pISSN 1976-913X
eISSN 2092-805X

J Inf Process Syst, Vol.9, No.3, September 2013 http://dx.doi.org/10.3745/JIPS.2013.9.3.365

An Efficient Load Balancing Scheme for Multi-Gateways in Wireless Mesh Networks

Junping Liu* and Sang-Hwa Chung**

Abstract—In Wireless Mesh Networks (WMNs), we usually deploy multiple Internet Gateways (IGWs) to improve the capacity of WMNs. As most of the traffic is oriented towards the Internet and may not be distributed evenly among different IGWs, some IGWs may suffer from bottleneck problem. To solve the IGW bottleneck problem, we propose an efficient scheme to balance the load among different IGWs within a WMN. Our proposed load-balancing scheme consists of two parts: a traffic load calculation module and a traffic load migration algorithm. The IGW can judge whether the congestion has occurred or will occur by using a linear smoothing forecasting method. When the IGW detects that the congestion has occurred or will occur, it will firstly select another available IGW that has the lightest traffic load as the secondary IGW and then inform some mesh routers (MPs) which have been selected by using the Knapsack Algorithm to change to the secondary IGW. The MPs can return to their primary IGW by using a regression algorithm. Our Qualnet 5.0 experiment results show that our proposed scheme gives up to 18% end-to-end delay improvement compared with the existing schemes.

Keywords—Wireless Mesh Network, Internet Gateway, Multi-Gateway Multi-Root, Load Balancing

1. INTRODUCTION

Nowadays, Wireless Mesh Networks (WMNs) are attracting more and more attention from both researchers and enterprises. Due to its high bandwidth, large coverage range, self-healing, auto-configured, and extensible features, it can provide wireless service for a large variety of applications in personal, local, and metropolitan areas. By configuring the IGW, WMNs can be integrated with the Ethernet, sensor network, wireless-fidelity (Wi-Fi), worldwide interoperability for microwave access (WiMAX), and WiMedia networks conveniently [1].

The IGW has a fixed bandwidth shared by all of its clients so that the performance of per client reduces as the number of client increases. It is also easier for the IGW to suffer bottleneck issues. To address these problems, the WMN is usually configured with multiple IGWs. Even though the usage of multiple IGWs can provide significant capacity benefits, the introduction of multiple IGWs does not enable a linear increase in capacity [2]. Only with a proper load-balancing scheme, can the capacity increase linearly. Consequently, preventing the IGW from

* This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(2012R1A1A2043531)
Manuscript received July 30, 2012; first revision October 25, 2012; accepted January 1, 2013.

Corresponding Author: Sang-Hwa Chung
* Dept. of Logistic Information Technology, Pusan National University, Korea (iujunping8752@gmail.com)
** Dept. of Computer Engineering, Pusan National University, Korea (shchung@pusan.ac.kr)

suffering from the bottleneck problem and distributing the network traffic load evenly among IGWs should be realized as the key issues.

There are two main categories of reasons for the uneven load distribution across WMNs. The first reason is the uneven client distribution. Generally, users are unevenly distributed throughout the WMN, and furthermore, the distribution also changes with time. The second reason is the uneven user demands. The user demands may vary widely within a given network domain. The traffic consists of several short flows and long flows. Thus, the demands of users greatly vary as well. These two factors cause a strong difference in the aggregated load imposed on each IGW and thus some IGWs may suffer congestion problems, whereas some IGWs may be severely underutilized. The congestion may have an effect on the network performance, such as aggregated network throughput, packet delivery ratio, and end-to-end delay. In order to improve the network performance, we propose an efficient method to balance the traffic load among different IGWs.

The remainder of this paper is organized as follows: in Section 2, we give a brief introduction to the previous related work. Then, in Section 3, we describe the Multi-Gateway Multi-Root Hybrid Wireless Mesh Protocol (MGMR-HWMP). Based on the MGMR-HWMP, we deploy a load-balancing scheme at each IGW to improve the performance of the WMN and the details of this load-balancing scheme will be described in Section 4. The simulation experiments results and some analyses will be shown in Section 5. Finally, in Section 6, some conclusions about our research will be given.

2. Related Work

Researchers have done a huge amount of research in designing multiple gateway routing protocols for wireless networks. Wakikawa in [3] mainly describes the proactive and reactive gateway discovery mechanisms. In the reactive approach, when a router needs a gateway, it broadcasts a solicitation to the network to find the gateway. In the proactive approach, the gateway broadcasts a periodic message to the network to inform other nodes of its existence. As there are two methods for gateway discovery, some researchers have discussed which one has less overhead. Some researchers consider that the reactive approach have less overhead compared with the proactive approach, but [4] shows that the reactive approach could be as costly as the proactive approach. Authors in [5] proposed a hybrid approach, which uses a combination of both reactive and proactive approaches. In [6], Hamidian provides an implementation of all the three gateway discovery approaches based on the AODV protocol [7]. However, it uses hops as the only metric to select a gateway, which may introduce the issue of bottleneck problem.

