A Novel Solution for End-to-End Fairness Problem in Wireless Mesh Networks

Liang Zhang Shigang Chen Ying Jian Ming Zhang
Department of Computer & Information Science & Engineering
University of Florida, Gainesville, FL 32611, USA *

***Abstract*—A wireless mesh network (WMN) provides a flexible and low-cost solution for end users to connect to the Internet through its multi-hop infrastructure. For such a network to proliferate, a fundamental problem that must be solved is to ensure the fair allocation of network bandwidth to all participating parties. This paper proposes a cross-layer design for achieving end-to-end maxmin fairness in WMNs. At the network layer, it allocates maxmin shares of network capacity to end-to-end flows. At the MAC layer, it realizes the maxmin bandwidth allocation through a two-level packet scheduling algorithm. The proposed design is able to equalize the end-to-end bandwidth allocation to competing flows that share common bottlenecks, while fully utilizing the network capacity. Comparing with previous works, our solution has two advantages. It is based on the popular IEEE 802.11 DCF. It achieves far better fairness (or weighted fairness) among end-to-end flows.**

I. Introduction

In recent years, wireless mesh networks (WMNs) have emerged as a platform to provide a flexible and low-cost lastmile broadband Internet access service [1]. In a WMN, mesh routers not only transmit packets for mesh clients in their own regions, but also forward packets for other clients whose destinations are beyond the direct wireless transmission range.

This paper studies a fundamental problem, how to support weighted maxmin bandwidth allocation among all end-to-end flows in an IEEE 802.11 DCF based WMN. While much research concentrates on the MAC layer, the user's perception on WMNs is however determined mainly based on the networks' end-to-end effectiveness. For example, most users in a WMN compete for accessing the Internet through a few gateway routers which are directly connected to the Internet. Each user should be guaranteed to obtain a fair share of network bandwidth for accessing gateway routers or communicating with other users in the same WMN regardless of the distance between the two ends. Another example is that if a user contributes more to the network, she may demand that her traffic is given more weight than others' traffic, which is a weighted fairness problem. The maxmin fairness solution in wired networks [2] cannot be applied to WMNs because it requires each link has a fixed bandwidth capacity, which is not true in a wireless environment where link bandwidth is dynamic, depending on channel conditions and contention from nearby links.

Researchers have proposed solutions for MAC-layer fairness [3], [4]. However, for *end-to-end flows*, which are common in WMNs, those solutions ignore the relationship among the *subflows* from the same end-to-end flow [5]. If an upstream subflow is allocated more bandwidth than its downstream subflow (of the same end-to-end flow), the router in the middle will receive packets at a faster rate than it can forward. The buffer of the router may be overflowed and cause packet drops, wasting channel bandwidth that could have been used by other flows. Among all subflows of an end-to-end flow, the one with the lowest rate becomes the bottleneck. All subflows should have the same rate defined by the bottleneck.

There are other works addressing the fairness issue in wireless networks. One-hop flow maxmin is studied in [6]. End-to-end maxmin in CDMA/FDMA networks is investigated in [7]. A distributed algorithm that achieves aggregate fairness in sensor networks is proposed in [8], assuming that all flows are destined to the same base station. There are also some utility-based solutions, assuming each wireless node has a fixed bandwidth capacity [9], considering only single-hop flows [10], eliminating contention among neighboring nodes by using separate CDMA/FDMA channels for wireless links [11]. However, they do not provide a maxmin solution for end-to-end flows in an IEEE 802.11 DCF multihop network.

In this paper, we study the end-to-end fairness problem in WMNs based on IEEE 802.11 DCF. We first present a generalized maxmin model. By applying this model on a WMN, we are able to assign each end-to-end flow a maxmin fair share. A two-level packet scheduling algorithm is proposed to allocate channel bandwidth such that the maxmin fair shares of the flows can be realized. Comparing with previous works, our solution has two advantages. It is based on the popular IEEE 802.11 DCF. It achieves far better fairness (or weighted fairness) among end-to-end flows.

The rest of the paper is organized as follows. Section II describes the network model and our objective. Section III presents a generalized maxmin model, based on which we design the maxmin bandwidth allocation algorithm for WMNs. Section IV presents the two-level packet scheduling algorithm. Section V evaluates the performance of our solution. Section VI draws the conclusion.

