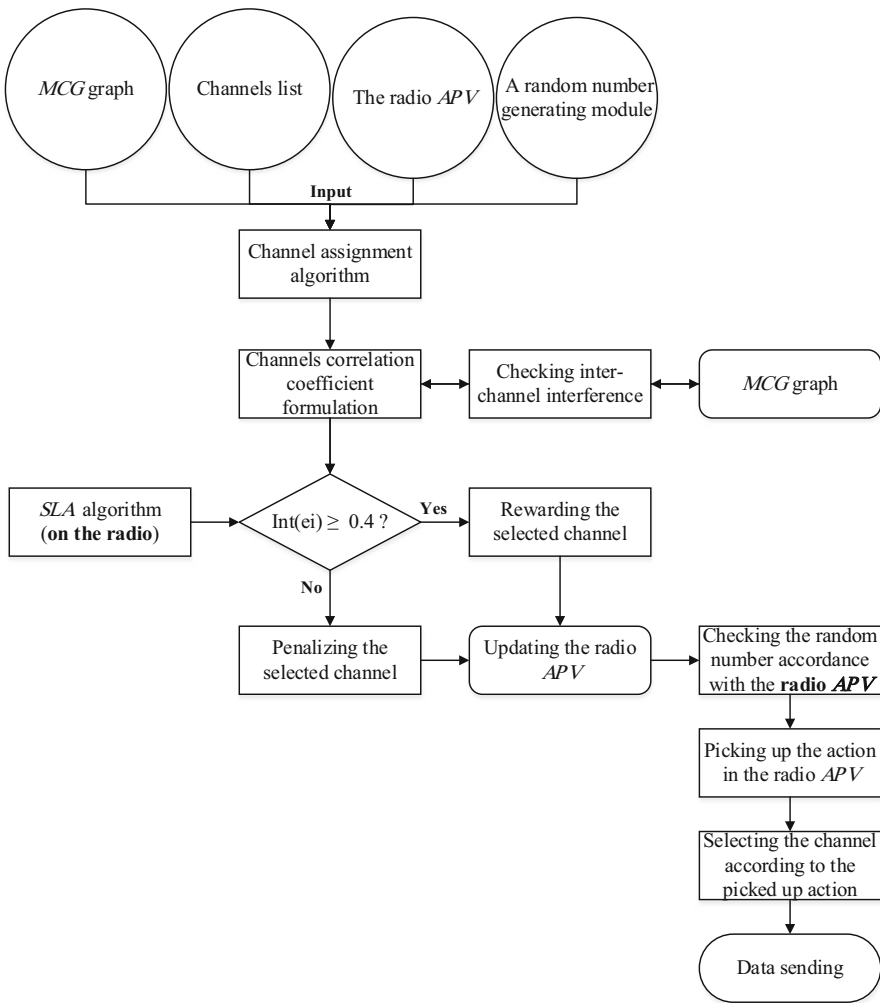


Fig. 7 Block diagram of the channels correlation coefficient and proper channel selection

Each channel is penalized to the value of the determined interference percentage and this penalty is apportioned among all other channels as a reward. For rewarding, the selected channel probability is deduced from the other channels probability and added to the selected channel. Channels with more probability receive more reward. Reward and penalty is calculated by Eqs. (9) and (10) in which b denotes the interference percentage.

$$\begin{aligned}
 P_i(t + 1) &= P_i(t) + a \cdot \sum_{i \notin \{1 \leq k \leq 13\}} P_k(t) \\
 P_j(t + 1) &= (1 - a)P_j(t) \quad \forall j, j \neq i \quad 0 < a < 1
 \end{aligned}
 \tag{9}$$