Once routers have received the gateway information, gateway selection should be processed. Generally, gateway and route selection can be coupled together in that the network tries to find the best route to any of the available gateways by evaluating the route metrics on the available path. The traditional gateway selection is based on the metric of minimum hop count. However, hop count is a poor choice as a routing metric in multi-hop wireless networks [8]. In [9], the contention level metric was used instead of the minimum hop metric for gateway selection. The authors in [10] use the Expected Transmission Time (ETT) metric [11] for backhaul WMNs. In [12], the authors take into account the traffic load along a path in addition to using a minimum hop count to select an efficient gateway selection. [13-16] propose to use the traffic load information of the gateway as the metric for gateway selection. In [17], the authors propose a gateway and route selection scheme for backbone WMNs that takes the gateway load, route interference, and path quality metric into account.

Recently, most of the schemes focus on load balancing over the links or MPs connecting to the IGW and do not focus on alleviating the congestion at the IGW. In [18], the authors devise a scheme for load balancing over the links that uses path capacity and gateway link capacity as the cost metric. In [16], each IGW maintains the intersection area of the other IGWs. MPs within this intersection area are considered as the candidates for migrating the traffic load as they have similar distance to each available IGW. When the MP receives multiple advertisement messages from IGWs, it compares the traffic status of the IGWs. If the MP recognizes that the traffic gap among all of the available IGWs is bigger than the balance threshold value, it will change its default IGW to the candidate IGW with the lightest traffic load. The traffic load gap is defined as the ratio of the maximum traffic load dividing the minimum traffic load of the available IGWs. However, it is possible that both the maximum and minimum traffic load of the IGW is below the threshold, and the traffic load gap will still exist. In this situation, it is unnecessary to change the primary IGW to the candidate IGW. [16] does not consider the set of MPs which need to change the service primary IGWs to the candidate IGWs. If many MPs decide to change the primary IGWs to the candidate IGWs, it may potentially induce a congestion problem on the candidate IGWs.

3. THE MULTI-GATEWAY MULTI-ROOT HYBRID WIRELESS MESH PROTOCOL

The original HWMP is configured with only a single IGW and this IGW is equipped as the root. All of the MPs in WMNs are connected with the only IGW. This may induce the bottleneck problem in WMNs. For this reason, we propose the Multi-Gateway Multi-Root Hybrid Wireless Mesh Protocol (MGMR-HWMP), which is configured with multiple IGWs and each IGW is set as the root.

In MGMR-HWMP, each IGW builds and maintains bidirectional routes towards each MP during fixed interval and deals with the following two types of route management tables: a Route Table (RT) and an Internet Gateway Information Table (IGIT). For MPs, they also maintain a Route Table (RT) and in addition they keep an Available Internet Gateway Table (AIGT). The Route Table (RT) records the route information, such as the destination address, sequence number (SN), hop count, and routing metric. The Internet Gateway Information Table (IGIT) caches the information of the available IGWs, such as the IGW address, traffic load, IGW state, and the information of the available MPs (AvailableMPInfo). The Available Internet Gateway Table (AIGT) keeps the information of the available IGWs (e.g., the IGW address and the related management message data).

Since most of the traffic in WMNs is oriented towards the Internet, we concentrated our attention on the proactive routing mode in MGMR-HWMP. There are three categories of messages in MGMR-HWMP: Root Announcement (RANN), Route Request (RREQ), and Route Reply (RREP). In the RANN message, it records the sequence number (SN), IGW address, hop count, metric, lifetime, traffic load, and IGW state information.

Each IGW will build its tree-based topology by broadcasting the RANN message. When the MPs receive the RANN message, they should select one IGW as the primary gateway and keep the other IGWs as the secondary gateway. When mobile nodes connect with a mesh access point (MAP) it should firstly send its packet to the MAP and then the MAP will rewrite its primary gateway information to its own primary gateway. MPs also maintain the tree-based topology and remain aware of the distribution of the traffic load of all the available IGWs to achieve better performance. After receiving the RANN message, each MP should use RREQ and RREP mechanisms to guarantee the bidirectional route between the MP and the primary IGW. The MP sends a unicast RREQ message to its selected parent MP. When the parent MP receives the unicast RREQ, it forwards this RREQ to its selected parent and updates the corresponding entry of the RT with the address of the next hop toward the MP, which originates the RREQ. When the RREQ arrives at the IGW, the IGW updates its entry of the RT and IGIT and replies with a unicast RREP to the MP that originated the unicast RREQ. After the MP receives the RREP message, the bidirectional communication between the MP and the IGW is established. If the MP acts as the MAP connecting with some clients, it should it should send a gratuitous RREP to the primary gateway to register the information of the clients. We can obtain the multi-gateway multi-root topology after using all of the procedures we described above.

Once establishing the tree-based proactive topology, when MPs receive a RANN message, they should maintain the tree-based topology as shown in Fig. 1. The procedure in Fig. 1 causes an MP to obtain the best route towards an IGW.

