A New Unicast Routing Algorithm for Load Balancing in Multi-Gateway Wireless Mesh Networks

Mohammad Karim Jafaryan¹, Mohammad Ali Jabraeli Jamali², Shahin Akbarpour³

¹Faculty of Engineering, Department of Computer, East Azarbaijan Science and Research Branch, Islamic Azad University, Tabriz, Iran
², ³Department of Computer, Shabestar Branch, Islamic Azad University, Shabestar, Iran

E-mail: ¹mohammad.jafaryan@gmail.com, ²m_jamali@csri.ac.ir, ³sh_akbarpour@csir.ac.ir

ABSTRACT

Gateways are a very important element of wireless mesh networks. They provide the connection between the whole network and the outside world, including the Internet. Since all of input and output traffic in the network go through gateways, these points are simply prone to congestion. Despite having multiple gateways in the network, in the absence of load balancing in the network, some of these gateways are still in danger of congestion that causes traffic aggregation in gateways, and hence, degrading of network throughput. In this paper we propose a method named LBMGW for selecting appropriate gateways that aims load balancing in gateways and increasing network performance. LBMG is the measure for selecting a gateway which is a combination of measures such as negative percent hop, time, and internal load in each gateway. Furthermore, some conditions have been considered to prevent control overhead inside the network. GloMoSim has been used for simulations. Then the results are compared with other available methods that show improvement in throughput of wireless mesh networks with the proposed method.

Keywords: Wireless Mesh Network, Gateway load Balancing, Negative Percent Algorithms.

1 INTRODUCTION

Recently, Wireless Mesh Networks [1, 2] (WMN) has received attention as a new generation of wireless networks for providing better access service. Figure 1 illustrates a Wireless Mesh Network including some Mesh Gateway (MG), Mesh Router (MR), and Station (STA) [3] nodes.

Fig. 1. A Wireless Mesh Network
of the network cause congestion, packet loss, and queue overflow at that part [5]. It is possible to balance network traffic, and also increase network throughput and performance by combining different measures.

2 RELATED WORKS

A lot of works have been done about load balancing in wireless networks. The proposed methods are usually different from each other in terms of gateway discovery method (reactive [10]/proactive [11]) and various measures. In Ref [4], a method has been proposed in which in case of excessive packet drops (more than a threshold) a gateway detects the active source and asks the source to send its traffic to other gateway. In Ref [5], another method has been proposed that uses a combination of wired and wireless networks to select the appropriate gateway. MR nodes select the desired gateway based on the orders received from the central dispatcher on the wired side of the network. ETX [9] is the measure to select the route in central dispatcher. In Ref [6], a method called Back Bone Selection (BGS) is proposed that is a combination of Link Interface Factor (LIF), Expected Link Quality (ELQ) [8], and hop count measures. Although the combination of measures leads to better load balancing, it increases routing overhead. In Ref [10], a distributed method is used and the measure for this algorithm is a combination of bandwidth utilization and hop count. In TBMGA protocol (the base of our proposed method), each gateway broadcasts a Root Announcement (RANN) message to the network in constant intervals. This message is received by all mesh nodes. Each node selects the best message (among all of received RANN messages), and sends a RREP message towards the gateway. Therefore, each mesh node builds its own tree route to the gateway and sends its packets through the route.

Each mesh node compares RANN messages based on the ETX measure. The mesh node checks the ETX measure of the RANN message. If the measure is better than previous measures then it will be stored by the node. Otherwise, the node discards the packet. In the end, each mesh node has selected its best gateway which is a new RANN with the best measure. Anyway, to ensure a bidirectional connection, the gateway has to also be aware of new routes to reach mesh nodes. To do so, each mesh node - after selecting its main gateway (based on the ETX measure) - sends a RREP message to the gateway through the new route.

Once the gateway receives the message, it updates its routing table and detects the nodes in its neighborhood.