II. Network Model and maxmin Model

A. Network Model

We adopt infrastructure (backbone) WMNs as our network model. In an infrastructure WMN, mesh routers interconnect through wireless links to form a communication backbone. Clients are connected to mesh routers via wired or wireless means. Clients connected to different routers communicate with one another through multi-hop wireless paths. To simplify the discussion, we assume that two different channels are used for wireless communication *between mesh routers* and wireless

*This work was supported in part by the US National Science Foundation under grant CNS-0644033 and grant CNS-0721731.

978-1-4244-2324-8/08/$25.00 © 2008 IEEE.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2008 proceedings.

communication *between a router and a client*. Therefore, router-router communication does not interfere with router-client communication. This paper focuses on router-router communication. We assume the existence of the IEEE 802.11 DCF MAC protocol. Two wireless links (between routers) *contend* if they cannot transmit simultaneously.

A WMN connects to the Internet through one or multiple *gateway mesh routers*. All communication traffic from clients of one router to clients of another router constitutes an *internal flow*. All communication traffic from clients of one router to the Internet or in the reverse direction constitutes an *external flow*. Internal flows are common in WMNs deployed for campus communication. External flows are common in WMNs deployed in residential areas for Internet access. In our abstract model, we consider the beginning router of a flow as the data source of the flow and the ending router as the destination. We assume the existence of a routing protocol that establishes a routing path for each flow.

This paper studies end-to-end flows, which are referred to simply as *flows*. A flow consists of one or more single-hop flows, which are called *subflows*. Two subflows *contend* if they are carried by the same link or two contending links. Two flows *contend* if any of their subflows contend.

B. Maxmin Model

Each end-to-end flow is assigned a *nominal flow weight*. The network is expected to allocate bandwidth to the flows in proportion to their nominal weights whenever possible. The nominal flow weights can be decided based on administrative policies (e.g., higher weights for more important flows), commercial policies (e.g., higher weights for customers who pay more), or incentive policies (e.g., higher weights for flows whose source routers contribute more in carrying others' traffic).

Each flow is entitled to a fair share of network bandwidth in proportion to its weight. It is well known that maintaining fairness and maximizing network throughput are contradictive goals [4]. Stricter fairness can be achieved often at the expense of lower network throughput. Some previous studies focused more on throughput optimization under certain basic, relaxed fairness criteria [12], [5]. This paper puts more focus on fairness. Specifically, we want to achieve the classical maxmin fairness among end-to-end flows in WMNs. The maxmin fairness requires the network to first maximize the smallest flow rate, then maximize the second-smallest flow rate, and so forth.

The classic maxmin model *for wired networks* is described as follows. Given a set Q of resources (i.e., links), a capacity b_q for each resource $q \in Q$ (i.e., bandwidth), a set F of flows, a nominal weight w_f for each flow $f \in F$, and a routing path p_f for each flow f, the problem is to assign a rate r_f for each flow f such that

1) $\forall q \in Q, \ \sum_{f \in F, \ q \in p_f} r_f \leq b_q$, and
2) for any flow $f \in F$, its rate r_f cannot be increased without decreasing the rate $r_{f'}$ of another flow f', for which $r_{f'}/w_{f'} \leq r_f/w_f$.

The set of rates $R = \{r_f \mid f \in F\}$ that satisfy the above conditions are called the *maxmin rates*.

The above model assumes that each resource has a fixed capacity and that a resource can appear in a flow's routing path at most once. In order to apply this model to WMNs, we have to identify what the resources are. Wireless links cannot be used as the resources because they do not have individually fixed capacities. Following Huang and Bensaou's work [3] which considers only one-hop flows, we shall use "cliques" from the contention graph as the resources, which will be explained in detail in Section III. However, in order to accommodate the "clique resources" in the context of end-to-end flows, we must generalize the maxmin model first in the following section to allow a resource to appear in a flow's routing path for multiple times.

III. A GENERALIZED MAXMIN MODEL

In this section, a generalized maxmin model is introduced. By applying this model, each flow in the network is assigned a maxmin fair share that will be used by the packet scheduling algorithm presented in Section IV.

A. Resources in WMNs

In wired networks, all flows that pass a link between two routers compete for the link bandwidth. The links serve as the resources in the classical maxmin model. In a WMN, the medium is shared by a group of nearby mesh routers. Not only flows passing the same wireless link but also those passing nearby wireless links compete for the shared channel capacity.