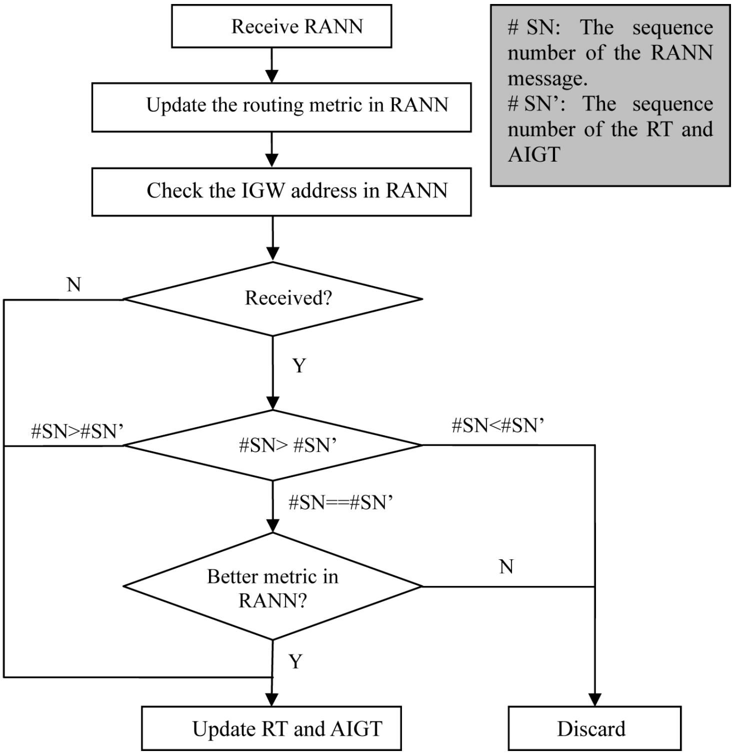
Receive RANN
Update the routing metric in RANN
Check the IGW address in RANN
# SN: The sequence number of the RANN message.
# SN': The sequence number of the RT and AIGT
N
Received?
Y
#SN>#SN'
#SN> #SN'
#SN<#SN'
#SN==#SN'
Better metric in RANN?
N
Y
Update RT and AIGT
Discard

Fig.1. The procedure for maintaining the tree topology

4. THE GATEWAY LOAD BALANCING SCHEME

Even though the MGMR-HWMP described in Section 3 is configured with multiple IGWs, it does not enable a directly and linear increase of the network performance [2]. To achieve better WMN performance, we propose an efficient gateway load-balancing scheme based on the MGMR-HWMP, which is called Multi-Gateway Multi-Root Hybrid Wireless Mesh Protocol with Load Balancing (MGMR-LB). The proposed gateway load-balancing scheme consists of two parts—a traffic load calculation mode and a traffic load migration algorithm. The basic procedures are shown as follows:

1. When the RANN interval is coming, each gateway should calculate the traffic load of the current RANN interval and predict the traffic load of the next interval by using Brown's Linear Smoothing method.
   (1) The gateway should judge whether it has suffered or will suffer congestion problem.
      ① If the gateway has suffered or will suffer congestion problems, it should use the load balancing scheme.
         A. The gateway should calculate the traffic load it intends to transfer.
         B. After calculating the transferred traffic load, the gateway should use the Knapsack Algorithm to find the best set of MAPs (mesh access points) that need to change their primary gateway to the secondary gateway to relieve the traffic load of the primary gateway. In the Knapsack Algorithm, we consider the traffic load of the MAP, the transferred traffic load, and the number of MAPs.
         C. After getting the best MAP set, the gateway sends notification messages to those MAPs.
         D. When the MAP receives the notification message, it should change its primary gateway to the selected secondary gateway.
   (2) After using the load balancing scheme, if the congested gateway detects that it is in the stable state for several RANN intervals, it should use the mesh router regression algorithm to inform the MAP return to the primary gateway. We propose the mesh router regression algorithm, as we want to reduce the potential congestion problem of the secondary gateway.

We will introduce the details of our load-balancing scheme in the subsections below.

4.1 The Traffic Load Calculation Mode

Firstly, when the RANN period is coming, each IGW should calculate the traffic load of the current interval. As our proposed scheme is based on MGMR-HWMP, we use the RANN interval as the calculation interval to calculate the average traffic load. It can be calculated as:

$$TL_i = \frac{NumofPacketsReceived_i \times AveragePacketSize \times 8}{\mathrm{IntervalLength}} \quad (1)$$

where *NumberOfPacketReceived*$_i$ is the total number of data packets received by the IGW. *AveragePacketSize* is the average size of the received data packets. IntervalLengh is the calculation interval, which means the RANN interval.

We cannot judge the IGW state only by the traffic load information of the current interval. We also need to consider the average traffic load of the IGW in the next interval. We used the double exponential smoothing method to predict the traffic load. The reason for using this method is that the double exponential smoothing method can be applied to time series data to make forecasts. The time series data are a sequence of observations. The observed phenomenon may be an essentially random process or may be an orderly observations process. The exponential smoothing assigns exponentially decreasing weights over time.