But, after a while it is possible that some queues of a gateway become congested (compared to other gateways). In that case, the congestion must be distributed among other gateways. To have a balanced load among the control nodes, the average of gateways’ queue congestion must not exceed a threshold. In the case of exceeding the threshold, the gateway must find one of the connected mesh nodes with the highest value and send a CHANGE-PKT message to it and ask the node to find another gateway (if that is possible) for transmitting its packets. Once the mesh node receives the CHANGE-PKT message, it sends a GW-REQ to other gateways. Those gateways that receive the message and can accept a new MN responds to it with a GW-REP message. Then the MN will be an associate of the first respondent gateway. Once a GW-REP is arrived at a MN, the node sends all of its extra traffic towards its new gateway, but if the traffic is not accepted by any gateway, the MN node continues sending its packets towards its last gateway. The first drawback of this method is that when a gateway sends a CHANGE-PKT to a node and changes the route of that node, it cannot anticipate the status of queue congestion at other gateways compared to its own queue. Thus, each node has to send some extra messages to other gateways. It is possible that the queues of other gateways are much busier than the default gateway, and hence, the packets are sent towards the last gateway which will possibly be rejected by that gateway and causes excessive rush. The second drawback is if a CHANGE-PKT causes a node to change its route, leaf nodes of that node also change their route. In other words, a CHANGE-PKT can change multiple nodes at the same time, and other gateways reject the MN. Thus, finding the best node for change and separating from the gateway has to be in such a way that does not cause a problem for other gateways. Changing the gateways might forms a ping-pong state. It means other gateways suffer from congestion and send CHANGE-PKT. If this situation continues the network becomes unstable. Sending CHANGE-PKT has to be done periodically not simultaneously, because if the congestion at all gateways exceeds the threshold at the same time, the network will be flooded by CHANGE-PKT messages, and hence, all mesh nodes have to send their packets towards their former gateway which is an unstable situation.
3 THE PROPOSED METHOD

In the proposed method, each mesh node which is recognized as a gateway starts the route discovery periodically. Each gateway broadcasts a Hello-Packet message to the network. Each node that receives the message sends a RREP message to the gateway. This allows the gateway and the node to have a bidirectional connection without any need to an already established route (proactive). Once a mesh node receives a Hello-Packet message, it creates its route towards the gateway or updates the existent entry (only in the case that sequence number of the Hello-Packet is greater than current route, or sequence numbers are the same, but the node has a better measure than the current route). Since the route measure between the source and the destination is cumulative calculated, each intermediate mesh node has to update the route discovery measure field in the Hello-Packet. Once a mesh node receives Hello-Packet messages, it sends the RREP message to the gateway.

3.1 The Proposed Algorithm

Assume that to provide an appropriate capacity in the network a new gateway node is added to the WMN. At first, the new gateway broadcasts a Hello-Packet message and announces its presence. Then, the gateway constantly broadcasts the message in specific intervals. This leads to a proactive gateway discovery at other mesh nodes. Once a mesh node receives the message, it adds a new row to its gateway table and records its specifications. Then, mesh nodes select the best message based on predetermined measures. In traditional methods that are inspired by ad hoc networks, route-relevant measures such as hop count are usually used which is not a suitable measure to select the best gateway. In the proposed method, a combination of route-relevant measures and gateway-relevant measures are used to select the gateway. The mesh router selects its default gateway based on route and gateway traffic information and updates its access route to the Internet. Negative-percent time and negative-percent hop are used as measures for route information. Moreover, negative-percent load (number of available links at the gateway, i.e., number of MR nodes that the gateway is currently serving) is used as the gateway-relevant measure. Once a Hello-Packet is arrived at a MR node, the node updates its list of gateways and selects the gateway with the lowest measure value.

3.2 Introducing Gateway Selection Measures

**Load measure:** Each gateway advertises its current load (number of mesh nodes that are currently served by the gateway) to other nodes. A mesh node calculates a negative percentage for the gateway based on the load, and selects the lowest percentage because the gateway with the lowest negative percentage can receive more traffic. Relation 3-1 shows the load measure.

$$negativeloadmeasure = \frac{\text{load of gateway}}{\text{number of gateways}}$$

$$1 < i < n, \quad n = \text{number of gateways}$$

$$Lw_{im} = \frac{l_i}{l_1 + l_2 + \ldots + l_n} \quad (3 - 1)$$

**Hop measure:** The percentage of load is not merely enough to select the best gateway, because if the distance from a mesh node to the N gateway is one hop, and to the M gateway is multiple hops, we have to also consider this distance (hop count) to select the gateway. So we calculate hop counts from current mesh node to the gateway. A negative percentage of hop counts is derived which is used as the second measure. Relation 3-2 shows the hop measure.

$$hopmeasure = \frac{\text{hopcount of gateway}}{\text{number of gateways}}$$

$$1 < i < n, \quad n = \text{number of gateways}$$

$$hw_{im} = \frac{h_i}{h_1 + h_2 + \ldots + h_n} \quad (3 - 2)$$

**Time measure:** Each data packet that is sent by the gateway spends some time to arrive at a mesh node. So if the distance between two gateways is equal then we should also enter the time into our calculations. By calculating a negative percentage of these times, we assign a third measure (time) to each gateway. Relation 3-4 shows the negative percent time measure.