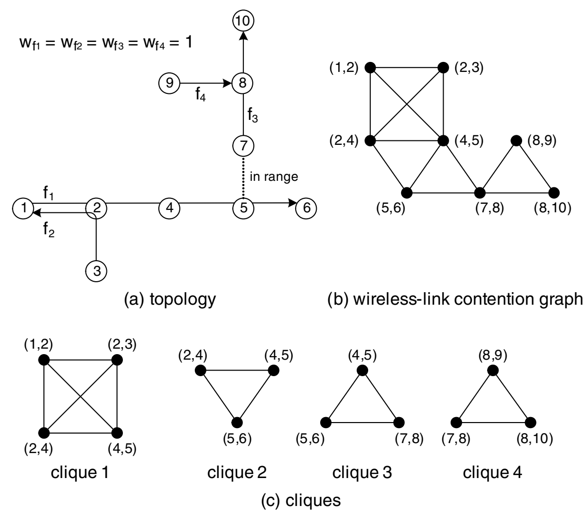

Fig. 1: A wireless-link contention graph and cliques

A *wireless-link contention graph* can be employed to describe the spatial contention relationship among contending links. Vertices in a wireless-link contention graph represent wireless links in the corresponding network topology. Two vertices are connected if the corresponding links contend with each other. A wireless link is *idle* if there is no flow passing it. A *simplified* wireless-link contention graph can be constructed from a network topology with all idle links removed. An example of simplified wireless-link contention graph is given by Fig. 1 (b).

A *clique* is a complete subgraph with a link between every pair of nodes. A *maximum clique* is a clique that is not contained in another clique. In this paper, we refer to maximum cliques as cliques henceforth. A clique in a wireless-link contention graph represents a group of mutually contending wireless links in which only one link can be in transmission at any time. The channel bandwidth is shared by all wireless links of a clique.

978-1-4244-2324-8/08/$25.00 © 2008 IEEE.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2008 proceedings.

The cliques from the wireless-link contention graph can be used as resources.

Following the routing path of a flow, we can obtain a sequence of cliques that the flow passes. When a flow passes multiple links of a clique, we consider the flow passes the clique multiple times. For a wireless link belonging to multiple cliques, if a flow passes this link, we consider the flow passes those cliques in turn.

B. Generalized Maxmin Model

In classic maxmin model, a resource can appear in a flow's routing path at most once. Motivated by the above characteristics of WMNs, in this subsection, we generalize the classic maxmin model and then apply the generalized model to WMNs.

In the generalized model, a resource is allowed to appear in a flow's routing path for multiple times in different positions.The number of appearances of resource $q \in Q$ in flow f's path p_f is denoted by n_f^q. We have the following feasibility constraint for a set of flow rates $R = \{r_f \mid f \in F\}$.

$$\sum_{f \in F,\ q \in p_f} n_f^q \times r_f \leq b_q \quad (1)$$

A set of rates that satisfies the above constraint is said to be *feasible*. It is *maxmin fair* if it is feasible and, for each $f \in F$, r_f cannot be increased while maintaining feasibility without decreasing $r_{f'}$ for another flow f', for which $r_{f'}/w_{f'} < r_f/w_f$. Our goal is to find a set of flow rates that is maxmin fair.

An algorithm that calculates the maxmin rates of the flows can be found in [13]. Below we adapt it for the generalized maxmin model. For each resource q, compute the average capacity share available for a unit weight of one appearance of each passing flow, which is $\frac{b_q}{\sum_{f \in F,\ q \in p_f} n_f^q \times w_f}$. Find the global bottleneck resource that has the smallest capacity share. Assign an equal share of the resource's capacity to a unit weight of one appearance of each passing flow f. It can be proved that the equal share is the maxmin normalized rate r_f/w_f of this flow. Remove the bottleneck resource and the flows passing it from the network. When a flow f is removed, the capacity of each resource q on its routing path is reduced by $n_f^q \times r_f$. Repeat the above process until every flow is assigned a rate and removed from the network.