In our case, the time series data is the average traffic load of the IGW. There are many methods for double exponential smoothing, but in our paper we use Brown's Linear Smoothing (LES) [19]. The raw traffic load data sequence is often represented by $\{TL_i\}$. We use $\{S_i\}$ to present the smoothed value for interval *i*, and $\{b_i\}$ is our best estimate of the trend at interval *i*. The output of the algorithm is now written as TL_{i+m}, which is an estimate of the value of the traffic load in the *(i+m)*th interval, where $m>0$ is based on the raw traffic load information up to interval *i*. In our case, we are just trying to predict the traffic load of next interval, which means *m* equals to 1. The common calculation is:

$$S_i^{'} = \alpha \times TL_i + (1-\alpha) \times S_{i-1}^{'} \tag{2}$$

$$S_i^{''} = \alpha \times S_i^{'} + (1-\alpha) \times S_{i-1}^{''} \tag{3}$$

$$a_i = 2 \times S_i^{'} - S_i^{''} \tag{4}$$

$$b_i = \frac{\alpha}{1-\alpha} \times (S_i^{'} - S_i^{''}) \tag{5}$$

We set the initial value of S_0' and S_0" to TL_1 which is the traffic load of the first calculation interval. Then we can predict the traffic load of the next interval as:

$$TL_{i+1} = a_i + b_i \tag{6}$$

Since we have obtained the traffic load information of both the current and next interval, we can judge the state of the IGW. There are three states of the traffic load: Low State, the Medium State, and the High State.

1. Low state: $TL_i < TL_{lower\_threshold}$
2. Medium state: $TL_{lower\_threshold} \leq TL_i < TL_{upper\_threshold}$
3. High State: $TL_i \geq TL_{upper\_threshold}$

$TL_{upper\_threshold}$ is the upper bound value and $TL_{lower\_threshold}$ is the lower bound value.

After judging the traffic load state, we should judge the state of the IGW. There are also three states for the IGW: the congestion state, the stable state, and the underutilized state, as shown in Table 1.

When the IGW is in the congestion state, the IGW has a high probability of suffering from the bottleneck problem and should use the load-balancing scheme. In addition, when the IGW is in the congestion state, it should not be selected as the secondary gateway by other congested

Table 1. Internet Gateway state judgment

Predicted Current	Low State	Medium State	High State
Low State	Underutilized	Stable	Stable
Medium State	Underutilized	Stable	Congestion
High State	Stable	Stable	Congestion

IGWs. For IGWs in underutilized state, they can work in a stable state and can be considered as the best choice to be the secondary IGW for the congested IGW. For IGWs in the stable state, they can work in a stable state and can be considered as the second choice to be the secondary IGW for the congested IGW. The traffic load information and Internet gateway state information will be packaged in the RANN message.

4.2 The Mesh Router Migration Algorithm

When the IGW is in the congestion state, it should consider whether it needs to use the load-balancing scheme. To achieve this target, the congested IGW should calculate the average traffic load of all the available IGWs recorded in its Internet Gateway Information Table (IGIT) and itself. It can be calculated as:

$$TL_{IGW} = \frac{1}{NumOfIGW} \times \sum_{i=1}^{NumOfIGW} TL_{IGW\_i} \tag{7}$$

where *NumOfIGW* is the total number of IGWs including itself and TL_{IGW_i} is the traffic load of IGW i. In addition, we presentthe traffic load of the IGW itself as: TL_{IGW_self}.

Now we can judge whether this congested IGW needs to use the load-balancing scheme. If $TL_{IGW\_self} \geq \beta \times TL_{IGW}$ (β is the experienced parameter), this congested IGW should use the load-balancing scheme. Fig. 2 describes the load migration algorithm. If not, there is no need to use it. When the congested IGW decided to use the load-balancing scheme, it firstly selects an IGW with the lightest traffic load among all the available none congested IGWs by using the SearchAvailableIGWwithLightestTL function. If there is an available IGW, the IGW should calculate the traffic load it intends to transfer $TL_{transfer}$. The congested IGW should also get the traffic load information TL_{IGW_Light} of the IGW_{Light} by using the GetTrafficLoadInfo function. If TL_{IGW_Light} is smaller than TL_{IGW}, the congested IGW continues to calculate the traffic load the IGW_{Light} can provide $TL_{Provide}$. If $TL_{Provide}$ is larger than $TL_{transfer}$, then we can use the Knapsack Algorithm to find the best MAP set to change the primary IGW to the IGW_{Light}.

The exact Kanpsack Algorithm in the mesh router regression algorithm is the 0-1 knapsack algorithm. In the 0-1 knapsack algorithm, there are n items (xn). Each item has value (vn) and weight (wn). The problem is that if the weight (W) of packets is limited, then we have to figure out the best set of these items, as shown in Equation (8).

$$\max \sum_{i=1}^{n} v_i x_i \text{ subject to } \sum_{i=1}^{n} w_i x_i \leq W \text{, } x_i \in \{0,1\}\text{;} \tag{8}$$

In our 0-1 knapsack algorithm, the item is the MAP. The value and the weight of the item is the metric of the MAP (MP_metric) and the transferred traffic load of MAP (TL_{MPi}). The total value W is $TL_{transfer}$. We can show the algorithm as shown in Equation (9):

$$\min \sum_{i=1}^{n} MP\_metric_i x_i \text{ subject to } \sum_{i=1}^{n} TL_{MP_i} x_i \le TL_{transfer}, x_i \in \{0,1\}; \tag{9}$$

As we know the decision of the 0-1 Knapsack Algorithm is a NP-complete, the complexity of running a traditional Knapsack Algorithm is O(nW). The complexity of our Knapsack Algorithm is O($Set_{transfer}$(MAP)* $TL_{transfer}$). The complexity is based on the number of the MAPs set and the total traffic load that we want to transfer.