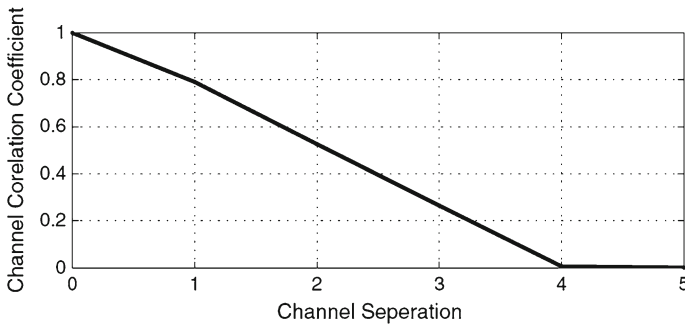


Fig. 8 Channel correlation coefficient versus channel separation for the thirteen-channel 802.11g [37]

$$\begin{aligned}
 P_i(t+1) &= (1-b)P_i(t) \\
 P_j(t+1) &= P_j + \frac{(1-b)P_i(t)P_j(t)}{\sum_{i \notin \{1 \leq k \leq 13\}} P_k(t)} \quad \forall j, j \neq i \quad 0 < b < 1 \quad (10)
 \end{aligned}$$

5 Performance evaluation

In this study, the *OMNET++* [38] open source simulator and *INETMANET* framework were used to evaluate the performance of the proposed *LAGS* algorithm. The obtained results compared with other protocols, such as *NGW* [24, 34, 39, 40], *ETX* [8, 34], *BP2BG* [25], *MLI* [6, 25, 39], and *RLBPR* [6]. Experimental results demonstrate that *LAGS* improved *PDR*, throughput, and efficient utilization of network capacity compared with the aforementioned protocols. Also *LAGS* provided a smaller number of *GWA* which led to lower overhead, resulting in average end-to-end delay decreases in network. *PDR* is the ratio of data packets being successfully received by the mesh gateways versus data packets being sent by the all mesh routers. Network throughput represents the total number of bits that are correctly sent by each one of nodes in the *WMN* during a second in kilo bit per second (*Kbps*). Control messages overhead is the ratio of bytes sent by the mesh routers as control messages versus all bytes sent by the mesh routers which contain data packets and control messages. Average end-to-end delay represents the average delay for sending constant bit rate (*CBR*) packets from source nodes to *GWs*.

Table 1 shows the simulation environment initial settings. This study attempted to apply identical settings and the same topology for all protocols in experiments to metrics measurement, so as to make the comparison quite fair. No hypothesis was considered on the network topology and all distributions were normal. The *LA* type used in this paper was *L_{REP}* and the model of automata interacting environment was the *S* model which sent the reinforcement signal as a random variable in the interval [0, 1]. Empirically, in the learning algorithm in Eqs. (5) and (6) the δ and θ parameters were considered 0.1 and 0.15 [28], and P_1 and P_2 in Eqs. (7) and (8) have been set to 0.45 and 0.75, respectively.

Table 1 The *LAGS* simulation parameters

Simulation parameters	Values
Simulation time	600 s
Network area	1000 × 1000 m ²
MAC layer protocol	802.11g
Data rate	Up to 54 Mbps
Number of each radio's channels	13 channels
Radio interfaces	3 radios
Traffic type	CBR (UDP)
Packet size	1000 Bytes
Number of mesh nodes	10–50
Number of gateway nodes	1–5
Advertisement interval	10 s
Gateways position	Random (fixed for all scenarios)
Routers position	Random (fixed for all scenarios)
Advertisement zone (TTL)	4 hops
Gateway-internet link	100 Mbps wired

Experimentally, in the present study the depth of the broadcasting area is considered 4 hops as *TTL* parameter; that is, different *TTL* were set, and *TTL*=4 indicated the best results. We choose *IEEE* 802.11g for our simulations since it is widely accepted radio technique and is commonly used. Besides, it has longer transmission range for WMNs (compared to *IEEE* 802.11a/b) [5, 36, 41–44].

In Fig. 9 *PDR* versus the number of mesh routers is compared for the given algorithms. It is evident that the *LAGS* algorithm increased *PDR* 17.66% more than *RLBPR*, the best studied algorithm. This result indicates that *LAGS* leads to low collision in radio channels and proper routing between routers and gateways. The main reason for this improvement is utilizing *LA* to select one of the gateway nodes to receive data from the router. The *LAGS* mechanism selecting the most appropriate gateway led to decreasing control messages' shift distance and the number of control messages.