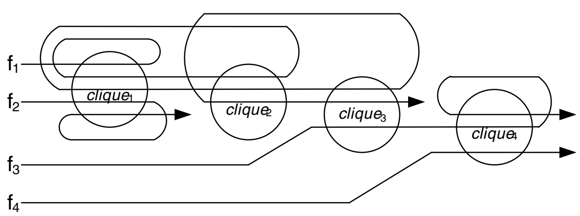

Fig. 2: Flows described by the generalized maxmin model

The algorithm described above can be used to compute the maxmin flow rates. The rate of a flow reflects the amount of the bandwidth a flow should received at each node on its routing path. The larger rate a flow has, the more bandwidth a flow should receive at each node on its path. The flow rates calculated by the generalized maxmin model are also called flows' *maxmin fair shares*. By applying the generalized maxmin model, the flows in Fig. 1 can be redrawn in Fig. 2, where the resources (circles) are cliques. If we assume the bandwidth capacities of all cliques are the same and are normalized to *one* unit, then the maxmin fair shares of f_1, f_2, f_3, f_4 are $1/5, 1/5, 1/3, 1/3$, respectively.

Some previous works, e.g., [3], [5], use channel capacity as clique capacities. If the same way is followed, for some link contention graphs, flow fair shares calculated by the maxmin algorithm are upper bounds of the true maxmin fair shares. One example is the odd cycles of length at least 5 without chords in link contention graph [10]. To solve this problem, we use the *effective channel capacity* of a clique q as b_q in our model. A node measures the *effective bit rates* of its incident links. The concept of effective bit rate is similar to the one in [14], which incorporates link layer details. A clique's effective channel capacity can be obtained by summing up the effective bit rates of all links of that clique.

In infrastructure WMNs, mesh routers have relatively strong computing capability and stable positions which make the centralized implementation of the algorithm feasible. The implementation can also be distributed. Nodes only work on local link contention graph which is much smaller than the global one. The work of clique decomposition is reduced remarkably. Some distributed maxmin algorithms for wireline networks (e.g., [15]) could be customized to calculate flow maxmin fair shares.

IV. Packet Scheduling Algorithm

We have discussed how to calculate maxmin flow fair shares. These fair shares replace the nominal flow weights and become the *effective flow weights* used by the packet scheduling algorithm that will be described in this section.

A. Overview

The basic idea of our scheduling algorithm is to let each subflow receive bandwidth proportionally to its effective weight, which is equal to the effective weight of the flow it belongs to. This idea is similar to the scheduling in wired networks. However, scheduling in multihop wireless networks is more complex. In wired networks, contending subflows are backlogged in the same router. All scheduling work could be done within this router. In a wireless network, contending subflows may reside in different nodes, which could be as far as several hops away. They need to cooperate with each other to guarantee each subflow receive appropriate bandwidth. Our scheduling method includes two components:

- Inter-node scheduling. If we consider all packets in a router form a virtual queue, the weight of this virtual queue equals to the router's effective weight, which is the sum of the effective weights of all backlogged flows in the router. Our method schedules the transmissions of the packets from virtual queues to guarantee that the bandwidth each virtual queue obtains is proportional to its weight.
- Intra-node scheduling. Inside a router, packets from different flows are queued separately. The intra-node scheduling allocates the bandwidth obtained by the router to the backlogged flows proportionally to their effective weights. Some queuing algorithms proposed for wired networks (e.g., [16]) can be adopted to achieve this.

The rest of this section describes the inter-node scheduling algorithm that is based on the 802.11 DCF with RTS-CTS-DATA-ACK handshake.

B. Inter-node scheduling

Let B be the set of all backlogged flows in the network, B_i the set of backlogged flows at router i. The effective weight of flow f_i is denoted by w'_i and then the effective weight of router i is $\hat{w}_i = \sum_{j \in B_i} w'_j$. Each router i maintains a counter C_i. When packet P_i^k (the kth packet from router i) becomes the next-to-send packet of router i, it is assigned a tag $T_i = C_i$. Then $C_i = C_i + L_i^k / \hat{w}_i$, where L_i^k is the length of packet P_i^k. In order to achieve the short-term fairness, all counters are reset to zero every ϕ seconds at the same time, assuming clocks of all routers are loosely synchronized.

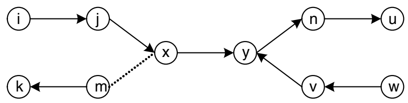

Fig. 3: Scheduling among contending nodes

Fig. 3 shows all possible contending packet transmissions within two hops away from router x and router y. In Fig. 3, circles represent routers. A line between two routers means they are within the transmission range of each other. An arrow from router x to y means the next-to-send packet of x need to be transmitted to y. Given the example in Fig. 3, the transmission from x to y conflicts with all other transmissions indicated by the arrows in Fig. 3.