We want to select the MAPs that maintain a better routing metric to the selected secondary IGW. In Equation (9), we want to get the minimum value of the total routing metric of the selected MAP set .We also constrain the traffic load of the MAPs as we want to get the best set of MAPs to change the gateway, and then the congested gateway can be relieved better. After selecting the MAPs, the congested IGWshould send notifications to the selected MAP set to inform them to change from the primary gateway to the secondary gateway by using the SendNotificationMsg function.

Even though some MAPs have changed the IGW from the primary to the secondary IGW, it does not mean that these MPs will never return to their primary IGW. If these MAPs keep communicating with the secondary IGW, it may potentially induce a congestion problem in the secondary IGW. To avoid this ping-pong problem, MAPs cannot return to their primary IGW as soon as their primary IGW is in non-congestionstate. MAPs can return to their primary gateway only when their primary gateway is in non-congestionstate for several intervals and then they can return one by one. To avoid this ping-pong problem, a MAP with a light traffic load has a

Load Migration Algorithm

```
$IGW_{Light}$=SearchAvailableIGWwithLightestTL(IGIT);
If(exist $IGW_{Light}$){
    $TL_{transfer}$= $TL_{IGW\ self}$- $TL_{IGW}$;
    $TL_{IGW\ Light}$=GetTrafficLoadInfo($IGW_{Light}$);
    If($TL_{IGW\ Light}$< $TL_{IGW}$){
        $TL_{Provide}$= $TL_{IGW}$- $TL_{IGW\ Light}$;
        If($TL_{Provide}$ > $TL_{transfer}$){
            $Set_{transfer}$(MAP)=
  KnapsackAlgorithm(IGIT(AvailableMPInfo));
                If($Set_{transfer}$(MAP)= = NULL){
$Set_{transfer}$(MAP)= SearchAvailableMPwithHighestTL
(IGIT(AvailableMPInfo));}
SendNotificationMsg($Set_{transfer}$(MAP));
}
    }
}
```

Fig. 2. Procedure for the load migration algorithm

Mesh Router Regression Algorithm

```
If (IGW is under none congestion state)
    ++IGW_NonecongestionStateCounter;
If(IGW_NonecongestionStateCounter = = K)
      If($Set_{transfer}$(MAP)!=NULL){
$MAP_{Light}$=SelectMPwithLightestTrafficLoad($Set_{transfer}$(MAP));
         SendRegressionNotifyMsg($MAP_{Light}$);
         Reset IGW_NonecongestionStateCounter=0;}
       Else
          {End;}
Else { Reset IGW_NonecongestionStateCounter=0;}
```

Fig. 3. Mesh router regression algorithm

high priority to be selected to return to its primary IGW. This MAP regression algorithm will be processed until all the transferred MAPs return to the primary IGW. The IGW that uses the load-balancing scheme cannot receive any traffic load from other IGWs as its state is still unstable and it should continue to use the regression algorithm. Fig. 3 shows the details of the MAP regression algorithm.

5. Simulation Performance Evaluation

5.1 Simulation Environment

The efficiency of our proposed load-balancing scheme was evaluated using Qualnet 5.0. A wireless mesh network with an area of 1500×1500 m^2 was established. There are 34 static MPs and 3 mesh portals distributed on the canvas, as shown in Fig. 4. In Fig. 4, MPs are marked from MP5 to MP38, and MP2, MP3, and MP4 are configured as the IGWs that connect with the Ethernet network node1.

The parameters for all the simulations are listed in Table 2. We obtained the value of the IGW

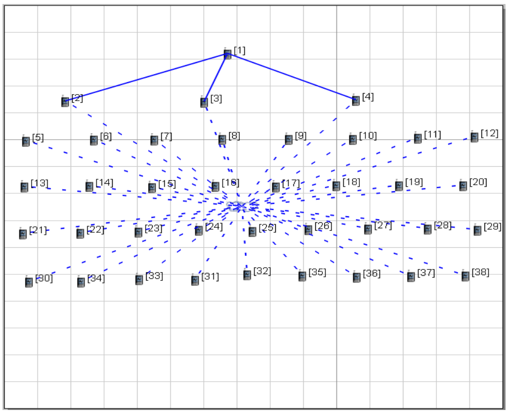

Fig. 4. Simulation scenario in Qualnet simulator

Table 2. Simulation Parameters

Simulation Time	300s
Simulation Area	1500×1500
PHY Layer	802.11a
Transmission Range	250m
Transmission Data Rate	9Mbps
Traffic Type	CBR(UDP)
Packet Size	2048bytes
Number of mesh routers	37
RANN interval	4s
$\text{Threshold}_{\text{upper}}$, $\text{Threshold}_{\text{lower}}$	2.5Mbps, 1.0Mbps
α, β	0.7, 1.3

$\text{threshold}_{\text{upper}}$, $\text{threshold}_{\text{lower}}$, α and β from numerous experiments. The parameters for all the simulations are listed in Table 2. In this paper, the α, β is experienced values. Firstly, when we run the simulation, we chose the value of α, β according to some related papers. But we did not just choose the value of α, β only according to their methods. We checked the other value of α, β by doing experiments. We also checked the value of α from 0.5 to 0.9 with 0.1 increment and β from 1.1 to 1.5 with 0.1 increment. Among all of the tested values, we found 0.7 and 1.3 are the best values for α and β. For the number of IGW traffic load calculation intervals, we used the RANN interval as the calculation interval in order to calculate the traffic load of the IGW more easily. As the IGW intent to work smoothly, it should obtain the traffic load of the current and future interval. For this reason, we selected one RANN interval as the traffic calculation interval.