In Fig. 10, the volume of control messages that are necessary to be passed between gateways and routers in order to different algorithms can present their intended service are measured. Low control messages overhead enables the algorithm to use network capacities properly; therefore, network efficiency to data transfer increases. It is clear that one of the major challenges is to achieve a suitable balance between control messages' volume and providing the appropriate service in the network. This study aimed to address this problem, and it was successful. *LAGS* could obtain the lowest control messages overhead for each number of gateway nodes; i.e., it improved control messages overhead 6.34% more than *RLBPR* which was better than the others for reducing network overhead.

In Fig. 11, the effect of routers' number on throughput among different algorithms is compared. In a definite traffic load, each router's throughput is affected by the number of waiting packets in the queue and neighbors' readiness for receiving packets from

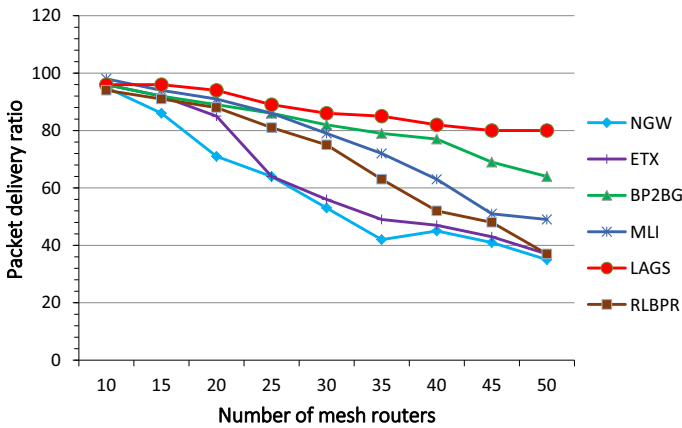


Fig. 9 PDR versus number of mesh routers

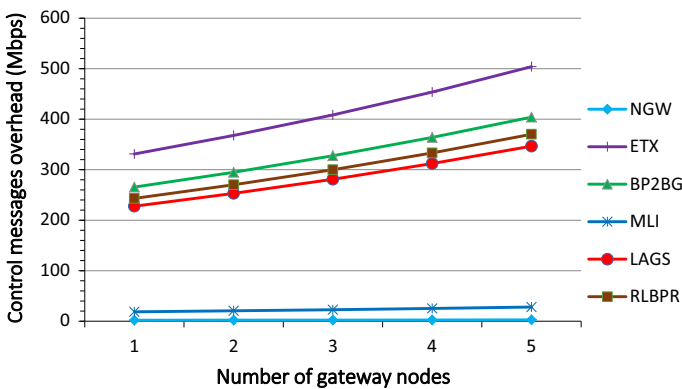


Fig. 10 Control messages overhead versus the number of gateway nodes

that router. The upper limit of this metric is equal to the throughput of each router which can change by its neighbors' numbers and radio interference rate. The *LAGS* algorithm is able to provide a higher throughput for router nodes in spite of using higher control overhead compared to some other algorithms. This was due to proper routes' selection and load balancing among different gateways and thus using the maximum capacity of router nodes. Dynamically proper channel selection by *LA* caused channels-switching diversity. It has a positive impact on the likely collision rate, resulting in an increase in the routers' total throughput. As the graphs show, *LAGS* not only increase throughput 5.36% more than *RLBPR*, but also has a significant difference with other discussed algorithms. A general rule in *WMNs* demonstrates that by increasing the number of mesh routers to 50 and more, the traffic and the network load would be increase, therefore environmental conditions of the network would lead to decrease in *PDR* and throughput which is shown in Figs. 9 and 11.

In Fig. 12, the average delay versus network traffic load rate is studied. The growth rate of received packets' average delay in a router in *LAGS* was lower than other algorithms. This reduction was 15.34% more than *RLBPR*, the best one of the former