If a node x has a packet to transmit, the *contending node set* of x, denoted by Ω_x, is defined as the group of nodes that are competing for the media access with x. In Fig. 3, $\Omega_x = \{i, j, m, y, n, v, w\}$. Let $\Omega_x^+ = \Omega_x \cup \{x\}$. When x has a packet to transmit and its backoff timer becomes zero, it should compare its tag with those of the nodes in Ω_x. Ideally, the packet from x should be transmitted immediately if its tag is the smallest. Otherwise, x's transmission should be withheld until all packets from Ω_x with smaller tags are transmitted first. To describe how to determine if a transmission should be withheld, we need first define three variables for each router x:

- T_x^s: *sending tag* of x, which is the tag of the next-to-send packet of x. If x does not have any packet to send, T_x^s is set to a very large value MAXTAG.
- T_x^r: *receiving tag* of x, which is the smallest tag of the packets to be received by x from its neighbors.
- T_x^n: $T_x^n = min_{i \in N_x}\{T_i^s, T_i^r\}$, where N_x is x's neighbor set.

When x has a packet to be transmitted to y, if $T_x^s > T_x^n$ or $T_x^s > T_y^n$, x can know it does not have the smallest tag in Ω_x^+ and the transmission should be withheld. In order to obtain most up-to-date T_x^n and T_y^n, RTS, CTS, DATA and ACK packets can piggyback necessary tags. Each router maintains a table to keep track of its neighbors' tags.

However, it is difficult to enforce the above strict conditions for each transmission. The reason is that sender x cannot always have the fresh tags of its neighbors, especially T_y^n from receiver y, which are based on the tags of nodes as far as three hops away from x. Stale tags may cause deadlocks. To avoid potential deadlocks, a heuristic method is used by x to estimate T_y^n. The basic idea is to estimate the increment rate of T_y^n, denoted by r_y^n. For each $i \in N_x$, besides T_i^n, x also records r_i^n it estimates and the time t_i when T_i^n gets updated. When x needs to transmit a packet to y, x uses $\hat{T_y^n}$ instead of T_y^n to check the second condition, where

$$\hat{T_y^n} = T_y^n + r_y^n \times (t - t_y) \tag{2}$$

t is the current time. Once T_y^n gets updated and becomes larger, the new r_y^n is computed as:

$$r_y^n = \alpha \times r_y^n + (1 - \alpha) \times \frac{\Delta T_y^n}{t - t_y} \tag{3}$$

where α is a parameter to control the influence of T_y^n's new increment rate on r_y^n. If above approach is employed, $\hat{T_y^n}$ will eventually be increased large enough such that the second withholding condition will become false.

We have discussed that ideally router x should withhold its transmission until its sending tag becomes the smallest in Ω_x^+. Actually, we do not have to enforce such strict conditions. x should be allowed to transmit a packet as long as its sending tag is not "very large" compared to the tags of the nodes in Ω_x. Now the two transmission withholding conditions can be formally given as follows. When router x need transmit a packet to y, the transmission should be withheld if:

1) $T_x^s > T_x^n + \beta \times L/\hat{w}_x$, or
2) $T_x^s > \hat{T_y^n} + \beta \times L/\hat{w}_x$

where L is the packet length, β is a parameter greater than zero. β specifies how many packets x is allowed to transmit ahead of its contending nodes. By introducing β, x do not have to wait until it has the smallest sending tag in Ω_x^+, but will withhold its transmission when its sending tag is "much larger" than those of the nodes in Ω_x.

V. Performance Evaluation

In this section, the proposed solution that achieves end-to-end maxmin fairness (referred to as MMF) will be evaluated through simulations. The simulation environment settings are described as follows. The channel capacity is 11Mbps. The transmission range of a mesh router is 250 meters. Each data packet is 1024 bytes long. Per-flow queuing is adopted by each router. We assume each source sends data at a constant bit rate (CBR) of 700 packets per second. The length of each simulation session is 200 seconds. The parameters of MMF are set as follows: ϕ is 10 seconds, α is 0.9, and β is 6.