5.2 Simulation Results and Analysis

We compared the performance of our proposed scheme MGMR-BL with the original single gateway HWMP and the load balancing approach LB-MPS [16]. We deployed 12 data flows generated by the clients, under MP15, MP17, MP20, MP21, MP25, MP29, MP31, MP33, MP34, MP35, MP 37, and MP38. In order to better observe the difference ofthe performances, we conducted experiments with the increasing data flow rate from 80 Kbps to 640 Kbps with the incremental of 80 Kbps. In Fig. 5, we can observe that when using a single gateway, the average packet delivery ratio decreases rapidly as the flow rate increases. This is due to all 12 flows being connected to the same IGW MP3, which leads to heavily congested paths that are in proximity of the IGW. With the introduction of more IGWs, the packet delivery ratio improves at every rate. The packet delivery ratio is an average of 36.8% better than HWMP and an average of 6.6% better than LB-MPS. We also observe the same trend for the end-to-end delay in Fig. 6. The introduction of more IGWs makes the traffic to be split along different paths, so that we can get a better spatial usage of wireless bandwidth. The end-to-end is a 41.1% improvement compared with the HWMP and a 15.3% improvement as compared with LB-MPS. Fig. 7 shows that when the flow rate is a small value, even though we increased the per flow rates, the total throughput increase gap is not so big. But as the flow rate increases, the total throughput of the single IGW situation will remain stable. However, for our proposed scheme, the aggregated network throughput rapidly increased. The results show that the aggregated network throughput is 37.3% and 6.3% better compared with HWMP and LB-MPS, respectively.

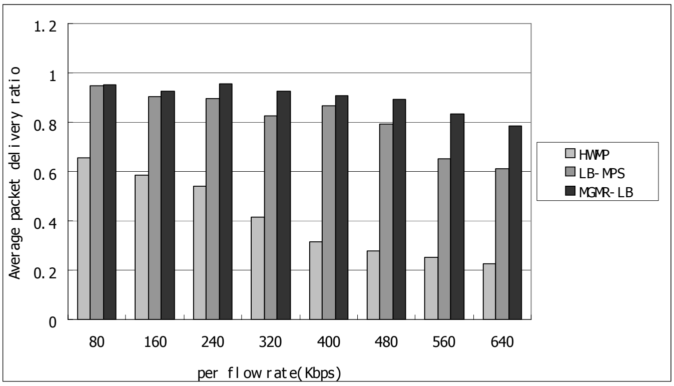

Fig. 5. Average packet delivery ratio

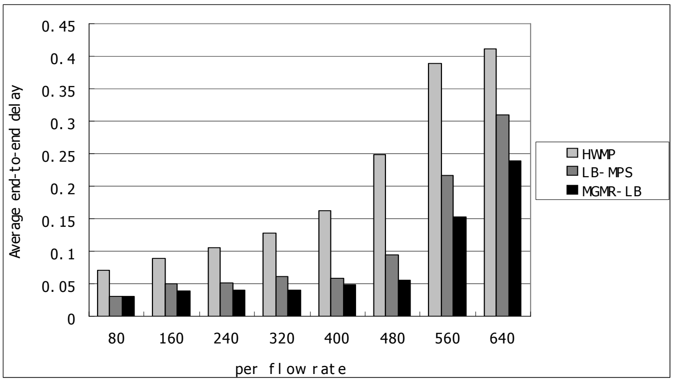

Fig. 6. Average end-to-end delay

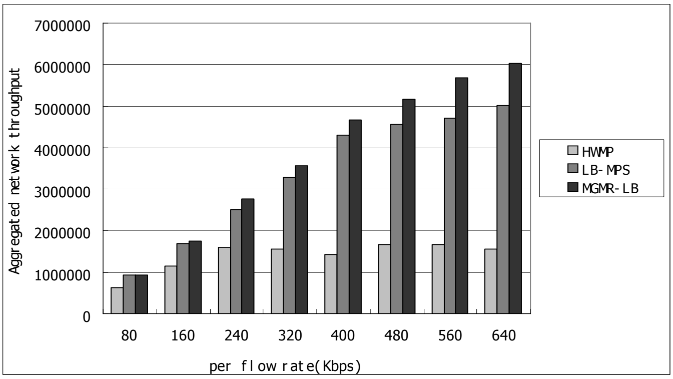

Fig. 7. Aggregated network throughput

We then compared the performance of our proposed load-balancing scheme with LB-MPS and MGMR-HWMP. In this experiment, we again deployed 12 data flows, which were generated by the clients, under MP15, MP23, MP25, MP26, MP29, MP30, MP31, MP32, MP34, MP35, MP37, and MP38. We also conducted experiments with the increasing data flow rate from 160

Kbps to 1040 Kbps with the incremental of 80 Kbps. We extended the tested flow rate range, as we wanted to better observe the difference of the performance when a certain IGW is under a high traffic load.