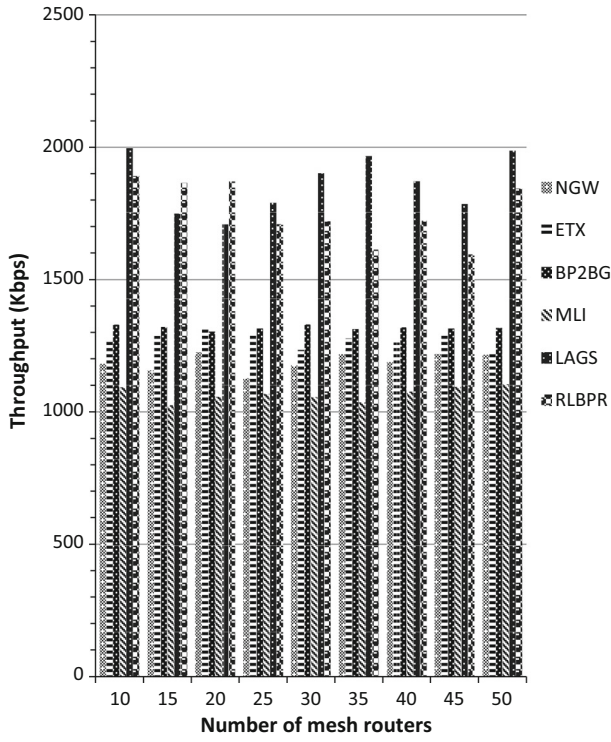


Fig. 11 Throughput versus the number of mesh routers

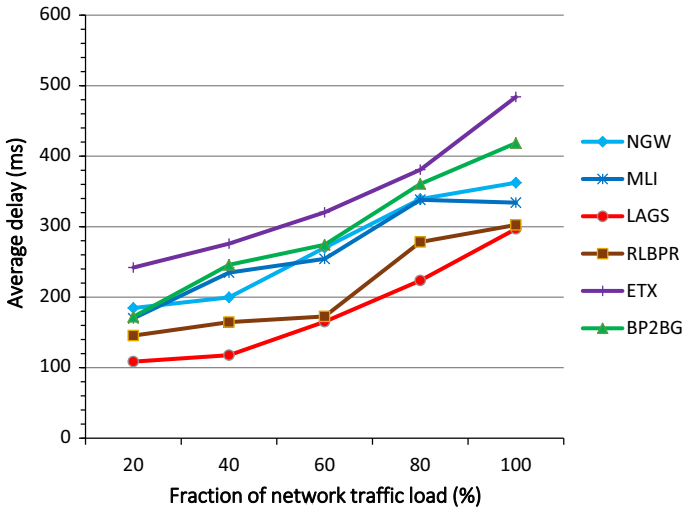


Fig. 12 Average delay versus traffic load rate

algorithms. In the *LAGS* algorithm, the dynamic manner of *LA* in paths switching was one of the effective factors on delay reduction. It led to load balancing among routers and thus more ability to conduct its outgoing traffic to gateway nodes.

The time complexity of the *LAGS* was analyzed through studying the *LA* convergence, control messages overhead, and the distributed algorithm behavior on the network scale. Since *LAGS* acts in an iterative manner, and the iterations continue until achieving a sub-optimal solution, and also because of low convergence rate of *LA*, the time complexity of proposed method is high; however, compare to the involved methods, provides lower time complexity by declining control messages overhead, enhancing *PDR* and throughput, and balancing the load between the different gateways.

6 Conclusion

In this paper, proper gateway selection and proper channel assignment in *MRMC WMNs* were studied which are known as *NP-hard* problems. *LA* has a lot of applications in developing non-deterministic complex algorithms and to solve different kinds of *NP* problems. The proposed method utilizes *LA* in two phases, proper gateway selection and proper channel assignment. According to the experimental result, the proposed algorithm is able to improve *PDR* and network routers throughput more than other algorithms. Using the *LAGS* algorithm leads to improvements in *PDR* by decreasing inter-channel interference and, as a result, decreased traffic congestion and end-to-end delay. Moreover, utilizing *LA* in gateway selection and channel assignment algorithms leads to increased network flexibility against traffic changes outcomes and even likely topological changes; hence, the network can adapt itself to the new environment in any situation. Summing all these contents together indicate that *LAGS* as a cross-layer and distributed algorithm can increase performance and decrease interference simultaneously, thereby can be utilized as a base method for performance evaluation of different network layers.