We compare the performance of MMF with (1) 802.11 DCF (abbreviated as 802.11); and (2) the two-phase protocol (abbreviated as 2PP) proposed in [5]. We compare the algorithms from two aspects: *end-to-end flow fairness* and *spatial reuse of spectrum.*

To evaluate the end-to-end fairness, we adopt the maxmin fairness index [13] (denoted by I_{mm}) and the equality fairness index [17] (denoted by I_{eq}).

$$I_{mm} = \frac{min_{f \in F}\{r_f\}}{max_{f \in F}\{r_f\}}, \qquad I_{eq} = \frac{(\sum_{f \in F} r_f)^2}{|F| \sum_{f \in F} (r_f)^2}$$

I_{mm} measures the ratio of the smallest flow rate to the largest flow rate. I_{eq} measures the overall equality among the flow rates; its value approaches to one if the rates of all flows approach toward equality.

To measure the spatial reuse of spectrum, we employ the *effective network throughput* U, which is defined as $\sum_{f \in F} r_f \times l_f$, where l_f is the number of hops on the routing

		802.11	2PP		MMF	
flow	length	thro.	effe. weight	thro.	effe. weight	thro.
$\langle 1,6 \rangle$	4	114.44	1.00	115.60	1.00	173.74
$\langle 3,1 \rangle$	2	198.82	2.50	288.65	1.00	173.74
$\langle 7,10 \rangle$	2	272.21	1.00	114.36	1.67	291.49
$\langle 9,8 \rangle$	1	414.32	6.00	682.03	1.67	292.24
effe. network thro.		1814.13	1950.46		1917.66	
I_{mm}		0.276	0.168		0.595	
I_{eq}		0.837	0.627		0.940	

TABLE I: Simulation results on the topology in Fig. 1

path of flow f. The packets dropped by the intermediate nodes do not count towards the effective network throughput as they do not contribute to end-to-end throughput. The effective network throughput gives us a measurement for network bandwidth utilization and the efficiency of a protocol.

We present simulation results in two network scenarios: a simple network topology shown in Fig. 1 and a complex network topology that will be described later. All flows in both scenarios have the equal nominal weights. In the rest of this section, the unit of the flow or network throughput is packets per second (PPS).

The simulation results of the example in Fig. 1 are shown by Table I. The length of a flow is the number of its subflows. MMF shows good end-to-end fairness and comparable bandwidth utilization. In 2PP, the objective of the basic fairness model is to maximize the total end-to-end throughput. Thus single hop flow $\langle 9,8 \rangle$ has much higher rate than other flows.

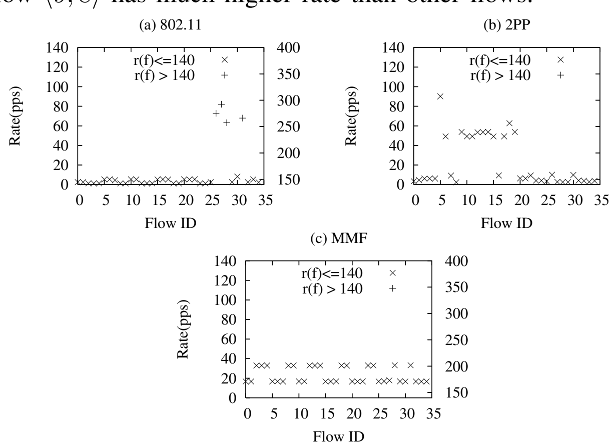

Fig. 4: Flow rates of the complex scenario

	802.11	2PP	MMF
effe. network thro.	1550.15	998.86	1528.45
I_{mm}	0.004	0.026	0.500
I_{eq}	0.136	0.453	0.895

TABLE II: Simulation results of the complex scenario

The complex scenario simulates the traffic in the backbone of a WMN. 27 nodes are placed in a 900×900 region, in which 25 are non-gateway nodes and 2 are gateway nodes. Gateway nodes are evenly placed in the horizontal midline of the region. The region is divided into 25 grids. Each non-gateway node is placed into a grid. The location of a non-gateway node in its grid is randomly chosen. A non-gateway node connects to the Internet through the nearest gateway node. Every non-gateway node has a download flow from its gateway node. 5 non-gateway nodes are randomly picked to have 5 upload flows to their gateway nodes. We also randomly create 5 internal flows among non-gateway nodes. The simulation results are shown in Fig. 4 and Table II. In Fig. 4, the flow rates that are under 140 pps use the numbers on the left vertical axis; the flow rates above 140 pps use the numbers on the right vertical axis.