As we can see from Fig. 8 to Fig. 10, when the flow rate is low, the average packet delivery ratio, end-to-end delay, and aggregated network throughput of the data flow is almost the same for all three schemes. However, when the flow rate increases, we can see that sometimes the performance of the LB-MPS is worse than the MGMR-HWMP and our proposed load balancing scheme. The reason is that for LB-MPS, MP decides to change the primary gateway according to the traffic load gap between different gateways. When the mesh router detects that the gap is bigger than the threshold for several intervals, it will change its primary gateway to another according to probability P, which is calculated based on the gateway information. As the MPs share the same gateway information, the MPs can simultaneously change their default gateways. Another reason is that, if several MPs change to the same gateway, it may induce congestion at that particular gateway. Consequently, the performance of LB-MPS is not stable or sometimes even worse than our proposed MGMR-HWMP. For the MGMR-HWMP without a load balancing scheme, the average packet delivery ratio will decrease linearly, as the flow rate increases. For our proposed load balancing scheme, even though the flow rate increases, the average end-to-end delay does not increase rapidly, and the aggregated network throughput keeps increase.

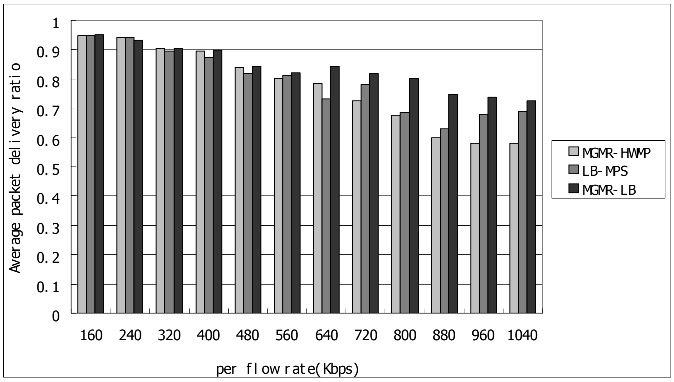

Fig. 8. Comparison of packet delivery ratio

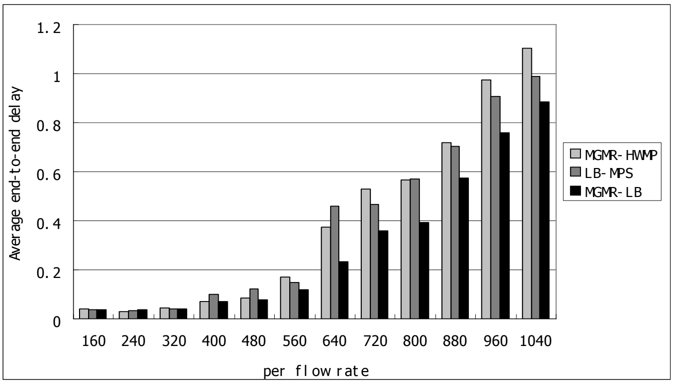

Fig. 9. Comparison of end-to-end delay

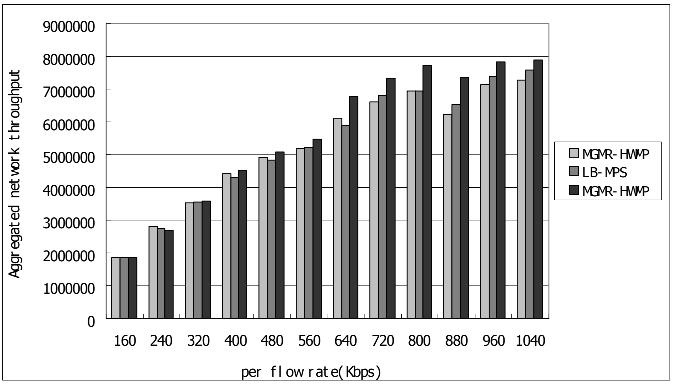

Fig. 10. Comparison of aggregated network throughput

For the packet delivery ratio, it does not decrease such rapidly. We can observe that the packet delivery ratio is improved by an average of 8.2% and 5.8% compared with MGMR-HWMP and LB-MPS, respectively. The average end-to-end delay is improved by an average of 15.6% and 18.1% compared with MGMR-HWMP and LB-MPS, respectively. For the aggregated network throughput, it can be improved by an average of 5.8% and 5.4%, respectively.

6. Conclusions

Our research addresses the IGW bottleneck problem that is caused by the high traffic load distributed in WMNs by deploying multiple IGWs and configuring an efficient load balancing scheme at the IGWs to avoid an IGW congestion problem.

The simulation results show that our proposed load balancing scheme gives a 36.8% improvement on packet delivery ratio and a 41.4% and 37.3% improvement on end-to-end delay and aggregated network throughput, respectively, compared with a single IGW HWMP. Compared with MGMR-HWMP and LB-MPS, the packet delivery ratio is 8.2% and 5.8% on average improved compared with MGMR-HWMP and LB-MPS, and the average end-to-end delay is 15.6% and 18.1% on average improved. For the aggregated network throughput, we can get an improvement by 5.8% and 5.4% on average.

References

[1] IF. Akyildiz, X. Wang and W. Wang, “Wireless mesh networks: a survey,” Computer Networks, vol. 47, no. 4, 2005, pp.445-487.