$$timemeasure = \frac{\text{time to gateway}}{\text{measured time}}$$

$$1 < i < n, \quad n = \text{number of gateways}$$

$$tw_{im} = \frac{t_i}{t_1 + t_2 + \ldots + t_n} \quad (3 - 4)$$

---

1 Load Weight of I gateway for M node
2 Hop Weight of I gateway for M node
3 Time Weight of I gateway for M node
Finally, we have the new measure LBMGW by combining all previous measures which is shown in the following relation. MR nodes select the gateway having the lowest LBMGW.

\[ \text{tw}_m = \frac{t_i}{t_1 + t_2 + \cdots + t_n} \quad (3-3) \]

(3-4) – the new measure for the proposed protocol

\[ W_{\text{LBGMGW1}_m} = tW_{mi} + hW_{mi} + IW_{mi} \]

(3-5) – calculating the best gateway without considering the weight

\[ GW_M = \min\{W_{\text{LBGMGW1}_M}, W_{\text{LBGMGW2}_M}, \ldots, W_{\text{LBGMGWn}_M}\} \]

With a low congestion, load has a more effective impact because duration of send and receive is varying. At high congestion that more time is needed, we represent the load with a coefficient to keep the effect of the load. This helps a mesh node to stay away from a gateway with more congestion. The more the number of users (links) is, the more effective this staying away will be. Since the Internet users use gateways, we are trying to direct the new user towards the gateway with less number of users by increasing number of users to help the network not lose its efficiency at high congestion.

(3-6) – calculating the new measure for the proposed protocol considering time weight

\[ W_{\text{LBGMGW2}_m} = tw_{mi} + hW_{mi} + IW_{mi} \times tw_{mi} \]

(3-7) – calculating the best gateway without considering time weight

\[ GW_M = \min\{W_{\text{LBGMGW2}_M}, W_{\text{LBGMGW3}_M}, \ldots, W_{\text{LBGMGWn}_M}\} \]

Each Hello-Packet is broadcasted at a specific time that causes an extra overhead on all network routes. The following method is used to decrease the overhead of sending Hello-Packet messages. Upon receiving a Hello-Packet, each router first compares the packet with the last table. If new packet information is better than the old one, the entry is updated and the router broadcasts the packet. Other intermediate nodes do the same. On the other hand, if the current Hello-Packet is not better than the last one, the router updates the entry but does not broadcast the packet. This causes some mesh nodes do not broadcast Hello-Packets. So intermediate nodes do not receive these packets, and hence, the overhead is decreased.

### 4 SIMULATION AND RESULTS

In this section, performance of the proposed LBMGW method is evaluated against the TBMGA method in terms of average data transferred, packet delivery ratio, control overhead, and average power consumption. Simulations are carried out by GloMoSim network simulator which provides a suitable environment for simulating wireless communications.

#### 4.1 Simulation Parameters

We assume that all links have the same capacity. Transmission range for all nodes is 250 meters. Each simulation lasts 300 seconds. Number of 20-60 mesh nodes and 2-8 gateways are used. Simulation area is 1000 x 1000 meters and IEEE 802.11 is used as the MAC protocol. FTP and TELNET are chosen as traffic patterns. IP is the network protocol.

#### 4.2 Quality of Service Comparison (LBMGW VS. TBMGA)

Figure 2 shows the average delay for both protocols. With a fixed and variable number of nodes the proposed method has the lowest delay in sending data packets.
In the proposed method, each intermediate node imposes a short delay on sending packets. It is probable that some packets will not be delivered since, possibly, the link between one of the intermediate nodes and other nodes is broken for some reasons. Figure 3 verifies this possibility.

In Figure 4, packet delivery ratios of both protocols with a variable number of gateways are compared. As it shows, packet delivery ratio rises while number of gateways increase from 2 to 3.

The effects of network growth and traffic flows on average power consumption in the case of stationary nodes are shown in Figure 6. By using the proposed method, number of internetwork transmissions is decreased (compared to other methods). Thus, wireless nodes keep their power longer, and hence, their energy is saved.
5 CONCLUSION

In this paper we have proposed an algorithm for load balancing. The proposed method is seeking for a load-balancing-based solution to decrease overhead and delay, and increase network throughput. Simulation results show that with the LBMGW proposed method, if we take both route and gateway status into account, we can present a measure to perform load balancing well by uniformly directing traffic flows through the network and available gateways. It improves network capacity and decreases end-to-end delay compared to other methods.

6 REFERENCES

[5] LIN, P., YEUNG, K. H., WONG, K., Y., Multiple Path Routing using Tree-Based Multiple Portal Association for Wireless Mesh Networks