References

1. Hossain E, Leung KK (2008) *Wireless mesh networks: architectures and protocols*. Springer, Berlin, p 113
2. Akyildiz IF, Wang X, Wang W (2005) *Wireless mesh networks: a survey*. *Comput Netw* 47(4):445–487
3. Jahanshahi M, Dehghan M, Meybodi MR (2013) On channel assignment and multicast routing in multi-channel multi-radio wireless mesh networks. *Int J Ad Hoc Ubiquitous Comput* 12(4):225–244
4. Jahanshahi M, Dehghan M, Meybodi MR (2011) A mathematical formulation for joint channel assignment and multicast routing in multi-channel multi-radio wireless mesh networks. *J Netw Comput Appl* 34(6):1869–1882
5. Ramezani M, Jahanshahi M (2017) Load-aware multicast routing in multi-radio wireless mesh networks using FCA-CMAC neural network. *Computing* 1–29
6. Boushaba M, Hafid A, Belbekkouche A (2011) Reinforcement learning-based best path to best gateway scheme for wireless mesh networks. In: 7th International conference on wireless and mobile computing, networking and communications, WiMob. IEEE, pp 373–379
7. Akyildiz IF, Wang X (2005) A survey on wireless mesh networks. *Commun Mag* 43(9):S23–S30

8. Ashraf U, Abdellatif S, Juanole G (2009) Gateway selection in backbone wireless mesh networks. In: Wireless communications and networking conference, WCNC. IEEE, pp 1–6
9. Kim Y, et al (2007) Load-balanced mesh portal selection in wireless mesh network. In: Military communications conference, MILCOM. IEEE, pp 1–6
10. Kumar R, Misra M, Sarje AK (2007) An efficient gateway discovery in ad hoc networks for internet connectivity. In: International conference on computational intelligence and multimedia applications. IEEE, pp 275–282
11. Tada K, Yamamoto M (2009) Load-balancing gateway selection method in multi-hop wireless networks. In: Global telecommunications conference, GLOBECOM. IEEE, pp 1–6
12. Iqbal SMA, Kabir MH (2011) Hybrid scheme for discovering and selecting internet gateway in mobile ad hoc network. *Int J Wirel Mob Netw IJWMN* 3(4):83–101
13. Lakshmanan S, Sivakumar R, Sundaresan K (2009) Multi-gateway association in wireless mesh networks. *Ad Hoc Netw* 7(3):622–637
14. Kongara H, Kondareddy YR, Agrawal P (2009) Fairness and gateway classification algorithm (GCA) in multihop wireless mesh networks. In: 41st Southeastern symposium on system theory, SSST. IEEE, pp 77–81
15. Liu W, et al (2012) A novel gateway selection method to maximize the system throughput of wireless mesh network deployed in disaster areas. In: 23rd International symposium on personal indoor and mobile radio communications, PIMRC. IEEE, pp 771–776
16. Sun Y, Belding-Royer EM, Perkins CE (2002) Internet connectivity for ad hoc mobile networks. *Int J Wirel Inf Netw* 9(2):75–88
17. Huang C-F, Lee HW, Tseng YC (2004) A two-tier heterogeneous mobile ad hoc network architecture and its load-balance routing problem. *Mob Netw Appl* 9(4):379–391
18. Gottlieb YM, et al (2009) Gateway selection in multitiered mobile ad hoc networks. In: Military communications conference, MILCOM. IEEE, pp 1–7
19. Ahn S, et al (2005) Load balancing in MANET with multiple internet gateways. Draft-ahn-manet-multigateway-00. MANET working group, pp 1–10
20. Kim Y, Ahn S, Lee J (2007) Load-balancing proactive internet gateway selection in MANET. Draft-kim-autoconf-gatewaysel-01. MANET autoconfiguration, AUTOCONF, pp 1–11
21. Draves R, Padhye J, Zill B (2004) Routing in multi-radio, multi-hop wireless mesh networks. In: Proceedings of the 10th annual international conference on mobile computing and networking, MobiCom. ACM, pp 114–128
22. De Couto DSJ et al (2005) A high-throughput path metric for multi-hop wireless routing. *Wirel Netw* 11(4):419–434
23. Talay AC (2007) A gateway access-point selection problem and traffic balancing in wireless mesh networks. In: Applications of evolutionary computing. Springer, pp 161–168
24. Tajima S, et al (2006) An internet gateway access-point selection problem for wireless infrastructure mesh networks. In: 7th International conference on mobile data management, MDM. IEEE, p 112
25. Boushaba M, Hafid A (2011) Best path to best gateway scheme for multichannel multi-interface wireless mesh networks. In: Wireless communications and networking conference, WCNC. IEEE, pp 689–694
26. Ramachandran KN, et al (2006) Interference-aware channel assignment in multi-radio wireless mesh networks. In: Infocom, pp 1–12
27. Jahanshahi M, Dehghan M, Meybodi MR (2017) A cross-layer optimization framework for joint channel assignment and multicast routing in multi-channel multi-radio wireless mesh networks. *Int J Comput Math* 93(11):1–41
28. Kashanaki M, Beheshti Z, Meybodi MR (2012) A distributed learning automata based gateway load balancing algorithm in wireless mesh networks. In: International symposium on instrumentation and measurement, sensor network and automation (IMSNA). IEEE, pp 90–94
29. Nicopolitidis P et al (2011) Adaptive wireless networks using learning automata. *Wirel Commun* 18(2):75–81
30. Narendra KS, Thathachar M (1974) Learning automata-a survey. *IEEE Trans Syst Man Cybern* 4(4):323–334
31. Rasaneh S, Jahanshahi M (2017) A QoS aware learning automata based channel assignment method in cognitive network. *Wirel Pers Commun* 97(1):495–519
32. Jahanshahi M, Dehghan M, Meybodi MR (2013) LAMR: learning automata based multicast routing protocol for multi-channel multi-radio wireless mesh networks. *Appl Intell* 38(1):58–77