[2] S. Lakshmanan, K. Sundaresan and R. Sivakumar, “On Multi-Gateway Association in Wireless Mesh Networks,” 2nd IEEE workshop on Wireless Mesh Networks, Reston, AV, USA , 2006, pp. 64-73,.

[3] R. Wakikawa, J.T. Malinen, C.E. Perkins, A. Nilsson and A. J. Tuominen, “Global connectivity of IPv6 Mobile Ad Hoc Networks”, draft-wakikawa-manet-globalv6-05.txt, Mar.2006.

[4] P.M. Ruiz and A.F. Gomez-Skarmeta, “Adaptive Gateway Discovery Mechanisms to Enhance Internet Connectivity for Mobile Ad Hoc Networks,” AdHoc & Sensor Wireless Networks, 2005, vol.1, pp.159-177.

[5] P. Ratanchandani and R. Kravets, “*A Hybrid Approach to Internet Connectivity for Mobile Ad hoc Networks*,” *Wireless Communications and Networking Conference*, New Orleans, USA, 2003,

pp.1552-1527.

[6] A. Hamidian, Ulf K., A. Nilsson, "Performance of Internet Access Solutions in Mobile Ad Hoc Networks," Springer's Lecture Notes in Computer Science, vol.3427, 2005, pp.189-201.

[7] C.E. Perkins and E. M. Royer, "*Ad-Hoc on-Demand Distance Vector Routing*," *Mobile Computing Systems and Applications*, New Orleans, LA, Feb.,1999, pp.90-100.

[8] C. Shin, S.H. Kim, S. An, "Stable Gateway Selection Scheme based on MANET with Internet," International Conference on Computer and Information Technology, Seoul, Korea, Dec., 2006.

[9] Y. Fu, K. W. Chan, K.S. Tan and B.S. Yeo, "*Multi-metric Gateway Discovery for MANET*," *Vehicular Technology Conference*, Melbourne, Vic., 2006, pp.777-781.

[10] L. Luo, D. Raychaudhri, H. Liu, M. Wu and D. Li, "*Improving End-to-End Performance of Wireless Mesh Networks through Smart Association*," *Wireless Communications and Networking Conference*, April, 2008.

[11] R. Draves, J. Padhye, and B. Zill, *"Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks," ACM Mobicom,* Philadelphia, PA, Sept., 2004, pp. 1084-1094.

[12] S.M. Asif Iqbal, M.H. Kabir, *"An Improved Internet Gateway Discovery and Selection Scheme in Mobile Ad Hoc Network," Electrical and Computer Engineering,* Dhaka, Dec,2010, pp.372-375.

[13] D. Nandiraju, L. Santhanam, N. Nandiraju, and D. P. Agrawal, *"Achieving Load Balancing in Wireless Mesh Networks Through Multiple Gateways," IEEE International Conference on Mobile Adhoc and Sensor System,* Vancouver, BC,Oct,2006, pp.807-812.

[14] J. Shin, H. Lee, J. Na, A. Park, and S. Kim, *"Load Balancing among Internet Gateways in Ad Hoc Networks," IEEE Vehicular Technology Conference,* Sept, 2005,pp. 1677 - 1680 .

[15] R. Kumar, M. Misra, A. K. Sarje, *"An Efficient Gateway Discovery in ad hoc Networks for Internet Connectivity," Conference on Computational Intelligence and Multimedia Applications,* Sivakasi, Tamil Nadu , Dec., 2007, pp. 275-282.

[16] K. Yonggyu, J. Yeonkwon, S. Myunghwan and M. Joongsoo, *"Load-balanced Mesh Portal Selection in Wireless Mesh Network," IEEE Military Communications Conference,* Orlando, FL, USA, Oct., 2007, pp.1-6.

[17] U. Ashraf, S. Abdellatif, and G. Juanole, *"Gateway Selection in Backbone Wireless Mesh Networks," Wireless Communications and Networking Conference (WCNC),* Budapest,May,2009, pp.2548-2553.

[18] A. Raniwala, Prof Chiueh, *"Architecture and Algorithms for an IEEE 802.11-based Multi-channel Wireless Mesh Network," in IEEE INFOCOM,* Miami, FL, USA, Aug., 2005, pp 2223 - 2234.

[19] Brown R. G. and Meyer, R. F. Meyer, "The Fundamental theorem of exponential smoothing," Operations Research, vol. 9, 1961, pp. 673-685.

Junping Liu

She received the BS in Computer Science from Northeastern Uni. from 2006 to 2010 in China and MS degrees in Logistic Information Technology Department from Pusan National Univ. from 2010 to 2012. And now she is working in China Mobile Communications Research Institute. Her research interests include Wireless Mesh Network and Cloud Computing.

Sang-Hwa Chung

He received the B.S. degree in electrical engineering from Seoul National University, Seoul, Korea, in 1985, the M.S. degree in computer engineering from Iowa State University, Ames, in 1988, and the Ph.D. degree in computer engineering from the University of Southern California, Los Angeles, in 1993.

He was an Assistant Professor with the Department of Electrical and Computer Engineering, University of Central Florida, Orlando, from 1993 to 1994. He is currently a Professor with the Computer Engineering Department, Pusan National University, Busan, Korea. His research interests are in the areas of embedded systems, RFID systems, and wireless mesh networks.