For both 802.11 and 2PP, many flows have very low rates. For 802.11, all high rate flows are no longer than 2 hops. The reason is that for a long flow, many packets are dropped before arriving the destination due to buffer overflow caused by inconsistent subflow rates. For 2PP, all high rate flows have only one hop. The reason is that the basic fair share guaranteed for each flow in such large network is very small. Bandwidth is allocated to single hop flows whenever possible to maximize total end-to-end throughput. MMM shows much better fairness than the other two and also achieves good bandwidth utilization.

VI. CONCLUSION

In this paper, we have studied the problem of end-to-end fairness in WMNs. A generalized maxmin model is presented. This model is applied to WMNs by considering the unique characteristics of wireless networks to provide end-to-end maxmin fairness. A two-level packet scheduling algorithm is proposed to make each flow receive bandwidth at each node on its routing path proportionally to its maxmin fair share calculated by the model. Simulation results have demonstrated the effectiveness of the proposed solution in enhancing end-to-end fairness.

REFERENCES

[1] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," *Computer Networks Journal (Elsevier)*, vol. 47, no. 4, pp. 445–487, Mar 2005.

[2] J. M. Faffe, "Bottleneck Flow Control," *IEEE Transactions on Communications*, vol. COM-29, no. 7, pp. 954–962, July 1981.

[3] X. L. Huang and B. Bensaou, "On max-min fairness and scheduling in wireless ad-hoc networks: analytical framework and implementation," *Proc. of ACM MobiHoc'01*, pp. 221 – 231, 2001.

[4] H. Luo, S. Lu, and V. Bharghavan, "A new model for packet scheduling in multihop wireless networks," *Proc. of ACM MOBICOM'00*, 2000.

[5] B. Li, "End-to-End Fair Bandwidth Allocation in Multi-hop Wireless Ad Hoc Networks," *Proc. of ICDCS'05*, pp. 471–480, June 2005.

[6] L. Tassiulas and S. Sarkar, "Maxmin Fair Scheduling in Wireless Networks," *Proceedings of IEEE INFOCOM'02*, 2002.

[7] S. Sarkar and L. Tassiulas, "End-to-end Bandwidth Guarantees Through Fair Local Spectrum Share in Wireless Adhoc Networks," *IEEE Transactions on Automatic Control*, vol. 50, no. 9, September 2005.

[8] S. Chen and Z. Zhang, "Localized Algorithm for Aggregate Fairness in Wireless Sensor Networks," *Proc. of ACM MobiCom*, September 2006.

[9] Y. Qiu and P. Marbach, "Bandwidth Allocation in Wireless Ad-Hoc Networks: A Price-based Approach," *Proc. of IEEE INFOCOM'03*, March 2003.

[10] Z. Fang and B. Bensaou, "Fair Bandwidth Sharing Algorithms based on Game Theory Frameworks in Wireless Ad-Hoc Networks," *Proc. of IEEE INFOCOM'04*, March 2004.

[11] Y. Yi and S. Shakkottai, "Hop-by-Hop Congestion Control over a Wireless Multi-hop Network," *Proc. of IEEE INFOCOM'04*, March 2004.

[12] H. Luo, J. Cheng, and S. Lu, "Self-Coordinating Localized Fair Queueing in Wireless Ad-Hoc Networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 1, January-March 2004.

[13] D. Bertsekas and R. Gallager, "Data networks, 2nd ed." *Prentice-Hall Inc*, 1992.

[14] Y. Bejerano, S.-J. Han, and L. E. Li, "Fairness and load balancing in wireless lans using association control," *Proc. of ACM MobiCom'04*, 2004.

[15] J. Mosely, "Asynchronous distributed flow control algorithms, Ph.D. thesis," *MIT, Dept. of Electrical Engineering and Computer Science*, 1984.

[16] J. Bennett and H. Zhang, "WF^2Q: Worst-case fair weighted fair queueing," *Proc. of INFOCOM'96*, pp. 120–128, Mar 1996.

[17] D. Chiu and R. Jain, "Analysis of the Increase/Decrease Algorithms for Congestion Avoidance in Computer Networks," *Journal of Computer Networks and ISDN*, vol. 17, no. 1, pp. 1–14, June 1989.