33. Jahanshahi M, Meybodi MR, Dehghan M (2009) A new approach for task scheduling in distributed systems using learning automata. In: IEEE international conference on automation and logistics, ICAL '09
34. Akyildiz I, Wang X (2009) Wireless mesh networks, vol 3. Wiley, Hoboken
35. Jahanshahi M, Barmi AT (2014) Multicast routing protocols in wireless mesh networks: a survey. *Computing* 96(11):1029–1057
36. Behraves R, Jahanshahi M (2017) Interference-aware and cluster based multicast routing in multi-radio multi-channel wireless mesh networks. *J Comput Robot* 10(1):21–30
37. Karimi OB, Liu J, Li Z (2014) Multicast in multi-channel wireless mesh networks. In: Lecture notes in computer science, LNCS (6091), HAL Id: hal-01056316. Springer, pp 148–159
38. OMNet++ discrete event simulation system. <https://www.omnetpp.org/>. Accessed Aug 2017
39. Ma L, Denko MK (2007) A routing metric for load-balancing in wireless mesh networks. In: 21st International conference on advanced information networking and applications workshops, AINAW. IEEE, pp 409–414
40. De Couto DSJ et al (2003) Performance of multihop wireless networks: shortest path is not enough. *ACM SIGCOMM Comput Commun Rev* 33(1):83–88
41. Wang J et al (2015) Partially overlapped channel assignment for multi-channel multi-radio wireless mesh networks. *EURASIP J Wirel Commun Netw* 2015(1):25
42. Shahmirzadi MA, Dehghan M, Ghasemi A (2018) An optimization framework for multicasting in MCMR wireless mesh network with partially overlapping channels. *Wirel Netw* 24(4):1099–1117
43. Behraves R, Jahanshahi M (2016) a novel multicast tree construction algorithm for multi-radio multi-channel wireless mesh networks. *J Comput Robot* 9(2):33–41
44. Bozorgchenani A, Jahanshahi M, Tarchi D (2018) Gateway selection and clustering in multi-interface wireless mesh networks considering network reliability and traffic. *Trans Emerg Telecommun Technol* 29(3):e3215

Affiliations

A. R. Parvanak¹ · M. Jahanshahi² · M. Dehghan³

A. R. Parvanak
a.parvanak@gmail.com

M. Dehghan
dehghan@aut.ac.ir

¹ Young Researchers and Elite Club, Qazvin Branch, Islamic Azad University, Qazvin, Iran

² Young Researchers and Elite Club, Central Tehran Branch, Islamic Azad University, Tehran, Iran

³ Department of Computer Engineering, Amirkabir University of Technology, Tehran, Iran